Reversible Automata and Petri Nets, Duality and Representation: the Synthesis Problem

Eric Badouel

September 25, 1998

Chapter 1

Introduction

Since their introduction in the early sixties [75], Petri nets have come to play a pre-eminent role in the formal study of the behaviour of concurrent and distributed systems. Petri nets are a very simple and natural extension of automata in which states have a distributed nature and for which the occurrence of an event relies on local conditions. Therefore the study of their mathematical properties becomes a manageable task; in particular much effort have been devoted to decidability and complexity issues for Petri nets. Moreover, a lot of techniques and automated tools [76] support the verification of properties of systems modelled by Petri nets. For instance one can decide by constructing its coverability tree whether a Petri net is bounded [60], i.e. whether it is a finite state system. Reduction techniques [22] and linear algebra techniques have also received wide attention [66, 96]. Finally, Petri net is a graphical tool which is of a practical interest for the description and design of concurrent systems. For an extensive presentation of the theory and practice of Petri nets I refer to the monographs [74, 85, 87, 57, 36, 35, 86] and the survey [68] (the reader may also wish to consult the Petri net mailing list [77] maintained in Aarhus University). For the purpose of the present study the definitions of a Petri net and of its associated marking graph (via the token game) are the only prerequisites.

The problem of net synthesis is a graph-theoretic problem: given a labelled graph representing the sequential behaviour of some distributed system, the synthesis problem consists in deciding whether it is isomorphic to the marking graph of some Petri net. The synthesis problem was originally solved for the class of elementary net systems by Ehrenfeucht and Rozenberg [42, 43] (see also [37]). Their approach was based on the notion of *regions in graphs* defined as sets of nodes liable to represent extensions of places with boolean values: these are set of nodes uniformly entered, exited or left invariant by all transitions bearing the same label. Ehrenfeucht and Rozenberg gave a characterization of those automata which are isomorphic to the marking graphs of elementary nets in term of two *separation axioms* the first of which states that there exists sufficiently many regions to distinguish every pair of distinct states in the automaton. The second axiom of separation states that for every action and every state at which this action is not enabled there exists a region which "inhibits" this action in this state. This solution has been extended to various kinds of nets [67, 73, 9, 21, 38, 39, 13, 27, 79]. All variants of net synthesis have a representation theorem based on the same separation axioms but with variant notions of region. An interpretation of these separation axioms is the following: a region viewed as an abstract place can be associated with a quotient automaton representing the projection of the automaton relative to the "content" of that place, the automaton is then isomorphic to the marking graph of some net if and only if it is isomorphic to the synchonized product of those automata (associated with regions). The first separation axiom expresses that states of the original automaton can injectively be encoded as vectors of local states (each local state giving its value relative to some region). The second separation axiom states that an action is not enabled in some state of the automaton if and only if there exists some component automaton such that this action is not enabled in the corresponding local state. The synthesis problem then reduces to deciding whether an automaton can be presented as a synchronized product of automata of a given type.

This observation leads to a uniform presentation of the net synthesis problem [10] parametric on the type on nets: regions appear as morphisms from the underlying transition system of the automaton to a classifying transition system, called the *type of nets*, which characterizes the behaviour of the considered class of nets. We obtain, for each type of nets, a dual adjunction between a category of transition systems and a category of nets. whose kernel corresponds on one side to saturated net systems (with all redudant places) and to separated automata on the other side (i.e. automata that satisfy the separation properties). In [3] the presentation is simplified by observing that for a fixed alphabet one ends up with a Galois connection (between ordered sets) thus avoiding all the machinery of category theory. The approach is extended to the step semantics of nets by requiring that the type of nets be itsef a step transition system. This general framework allows a uniform presentation of the synthesis of elementary net systems [42, 43] and vector addition systems [21] from their sequential marking graphs and the synthesis of condition-event nets [73] and Petri nets [67] from their concurrent marking graphs.

Applications of the net synthesis from automata may be thought in the engineering of distributed software where it can be used as a technique of parallelization of sequential programs. Actually, net synthesis gives means to build parallel systems realizing behaviours specified by finite transition systems (e.g. communication protocols). More precisely the net synthesized from an automaton exhibits the maximal parallelism compatible with the structure of the automaton in the sense that for any n-uple of actions labelling an n-dimensional hypercube in the automaton the associated n-uple of transitions in the synthesized net are concurrently enabled at the marking associated with the origin of the hypercube. However, for the time being this technique suffers from some limitations, due to the complexity of the algorithms, that restrict its use in practical applications. The synthesis problem for elementary net systems is NP-complete [7], and although the synthesis problem for vector addition systems is solved in polynomial time [6], its solution is based on Khachiyan's ellipsoid method (see [91]) which in practice is usually replaced by the, theoretically inefficient, simplex method: actually, even though the latter is not a polynomial-time method it has a better average complexity than the ellipsoid method. However some tools have been developped. A computer assisted solution to the state encoding problem for asynchronous circuits supported by the tool `Petrify`, is described in [30]. The algorithm compute only minimal regions (for elementary net systems) of a transition system symbolically represented by Binary-Decision Diagrams [25, 26]. A computer assisted solution to the distribution of protocols, based on regions for vector addition systems and supported by the tool SYNET, is described in [28].

The polynomial time algorithm proposed in [6] for the synthesis of vector addition systems from finite automata is based on polynomial algorithms discovered in the late seventies for linear programming over the rational field. This algorithm has been adapted to the synthesis of Petri net from its sequential or concurrent marking graph [12] to the synthesis of Petri nets from languages [6, 33] and to the synthesis of stratified Petri nets [13], a weaker form of Valk's self-modifying nets [100, 101].

In the next chapter we present Ehrenfeucht and Rozenberg's solution to the synthesis problem for elementary net systems based on the notion of *region* in labelled graphs. Their representation result states that an automaton is isomorphic to the case graph of an elementary net system if and only if two separation properties are satisfied. We then indicate why this problem is NP-complete.

As already mentionned, the problem of Petri net synthesis can be construed as the representation of an automaton as a synchronized product of very basic automata (the atoms of the representation). For elementary net systems and for vector addition systems these atomic automata are full restrictions of Cayley graphs of $\mathbb{Z}/3\mathbb{Z}$ and $\mathbb{Z}$ respectively. In chapter three of this document we give a classification of the various ways an automaton can be fully embedded in a Schreier graph of a group. Of course such an automaton should be reversible in the sense that each event induces a local bijection on states, and conditions should be fulfilled by the representing group which are similar to the two separation properties. We then describe the computation of the canonical representation of a commutative automaton (automaton that fully embedds in the Cayley graph of an abelian group).

In chapter four we turn our attention to the state graphs of vector addition system. They are commutative automata and we conjecture that they are torsion-free, i.e. that their canonical representation does not contain torsion element. We state the generalization of Ehrenfeucht and Rozenberg representation theorem to the context of vector addition system and derive a polynomial time algorithm for the synthesis of vector addition systems. An alternative algorithm using the canonical representation of commutative automata is given, however the correction of this algorithm relies on the conjecture that state graph of vector addition systems are torsion-free.

In the chapter five we describe the duality between nets and automata, this duality is induced by a schizophrenic object and is parametric on the type of nets. This presentation can be extended to higher dimensional automata in order to take into account the concurrent behaviour of the net systems. All known variants of net synthesis can be obtained as instances of this construction corresponding to particular types of nets.

In the concluding chapter, we mention some miscellaneous results that have not been presented in this document and some possible directions for further research.

As witnessed by the list of publications, the work reported here is mainly a joint work with Philippe Darondeau and is the result of our long and fruitful collaboration. Various people in Rennes, including Vicente Sanchez-Leighton [88], Luca Bernardinello [20], Carlo Ferigato [44], Vincent Schmitt [90], Danièle Quichaud and Benoît Caillaud, give us the benefit of their views and ideas on the issues of duality, representation theory and net synthesis; even though it does not necessarily led to common publications.

Chapter 2

Elementary Net Systems

We first present Ehrenfeucht and Rozenberg's solution to the synthesis problem for elementary net systems based on the notion of *region* in labelled graphs. We then report on Desel and Reisig's result on *admissible* sets of regions, and on Bernardinello's work on *minimal* regions. Finally we present our own contribution namely that the synthesis problem for elementary net systems is NP-complete. This chapter is mostly an excerpt from [7].

2.1 Elementary Net Systems

We recall here a few basic definitions needed for self containment.

Definition 2.1.1 (C/E Net)
A condition/event net is a triple $N = (P, E, F)$ where P is a set of places or conditions, E is a set of events or actions disjoint from P, and $F \subset E \times P \cup P \times E$ is a bipartite relation between places and events called the flow relation. The flow graph is assumed to have no isolated element, in the sense that

$$\forall x \in E \cup P \;\; \exists y \in E \cup P \;\; [(x, y) \in F \;\vee\; (y, x) \in F]$$

In the graphical representation of nets, places are depicted by circles and actions by boxes. Since the flow relation is bipartite there is no arc between two places or between two actions. We adopt the notations ${}^\bullet x = \{y/F(y, x)\}$ and $x^\bullet = \{y/F(x, y)\}$ for the respective pre-set and post-set of an element $x \in P \cup E$. A C/E net is said to be *simple* when

$$\forall x, y \in P \cup E \quad ({}^\bullet x = {}^\bullet y \text{ and } x^\bullet = y^\bullet) \Rightarrow x = y$$

A *marking* of a C/E Net is a set of places representing a *state* in the evolution of the net by the set of conditions it satisfies. A *transition system* is a triple (S, E, T) consisting of a set of *states* S, a set of *events* E, and a transition relation $T \subseteq S \times E \times S$. We shall write $s \xrightarrow{e} s'$ as an abbreviation for $(s, e, s') \in T$. An *automaton* $A = (S, E, T, s_0)$ is a transition system together with an initial state

$s_0 \in S$. All the possible evolutions of a condition/event net are described in the transition system $(\mathcal{M}, E, T)$ whose set of states is the set $\mathcal{M}$ of markings, and whose transitions are given by

$$M \xrightarrow{e} M' \quad \textbf{iff} \quad {}^\bullet e \subseteq M \;\text{ and }\; (M \setminus {}^\bullet e) \cap e^\bullet = \emptyset \;\text{ and }\; M' = (M \setminus {}^\bullet e) \cup e^\bullet$$

Places in ${}^\bullet e \cap e^\bullet$, called the *side conditions* [1] of e, are just tested upon: these conditions are necessary for firing event e and they still hold thereafter. The conditions in ${}^\bullet e$ (*preconditions* of e) which are not side conditions of e are also necessary for firing event e but they no longer hold after it has been fired. Symmetrically, the conditions in $e^\bullet$ (*postconditions* of e) which are not side conditions of e hold never in markings giving concession to event e, and hold always after e's executions. A net free from side conditions ($\forall e \in E \; {}^\bullet e \cap e^\bullet = \emptyset$) is said to be *pure*. The transition relation of pure C/E nets simplifies to

$$M \xrightarrow{e} M' \quad \textbf{iff} \quad {}^\bullet e \subseteq M \;\text{ and }\; M \cap e^\bullet = \emptyset \;\text{ and }\; M' = (M \setminus {}^\bullet e) \cup e^\bullet$$

Definition 2.1.2 (Elementary Net System)
An elementary net is a pure and simple C/E net with no isolated element. An elementary net system $\mathcal{N} = (P, E, F, M_0)$ is an elementary net together with an initial marking M_0 such that every event $e \in E$ may be fired at some marking reachable from M_0.

Definition 2.1.3 (Case Graph)
The case graph of an elementary net system $\mathcal{N} = (P, E, F, M_0)$ is the automaton $\mathcal{N}^* = (S, E, T, s_0)$ whose initial state s_0 is the initial marking M_0 of $\mathcal{N}$ and whose underlying transition system (S, E, T) is the induced restriction of the transition system generated from the net (P, E, F) on the set of markings reachable from M_0.

Observation 2.1.4 *If an automaton $A = (S, E, T, s_0)$ is the case graph of an elementary net system, then it satisfies the following:* (i) *it has no loop: $s \xrightarrow{e} s' \;\Rightarrow\; s \neq s'$,* (ii) *it has no multiple transitions between states: $(s \xrightarrow{e_1} s' \;\wedge\; s \xrightarrow{e_2} s') \Rightarrow e_1 = e_2$,* (iii) *it is reduced: $\forall e \in E \quad \exists s \xrightarrow{e} s'$, and* (iv) *it is reachable: $\forall s \in S \quad s_0 \xrightarrow{*} s$ where $\xrightarrow{*} = (\bigcup_{e \in E} \xrightarrow{e})^*$.*

2.2 Regions in Graphs

In this section, we introduce regions in graphs as they were defined originally by Ehrenfeucht and Rozenberg, then we show an equivalent definition of regions as morphisms of transition systems, and last we define the dual of a graph as the net synthesized from all regions in that graph, seen as places fully specified by the attached flow arcs.

Let $A = (S, E, T, s_0)$ be an automaton. Solving the synthesis problem for this automaton in the context of elementary net systems consists in deciding

[1] for some authors however there is no side condition for an event in a C/E net.

whether there exists an elementary net system $\mathcal{N} = (P, E, F, M_0)$ with case graph $\mathcal{N}^*$ isomorphic to A (i.e. identical to A up to a bijective renaming of states and transitions). Suppose such an isomorphism exists between A and $\mathcal{N}^*$, then each state of the automaton A may be identified with a marking of the elementary net system $\mathcal{N}$ and a binary relation $\models \subseteq S \times P$ may be defined between states of A and places of N by setting $s \models x$ (read "s satisfies x") if and only if the condition x belongs to the marking associated with the state s. The elementary net $N = (P, E, F)$ provides a faithful *set-theoretic representation* of the transition system $T = (S, E, T)$: the pair of mappings $\| \cdot \|_S : S \to 2^P$ and $\| \cdot \|_E : E \to 2^P \times 2^P$ defined as $\|s\|_S = \{x \in P |\ s \models x\}$ (the marking associated with s) and $\|e\|_E = < {}^\bullet e, e^\bullet >$ are injective and the transition relation of T satisfies

$$s \stackrel{e}{\to} s' \in T \quad \textbf{iff} \quad \|s\|_S \setminus \|s'\|_S = {}^\bullet e \ \ \wedge \ \ \|s'\|_S \setminus \|s\|_S = e^\bullet$$

In order to construct a representation for a given transition system $T = (S, E, T)$, one has to guess an adequate set of places (the atomic symbols of this representation). For that purpose one may use reverse reasoning, starting from the assumption that an adequate representation $N = (P, E, F)$ has been found. Then each condition $x \in P$ can be represented by the set $\|x\|_P = \{s \in S |\ s \models x\}$ of states of T satisfying this condition. This set $\|x\|_P$, called the *extension* of x, satisfies the predicate:

$$\begin{array}{rcl} \textbf{Region}(X) & \equiv & \textit{for every event } e \in E \quad : \\ s \stackrel{e}{\to} s' & \Rightarrow & (s \in X \text{ and } s' \notin X) \\ \textit{or} \quad s \stackrel{e}{\to} s' & \Rightarrow & (s \notin X \text{ and } s' \in X) \\ \textit{or} \quad s \stackrel{e}{\to} s' & \Rightarrow & (s \in X \text{ iff } s' \in X) \end{array}$$

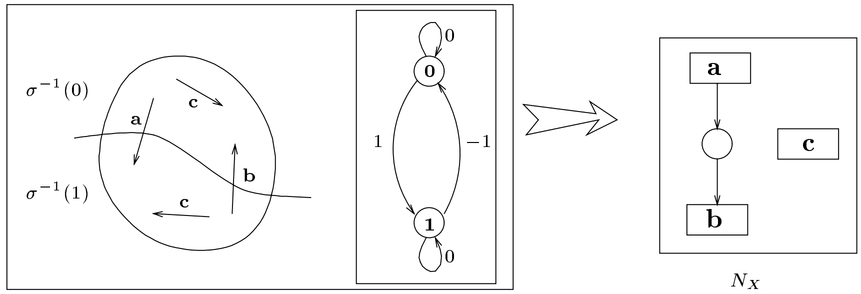
The three cases above are met respectively for $x \in {}^\bullet e$, $x \in e^\bullet$, and $x \notin {}^\bullet e \cup e^\bullet$ in N. Now forgetting about N, we call a *region* in T any subset $X \subseteq S$ satisfying $\textbf{Region}(X)$.

Observation 2.2.1 *Let $\mathcal{N}$ be an elementary net system with set of places P and marking graph $\mathcal{N}^* = (S, E, T, s_0)$ - thus $S \subseteq 2^P$. If $x \in P$ then its extension $\|x\| = \{M \in S / x \in M\}$ is a region of $\mathcal{N}^*$.*

A set $X \subseteq S$ is a region if and only if its characteristic function $\sigma = \chi_X : S \to \{0, 1\}$ admits a (unique) companion map $\eta : E \to \{-1, 0, 1\}$ such that $\sigma(s') = \sigma(s) + \eta(e)$ for every transition $s \stackrel{e}{\to} s'$ in T. From now on, we identify regions with such pairs of mappings (σ, η) which turn to be exactly the morphisms of transitions systems [2] from $T = (S, E, T)$ to the *classifying* transition system $\mathbf{2} = (\{0, 1\}, E_2, T_2)$ shown in Fig. 2.1, with $E_2 = \{-1, 0, 1\}$ and $T_2 = \{0 \stackrel{0}{\to} 0,\ 0 \stackrel{1}{\to} 1,\ 1 \stackrel{-1}{\to} 0,\ 1 \stackrel{0}{\to} 1\}$. A region $X \equiv (\sigma, \eta)$ determines an *atomic* elementary net $N_X = (\{X\}, E, F_X)$ with flow relation F_X set according to the mapping η, namely:

$$X \in {}^\bullet e \ \textbf{iff} \ \eta(e) = -1 \quad \textit{and} \quad X \in e^\bullet \ \textbf{iff} \ \eta(e) = 1$$

[2] A morphism of transition systems $(\sigma, \eta) : (S_1, E_1, T_1) \to (S_2, E_2, T_2)$ is a pair of mappings $\sigma : S_1 \to S_2$ and $\eta : E_1 \to E_2$ such that $s \stackrel{e}{\to} s' \in T_1 \Rightarrow \sigma s \stackrel{\eta e}{\to} \sigma s' \in T_2$



$X \equiv (\sigma, \eta) : T \to \mathbf{2}$

Figure 2.1: regions as morphisms

If $X = \|x\|_P$ is the extension of a place x of a net $N = (P, E, F)$ then N_X is just the atomic subnet of N induced by x.

Given a transition system $T = (S, E, T)$, an elementary net T^* may now be *synthesized* from the set $\mathcal{R}_T$ of regions of T by amalgamating on E all the atomic nets N_X for X ranging over $\mathcal{R}_T$. Thus, the net synthesized from$T$ is $T^* = \sum_{X \in \mathcal{R}_T} N_X$. Every region X of T expresses an *elementary synchronic constraint* on event behaviours, satisfied in all possible runs of T. For instance, the region depicted in Fig. 2.1 expresses the constraint that two successive occurrences of event b should be separated by one occurrence of event a and vice versa. If we supply T with an initial state $s_0 \in S$, then the elementary net N (and each of its atomic components N_X) comes equipped with an initial marking $M_0 \subseteq 2^{\mathcal{R}_T}$, containing exactly those regions which the initial state belongs to: $X \in M_0$ if and only if $s_0 \in X$. $A^* = (T^*, M_0)$ is the elementary net system *synthesized* from the automaton $A = (T, s_0)$. Suppose for instance that the initial state does not belong to the region represented in Fig. 2.1; then the unique place of the induced atomic net N_X is not marked initially, and the language of the atomic net system $\mathcal{N}_X$ is the shuffle of $(a \cdot b)^*$ and c^*. Since the regions of T are the morphisms $(\sigma, \eta) : T \to \mathbf{2}$, the language of A is included in the language of every atomic net system $\mathcal{N}_X$ induced from $X \in \mathcal{R}_T$ and thus it is included in their intersection i.e. in the language of A^*. Adding up synchronic constraints imposed by regions amounts in fact to intersecting behaviours. The case graphs of elementary net systems are precisely those automata whose behaviour may be totally specified in terms of elementary synchronic constraints (see Theo. 2.3.3 below).

2.3 Elementary Transition Systems

This section reports on Ehrenfeucht and Rozenberg's characterization of isomorphism classes of case graphs of elementary net systems as *elementary* transition systems, i.e. transition systems owning *admissible* sets of regions as defined by

Desel and Reisig.

We let the set of regions of an automaton $A = (S, E, T, s_0)$ be the set of regions of its underlying transition system T: $\mathcal{R}_A = \mathcal{R}_T$. Let A be isomorphic to the case graph of some elementary net system $\mathcal{N}= (P, E, F, M_0)$. If s_1 and s_2 are two distinct states of S, viewed as markings of $\mathcal{N}$, then there exists a place $x \in P$ which belongs to exactly one of these markings; hence there exists a region (the extension of the place x) that distinguishes between s_1 and s_2. If an event e has not concession at a given state s, viewed as a marking $M = [s]_S$, then either ${}^\bullet e \not\subseteq M$ or $e^\bullet \cap M \neq \emptyset$; hence there exists a region $X \equiv (\sigma, \eta)$ such that $\sigma(s) = 0$ and $\eta(e) = -1$ (in the first case, X is the extension of a place $x \in {}^\bullet e \setminus M$, and in the second case it is the complement of the extension of a place $x' \in e^\bullet \cap M$). Let $R_s = \{X \in \mathcal{R}_A \mid s \in X\}$ denote the set of (non trivial) regions of A containing state s, and let now ${}^\bullet e$ denote the set of regions $X \equiv (\sigma, \eta)$ of A such that $\eta(e) = -1$, then:

Observation 2.3.1 *The following two* separation axioms *are satisfied in every case graph A:*
–SSP– *(States Separation Property):*

$$\forall s, s' \in S \quad s \neq s' \Rightarrow [\exists R \in \mathcal{R}_A \quad (s \in R \Leftrightarrow s' \notin R)]$$

–ESSP– *(Events-States Separation Property):*

$$\forall e \in E \quad \forall s \in S \quad not(s \xrightarrow{e}) \quad \Rightarrow [\exists R \in \mathcal{R}_{\mathcal{A}}\ (R^\bullet e\ \wedge\ s \notin R)\ \vee\ (e^\bullet R\ \wedge\ s \in R)]$$

Definition 2.3.2 (Elementary Transition System) *An elementary transition system is an automaton that fulfils the four conditions stated in Obs. 2.1.4 plus the above two separation axioms.*

Theorem 2.3.3 *[43] An automaton is the case graph of an elementary net system if and only if it is an elementary transition system, and in that case $A \cong A^{**}$.*

Notice that the complement $\overline{X} = S \setminus X$ of a region X is a region with dual flow relations: $X \in {}^\bullet e \Leftrightarrow \overline{X} \in e^\bullet$ and $X \in e^\bullet \Leftrightarrow \overline{X} \in {}^\bullet e$. Even though the marking graph of an elementary net system $\mathcal{N}$ is always isomorphic to the marking graph of $\mathcal{N}^{**}$, $\mathcal{N}$ is generally embedded in $\mathcal{N}^{**}$ as a strict subnet with fewer conditions (the embedding maps a condition x to the region X of the marking graph formed of all markings containing that condition).

Definition 2.3.4 *An elementary net system $\mathcal{N}$ is said to be* saturated *if $\mathcal{N} \equiv \mathcal{N}^{**}$, or equivalently, if $\mathcal{N} \equiv \mathcal{A}^*$ for some elementary transition system $\mathcal{A}$.*

Example 2.3.5 *Consider the elementary net system shown in Fig. 2.2 together with its case graph. Some (non trivial) regions of that elementary transition system are shown in Fig. 2.3. The missing items can be obtained by symmetry. In each drawing, the grey states form a region, say X, the flow arcs attached to this region X and to the complementary region $\overline{X}$ are indicated, and one token is put in the place X or $\overline{X}$ that contains the initial state. Altogether, one ends up*

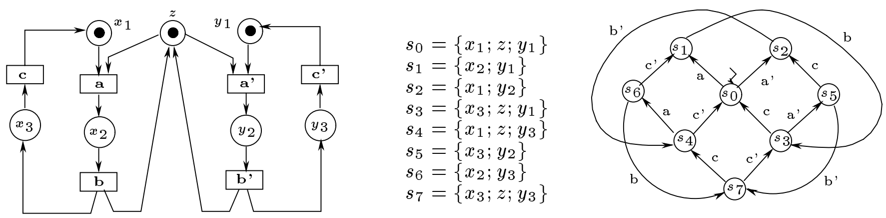


Figure 2.2: an elementary net system for mutual exclusion and its case graph

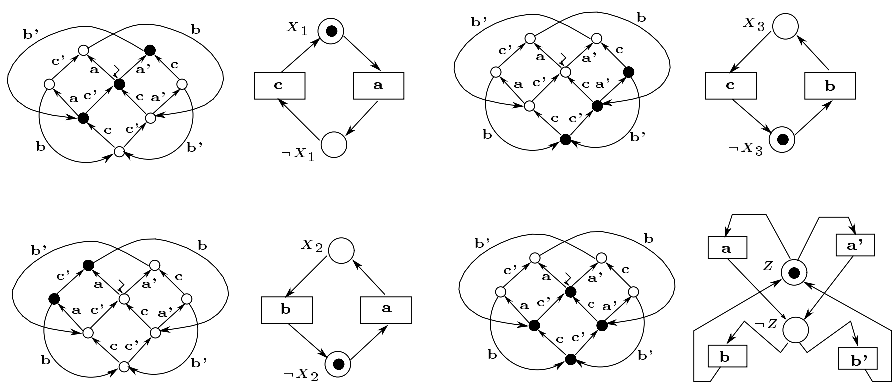


Figure 2.3: some regions of the elementary transition system of Fig. 2.2

with the elementary net system shown in Fig. 2.4. This net $\mathcal{N}^{**}$ is the saturated version of the net $\mathcal{N}$ shown in Fig. 2.2 and their respective marking graphs are both isomorphic to the graph $\mathcal{N}^{*}$ shown in Fig. 2.2. Observe the embedding of $\mathcal{N}$ into $\mathcal{N}^{**}$ via the function $x \to [\![x]\!]$ that maps places to their extensions. The set of places of $\mathcal{N}^{**}$ (or regions of the marking graph $\mathcal{N}^{*}$) may be enumerated as follows:

X	$=$	$[\![x]\!]$	$=$	$\{s_0; s_3; s_4; s_7\}$	$\overline{X}$	$=$	$\{s_1; s_2; s_5; s_6\}$
X_1	$=$	$[\![x_1]\!]$	$=$	$\{s_0; s_2; s_4\}$	$\overline{X_1}$	$=$	$\{s_1; s_3; s_5; s_6; s_7\}$
X_2	$=$	$[\![x_2]\!]$	$=$	$\{s_1; s_6\}$	$\overline{X_2}$	$=$	$\{s_0; s_2; s_3; s_4; s_5; s_7\}$
X_3	$=$	$[\![x_3]\!]$	$=$	$\{s_3; s_5; s_7\}$	$\overline{X_3}$	$=$	$\{s_0; s_1; s_2; s_4; s_6\}$
Y_1	$=$	$[\![y_1]\!]$	$=$	$\{s_0; s_1; s_3\}$	$\overline{Y_1}$	$=$	$\{s_2; s_4; s_5; s_6; s_7\}$
Y_2	$=$	$[\![y_2]\!]$	$=$	$\{s_2; s_5\}$	$\overline{Y_2}$	$=$	$\{s_0; s_1; s_3; s_4; s_6; s_7\}$
Y_3	$=$	$[\![y_3]\!]$	$=$	$\{s_4; s_6; s_7\}$	$\overline{Y_3}$	$=$	$\{s_0; s_1; s_2; s_3; s_5\}$

2.3.1 Admissible sets of regions

Given an automaton A, a subset of regions $R \subseteq \mathcal{R}_A$ is said to be *admissible* if $\mathcal{A} \equiv \mathcal{N}^{*}$ for $\mathcal{N}$ defined as $\sum_{X \in R} \mathcal{N}_X$.

Proposition 2.3.6 *Any set of regions that contains an admissible set of regions is admissible.*

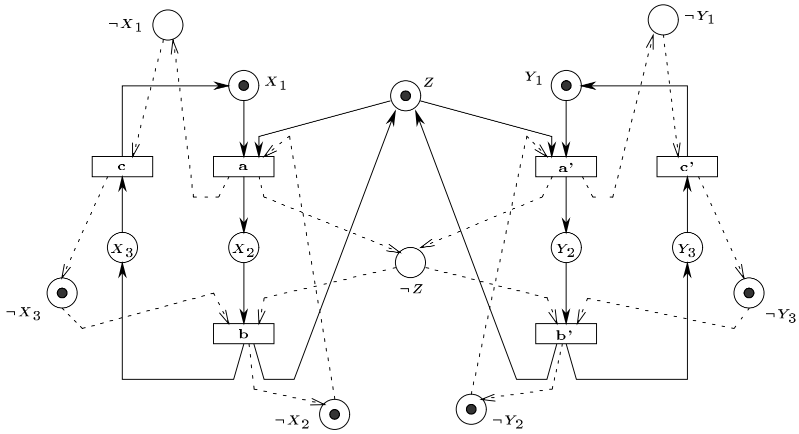


Figure 2.4: the elementary net system synthesized from the elementary transition system of Fig. 2.2

Therefore, as soon as an admissible set of regions has been computed in an elementary transition system, adding new regions as places in the resulting net won't modify the behaviour of that net. One aspect of the synthesis problem is to find admissible sets of regions as small as possible. In order to construct an admissible set of regions in A, it suffices to collect witnesses for the satisfaction of all the instances in A of the two axioms of separation which have been stated in Obs. 2.3.1.

Proposition 2.3.7 *[37] Given $A = (S, E, T, s_0)$, $s \in S$, and $R' \subseteq \mathcal{R}_A$, let $R'_s = \{X \in R' \mid s \in X\}$ i.e. $R'_s = R_s \cap R'$, then R' is an admissible set of regions of A if and only if the following are satisfied:*

1. $\forall s, s' \in S \quad R'_s = R'_{s'} \Rightarrow s = s'$,
2. $\forall s \in S \ \ \forall e \in E \quad ({}^{\bullet}e \subset R'_s \ \wedge \ e^{\bullet} \cap R'_s = \emptyset) \Rightarrow s \stackrel{e}{\rightarrow}$

In order to decide whether a finite automaton $A = (S, E, T, s_0)$ satisfying the conditions stated in Obs. 2.1.4 is elementary, it is therefore enough to compute at most $|S| \times (|S| \times |E|)$ regions in A.

Corollary 2.3.8 *If $\mathcal{A} = (S, E, T, s_0)$ is an elementary transition system, there must exist an elementary net system $\mathcal{N}$ with at most $|S| \times (|S| \times |E|)$ places such that $\mathcal{N}^* \equiv A$.*

Nevertheless, this does not indicate how to select these regions from $\mathcal{R}_A$. It is worth noting that there exists in general no least admissible set of regions. This fact is illustrated in Fig. 2.5 by the so-called "four seasons" example reproduced from [37]. The "four seasons" automaton may be realized by two minimal subnet systems of the dual saturated net system: one has four conditions and is contact-free while the other one has three conditions but is not contact-free.

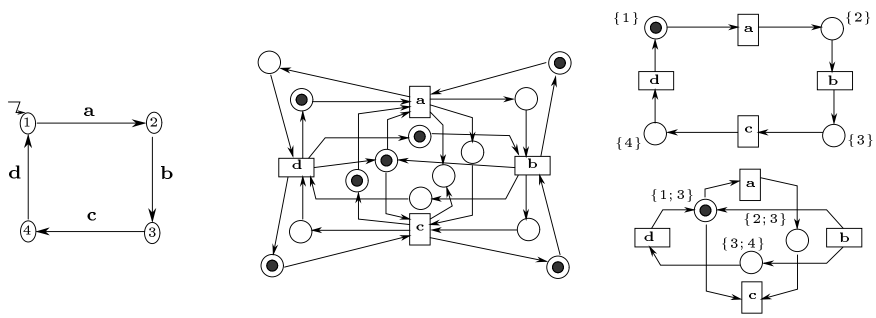


Figure 2.5: the four seasons example: the automaton (on the left), the saturated net system (on the middle) and two elementary net systems corresponding to minimal sets of regions (on the right)

Definition 2.3.9 *An elementary net system $\mathcal{N} = (P, E, F, M_0)$ is contact-free if ${}^\bullet e \subset M \Rightarrow M \cap e^\bullet = \emptyset$ for every event e and for every reachable case M.*

It is an open question whether there exists a contact-free elementary net system with a minimal number of places associated with any elementary transition system.

The following adaptation of Theo. 2.3.3, based on the use of complementary regions, is established in [37]

Proposition 2.3.10 *An automaton $A = (S, E, T, s_0)$ is isomorphic to $\mathcal{N}^*$ for a contact-free net system $\mathcal{N} = (P, E, F, M_0) = \sum_{p \in P} \mathcal{N}_p$ if and only if every atomic subnet system $\mathcal{N}_p$ of $\mathcal{N}$ may be defined from a corresponding region $R_p \in \mathcal{R}_A$ and the following properties of separation are satisfied:*
SSP : $\forall s, s' \in S \quad s \neq s' \Rightarrow [\exists p \in P \quad s \in R_p \Leftrightarrow s' \notin R_p]$
ESSP$^\sharp$: $\forall e \in E \quad \forall s \in S \quad not(s \xrightarrow{e}) \Rightarrow [\exists p \in P \quad R_p{}^\bullet e \wedge s \notin R_p]$.

Bernardinello proved in [19] that the set of minimal regions (w.r.t. set inclusion) of an elementary transition system is admissible and the corresponding elementary net system is contact free. He also showed that state machine components of an elementary net system are in bijective correspondence with the partitions of its case graph by minimal regions. In [20] it is also shown that the set of regions of an elementary net system ordered by inclusion is an orthomodular poset whose boolean sub-algebras are given by the set of regions of its sequential components.

2.4 The Synthesis of Elementary Net Systems is NP-Complete

Hiraishi proved in [52] that the separation problems SSP(s, s') and ESSP$^\sharp(s, e)$ are NP-complete in the respective data (A, s, s') and (A, s, e). Since regions in A are closed under complementation, the problem $\text{ESSP}(s, e)$ is also NP-complete. It does not follow therefrom that the synthesis problem for elementary net systems is NP-complete; however this is the case. The synthesis problem is obviously in NP since the total number of instances of separation problems in an automaton A is quadratic in the size of A, and it can be checked in polynomial time whether a non-deterministically chosen subset of states is a region solving a fixed separation problem. Now a polynomial reduction of 3-SAT to the synthesis problem of elementary net systems is established in [7], showing NP-hardness since 3-SAT is NP-complete (see e.g. [45]). The proof relies on the observation that one can encode the separations problems into a system of additive or multiplicative clauses over the boolean ring. The proof then consists in two stages. First we show that 3-SAT reduces to the satisfiability problem for such systems of clauses. Second we show that each clausal system of this form is associated with an automaton, with size polynomial in the size of the system of clauses, such that the system is satisfiable if and only if the automaton is elementary.

We won't describe here these two stages (see [7] for details). We only describe how the synthesis problem for elementary net systems can be encoded to systems of polynomial equations over the boolean ring. A similar algebraic setting is used in the Chapter 4 for the polynomial time synthesis of vector addition systems, with the main difference that all the equations used there are linear.

In view of the above, solving the synthesis problem for elementary net systems amounts to solving a quadratic (in the size of the automaton) number of instances of the following two separation problems.

States Separation Problem (SSP):
Given $T = (S, E, T)$ and a pair of distinct states $(s_1, s_2) \in S \times S$, find a region X such that $s_1 \in X$ if and only if $s_2 \notin X$.

Event/State Separation Problem (ESSP):
Given $T = (S, E, T)$ and a pair $(s, e) \in S \times E$ such that e *has no concession* at s ($s \xrightarrow{e} s'$ in T for no $s' \in S$), find a region X *inhibiting* e at s in the sense that $X \in {}^\bullet e$ and $s \notin X$.

Let us now proceed to the encoding of the separation problems into equations over the boolean ring $\mathbb{Z}/2\mathbb{Z}$.

Definition 2.4.1 *The second projections $\eta : E \to \{-1, 0, 1\}$ of regions (σ, η) are called* signed regions, *and the maps $\rho : E \to \{0, 1\}$ defined from signed regions by $\rho(e) = |\eta(e)|$ are called* abstract regions.

Given a region (σ, η), the abstract region $\rho = |\eta|$ indicates which transition labels entail a change of the current value for σ:

$$s \xrightarrow{e} s' \quad \in T \quad \Rightarrow \quad [\sigma(s) \neq \sigma(s') \Leftrightarrow \rho(e) = 1] \tag{2.1}$$

The abstract regions of a transition system (S, E, T) are elements of the $\mathbb{Z}/2\mathbb{Z}$-module $\mathbb{Z}/2\mathbb{Z}[E]$ which consists of the set of mappings from E to $\mathbb{Z}/2\mathbb{Z}$.

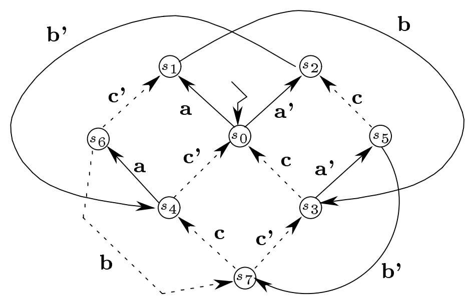


Figure 2.6: a graph and one of its maximal trees

Observation 2.4.2 *In a connected transition system, an abstract region determines always two complementary regions.*

In view of this remark, the main step towards a translation of the separation axioms to equations is to produce an algebraic characterization for the abstract regions of a given transition system $T = (S, E, T)$. We let he *Parikh image modulo 2* of a chain $c = \sum c_j \cdot t_j$ be the vector $\psi_2(c) = \sum c_j \cdot \ell(t_j)$ evaluated as an element of the $\mathbb{Z}/2\mathbb{Z}$ free module $\mathbb{Z}/2\mathbb{Z}[E]$.

Example 2.4.3 *Figure 2.6 displays a graph and a maximal tree in that graph. There are 7 transitions which are not part of the maximal tree:* $t_1 = s_5 \xrightarrow{c} s_2$, $t_2 = s_3 \xrightarrow{c} s_0$, $t_3 = s_7 \xrightarrow{c} s_4$ $t_4 = s_6 \xrightarrow{c'} s_1$, $t_5 = s_4 \xrightarrow{c'} s_0$ $t_6 = s_7 \xrightarrow{c'} s_3$, *and* $t_7 = s_6 \xrightarrow{b} s_7$. *For instance, the transition* $t_1 = s_5 \xrightarrow{c} s_2$ *determines the cycle*

$$c^{t_1} = (s_0 \xrightarrow{a} s_1) + (s_1 \xrightarrow{b} s_3) + (s_3 \xrightarrow{a'} s_5) + (s_5 \xrightarrow{c} s_2) - (s_0 \xrightarrow{a'} s_2)$$

with Parikh image $\psi_2(c^{t_1}) = a + b + c$. *One can verify that* $\psi_2(c^{t_1}) = \psi_2(c^{t_2}) = \psi_2(c^{t_3}) = a + b + c$, $\psi_2(c^{t_4}) = \psi_2(c^{t_5}) = \psi_2(c^{t_6}) = a' + b' + c'$, *and* $\psi_2(c^{t_7}) = 0$.

Let us now state the essential property of abstract regions:

Proposition 2.4.4 *If* $\rho : E \to \mathbb{Z}/2\mathbb{Z}$ *is an abstract region of the transition system* (S, E, T) *then* $\rho \cdot \psi_2(c) = 0$ *in* $\mathbb{Z}/2\mathbb{Z}$ *for every cycle* c *in the graph* $[S, T]$.

This fundamental property taken alone does not provide a full characterization for abstract regions. In the sequel, we consider a fixed automaton $A = (S, E, T, s_0)$ and a maximal tree $U \subseteq T$ rooted at s_0. For each $s \in S$, we let c_s be the branch from s_0 to s in that tree, and we let $\psi_s = \psi_2(c_s)$ denote its Parikh image modulo 2. Now suppose that $|\eta(e)| = 1$ for some region (σ, η) such that e has concession at states s and s', then necessarily $\sigma(s) = \sigma(s')$ by definition of regions. In $\mathbb{Z}/2\mathbb{Z}$, this may be expressed by the non linear equation

$$\rho(e) \times [\rho \cdot (\psi_s + \psi_{s'})] = 0$$

where $\rho(e') = |\eta(e')|$ for all $e' \in E$. We obtain in this way a full characterization for abstract regions:

Table 2.1: regions X_ρ defined from abstract regions ρ

$\rho \cdot \psi_s$	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7
ρ_1	0	1	0	1	0	1	1	1
ρ_2	0	0	0	1	0	1	0	1
ρ_3	0	0	1	0	1	1	1	1
ρ_4	0	0	0	0	1	0	1	1
$\rho_1 + \rho_2$	0	1	0	0	0	0	1	0
$\rho_3 + \rho_4$	0	0	1	0	0	1	0	0
$\rho_1 + \rho_2 + \rho_3 + \rho_4$	0	1	1	0	0	1	1	0

Proposition 2.4.5 *A map $\rho : E \to \mathbb{Z}/2\mathbb{Z}$ is an abstract region if and only if it satisfies*

$$\rho \cdot \psi_2(c^t) = 0 \tag{2.2}$$

$$\rho(e) \times [\rho \cdot (\psi_s + \psi_{s'})] = 0 \tag{2.3}$$

for every cycle c^t in the basis of cycles, for every event $e \in E$, and for every pair of states s and s' in S at which e has concession.

Example 2.4.6 (continued) *In our running example,the Parikh images of the cycles are $a+b+c$, $a'+b'+c'$ and 0, hence the equations of type (2.2) are*

$$\rho(a) + \rho(b) + \rho(c) = 0 \quad \textit{and} \quad \rho(a') + \rho(b') + \rho(c') = 0 \tag{2.4}$$

which determine altogether a four dimensional $\mathbb{Z}/2\mathbb{Z}$-module with basis as follows:

$$\rho_1 = a + c \quad ; \quad \rho_2 = b + c \quad ; \quad \rho_3 = a' + c' \quad ; \quad \rho_4 = b' + c'$$

A linear combination ρ of these vectors is an abstract region of the graph shown in Fig. 2.6 if and only if it satisfies the non linear equations of type (2.3) which modulo the equations (2.4) read:

$$\rho(a) \times \rho(c') = \rho(b) \times \rho(c') = \rho(c) \times \rho(c') = \rho(c) \times \rho(a') = \rho(c) \times \rho(b') = 0$$

Seven out of the 2^4 elements of the $\mathbb{Z}/2\mathbb{Z}$-module spanned by $(\rho_1, \rho_2, \rho_3, \rho_4)$ are abstract regions (they are indicated as the row entries of Table 2.1). Each abstract region ρ determines two complementary regions X_ρ and $\overline{X_\rho}$, where $X_\rho = \{s \in S | \ \rho \cdot \psi_s = 1\}$ and therefore $s_0 \notin X_\rho$. The states s_0 to s_7 are represented by vectors π_s as follows:

$$\begin{array}{llll} \psi_{s_0} = 0 & \psi_{s_1} = a & \psi_{s_2} = a' & \psi_{s_3} = a + b \\ \psi_{s_4} = a' + b' & \psi_{s_5} = a + b + a' & \psi_{s_6} = a' + b' + a & \psi_{s_7} = a + b + a' + b' \end{array}$$

The resulting regions X_ρ, tabulated in Table 2.1, are half of the 14 regions of Fig. 2.4; the correspondence is made explicit in Fig. 2.7.

Let us finally indicate the equational encoding for the instances of the separation problems SSP and ESSP.

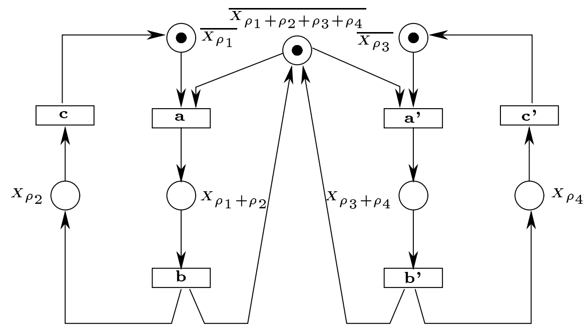


Figure 2.7: the net synthesized from the (admissible) set of abstract regions

Definition 2.4.7 *An abstract region ρ separates the states s and s' if it satisfies the equation*

$$\rho \cdot (\psi_s + \psi_{s'}) = 1 \tag{2.5}$$

An abstract region ρ inhibits event e at state s if it satisfies the equations

$$\rho(e) = 1 \tag{2.6}$$

$$\rho \cdot (\psi_s + \psi_{s'}) = 1 \tag{2.7}$$

where s' is some arbitrary state enabling e. A set R of abstract regions is admissible *if it provides a separating region for every pair of distinct states (s, s') and an inhibiting region for every pair (s, e) such that event e has no concession at state s.*

An admissible set of regions may always be constructed from an admissible set of abstract regions: if ρ separates states s and s' then both X_ρ and $\overline{X_\rho}$ separate s and s'; if ρ inhibits event e at state s then either X_ρ or $\overline{X_\rho}$ inhibits e at s (in our running example, the full set of abstract regions is admissible, but no strict subset of that set is admissible).

Proposition 2.4.8 *An automaton is the case graph of an elementary net system if and only if it may be fitted with an admissible set of abstract regions.*

Definition 2.4.9 (Systems of Clauses over the Boolean Ring) *Let $X = \{x_0, \ldots, x_n\}$ be a finite set of boolean variables, with a distinguished element x_0. A* system of clauses over the boolean ring *is a pair (Σ, Π) where Σ is a finite set of additive clauses σ_α ($\alpha \in A$) and Π is a finite set of multiplicative clauses π_β ($\beta \in B$) with respective forms $x_{\alpha_0} + x_{\alpha_1} + x_{\alpha_2}$ and $x_{\beta_1} \cdot x_{\beta_2}$, subject to the following restrictions: each additive clause has exactly three variables, two additive clauses have at most one common variable, each multiplicative clause has exactly two variables, and the distinguished variable x_0 does not occur in any multiplicative clause. The system (Σ, Π) is said to be* satisfiable *if there exists a truth assignment for X such that $x_0 = 1$, $\sigma_\alpha = 0$ for all $\alpha \in A$, and $\pi_\beta = 0$ for all $\beta \in B$. Such a truth assignment is called a solution of (Σ, Π).*

Observe that in the boolean ring, the equations $z_0 = z_1 + \ldots + z_n$ and $z_0 + z_1 + \ldots + z_n = 0$ are equivalent in view of the inversion law $z + z = 0$. Using this remark, each instance of problems SSP and ESSP may be reduced to the satisfiability of a corresponding system of clauses, with size polynomial in the size of the transition system.

Let CBR denote the satisfiability problem for systems of clauses over the boolean ring. In [7] it is shown that CBR, 3-SAT and the synthesis problem for elementary net systems are polynomially inter-reducible. Thus the synthesis problem for elementary net systems is NP-complete.

Chapter 3

Representations of Reversible Automata

States graphs of various kind of nets are reversible automata which means that events induce local bijections on the set of states. This property greatly simplifies the study of the algebraic properties of these automata as notions borrowed from the litterature on combinatorial group theory, see [49, 64, 63, 98], like the notions of coverings, group acting on graph, and homology are well investigated and more easy to manipulate than their counterparts used in the more general context of transformation monoids. In the first section of this chapter we give a review of this theory a variant of which is known as the study of *inverse monoids* (see [78, 97]). In the second part of this chapter we give a classification of the representations of reversible automata as full subgraphs of Schreier graphs (also called coset graphs) of groups. The condition that should be fufilled by the representing groups are very similar to the separation properties discovered by Ehrenfeucht and Rozenberg in their study of elementary transition systems. We also describe the computation of the canonical representation of a commutative automaton (automaton that fully embedds in the Cayley graph of an abelian group).

3.1 Reversible Automata

A transition system (S, E, T) is said to be *deterministic* (respectively *co-deterministic*) if $s \xrightarrow{e} s_1 \wedge s \xrightarrow{e} s_2 \Rightarrow s_1 = s_2$ (resp. if $s_1 \xrightarrow{e} s \wedge s_2 \xrightarrow{e} s \Rightarrow s_1 = s_2$).

Definition 3.1.1 *A reversible transition system is a deterministic and co-deterministic transition system.*

3.1.1 Permutation Transition Systems

A permutation transition system is a transition system in which each event induces a permutation on the set of states. It is therefore a *complete* reversible transition system. If G is a group, H a subgroup of G, and E a subset of G (usually a set of generators), the Schreier graph $S(G,H,E)$ is the permutation transition system whose states are the right-cosets $H\backslash G = \{Hg | g \in G\}$ and whose transitions are the triples (Hg, e, Hge). The Cayley graph $C(G,E)$ is $S(G,1,E)$ where 1 is the trivial subgroup; more generally if H is a normal subgroup of G and if not two distinct generators in E are equivalent modulo H, then the Schreier graph $S(G,H,E)$ is isomorphic to the Cayley graph $C(G/H, E/H)$. Any permutation transition system (S,E,T,s_0) whose underlying graph is connected is isomorphic to the Schreier graph $S(G,H,E)$ where G is the group generated by the permutations in E and $H = G_{s_0} = \{g \in G | s_0 \cdot g = s_0\}$ is the stabiliser of the initial state. Indeed each state $s \in S$ can be univocally encoded by the set $G_{s_0,s} = \{g \in G \mid s_0 * g = s\}$ which is a H right-coset and this correspondance is an isomorphism of transition systems. The stabilisers of the states of a connected permutation transition system are conjugate (if $s' = s * u$ then $G_{s'} = u^{-1} G_s u$), and the map $S(G,G_s,E) \tilde{\to} S(G,G_{s'},E) \ : \ G_s v \mapsto G_{s'} u^{-1} v$ is an isomorphism between the respective Schreier graphs. Conversely $S(G,H,E) \cong S(G,K,E)$ implies that the groups H and K are conjugate in G. [1]

3.1.2 Reversible Automata

Let $\overline{E}$ be an isomorphic copy of E consisting of formal inverses $\overline{e}$ of events $e \in E$. We recall that the free group generated by E is the group with presentation $F(E) = \mathbf{gp}(E \cup \overline{E}; e \cdot \overline{e}, \overline{e} \cdot e (e \in E))$. A word in $F(E)$ is a word $u \in (E \cup \overline{E})^*$; in order to ease notation we shall usually make no distinction between a word in $F(E)$ and the actual element of $F(E)$ that it represents. A word in $F(E)$ is termed *reduced* if it contains no subwords of the form $e \cdot \overline{e}$ or $\overline{e} \cdot e$ for some $e \in E$. A reduced word is therefore the canonical form of an element of $F(E)$. The free group $F(E)$ has a partially defined action on the set of states of a reversible transition system (S,E,T) given by letting $s * e = s'$ and $s' * \overline{e} = s$ when $s \xrightarrow{e} s'$. A is said to be *connected* when $F(E)$ acts transitively on S: $\forall s, s' \in S \ \exists u \in F(E) \quad s * u = s'$.

Definition 3.1.2 *A reversible automaton is a connected reversible transition system together with an initial state.*

3.1.3 Fundamental Group and Coverings

The *fundamental group* of transition system T in state $s \in S$ is $\pi_1(T,s) = \{u \in F(E) \mid s * u = s\}$. An element of $\pi_1(T,s)$ is called a *closed path* based on s. If the automaton is connected all fundamental groups are conjugate in $F(E)$ hence

[1] In fact the Schreier graph $S(G,H,E)$ is a graphical representation of the coset space $\cos(G{:}H)$, i.e. the set of right cosets $H\backslash G$ on which G acts by right mutiplication and the results mentioned above are the graph theoretic counterparts of the fact that any *transitive* G-space is isomorphic to a coset space $\cos(G{:}H)$ for some subgroup H of G and that one has therefrom a bijective correspondance between the isomorphic classes of transitive G-spaces and the conjugacy class of subgroups of G.

isomorphic: $\pi_1(T,s') = u^{-1}\cdot\pi_1(T,s)\cdot u$ when $s*u = s'$. We let $\pi_1(A) = \pi_1(T,s_0)$ denote the fundamental group of reversible automaton $A = (S,E,T,s_0)$. Let $U \subseteq T$ be some spanning tree of the automaton A. Let u_s be the (reduced) word labelling the (unique) path in U from the initial state s_0 to state s. Each chord $t \in T \setminus U$ where $t = s \xrightarrow{e} s'$ determines a closed path $c_t = u_s \cdot e \cdot u_{s'}^{-1}$ based on s_0. The fundamental group $\pi_1(A)$ is the group freely generated by the closed paths c_t associated with the chords of some spanning tree of A.

There is a bijective correspondance between the subgroups H of $F(E)$ and the right congruences of $F(E)$ given by $u \equiv v \Leftrightarrow u \cdot v^{-1} \in H$, and $\equiv$ is a congruence if and only if H is a normal subgroup of $F(E)$. We say that a subgroup H of $F(E)$ saturates a language $L \subseteq F(E)$ when its associated right congruence does, i.e. when L is a union of right cosets of H. The *language* of a reversible automaton $A = (S,E,T,s_0)$, let $L(A) = \{u \in F(E) \mid \exists s \in S \quad s_0 * u = s\}$ is saturated by its fundamental group, indeed $L(A) = \bigcup_{s\in S} \pi_1(A)u_s$ where u_s is the (reduced) word labelling the path in some fixed spanning tree of A from s_0 to s. Conversely, if H is a subgroup of $F(E)$ and $L \subseteq F(E)$ a prefix-closed language saturated by H, we let $S(L,H,E)$ be the subautomaton of the Schreier graph $S(F(E),H,E)$ induced by the vertices (H right cosets) included in L; i.e. $S(L,H,E) = (E,S,T,s_0)$ where $S = \{Hu \mid u \in F(E) \quad Hu \subseteq L\}$ and $T = \{(Hu,e,Hue) \mid u \in F(E) \quad e \in E \quad Hu \subseteq L \quad Hue \subseteq L\}$, and $s_0 = H$. Notice that $\pi_1(S(L,H,E)) = H$ and $L(S(L,H,E)) = L$ and $A \cong S(L(A),\pi_1(A),E)$. We have therefrom bijective correspondances between the set of right congruences saturating a prefix-closed language $L \subseteq F(E)$, the set of subgroups of $F(E)$ saturating L and the set of isomorphic classes of reversible automata recognizing L.

Dual to the inclusion of their fundamental groups is the covering of reversible automata: a *covering* of a reversible automaton A is another reversible automaton $\tilde{A}$ together with a map $f : \tilde{S} \to S$ between their respective set of states such that $\forall s \in S \quad \forall \tilde{s} \in \tilde{S} \quad f(\tilde{s}) = s \Rightarrow (\forall u \in F(E) \quad s * u \Leftrightarrow \tilde{s} * u)$. Therefore if $\tilde{A}$ and A are reversible automata there exists a covering from $\tilde{A}$ to A if and only if they have the same language and $\pi_1(\tilde{A}) \subseteq \pi_1(A)$ and such a covering is *uniquely* determined by $f(\tilde{s}_0 * u) = s_0 * u$. We write $A \leq \tilde{A}$ when this happens and said that $\tilde{A}$ *covers* A. We deduce that the set of isomorphic classes of reversible automata recognizing $L \subseteq F(E)$ ordered by the covering relation is dually-isomorphic to the set of subgroups of $F(E)$ saturating L and ordered by inclusion.

Notice that if $\pi_1(\tilde{A}) \subseteq \pi_1(A)$ then $\pi_1(\tilde{A})$ saturates every language saturated by $\pi_1(A)$ and thus the coverings $\tilde{A}$ of A can be classified up to isomorphism by the subgroups of $\pi_1(A)$; in particular there is a maximal covering corresponding to the trivial subgroup. Symmetrically the *quotients* of A correspond up to isomorphism to the groups H saturating the language of A and including its fundamental group.

A covering $f : \tilde{A} \to A$ is termed a *Galois covering* if $\pi_1(\tilde{A})$ is a normal subgroup of $\pi_1(A)$. The *Galois group* of the Galois covering f is the quotient group $\pi_1(A)/\pi_1(\tilde{A})$. A *path* in a reversible automaton $A = (S,E,T,s_0)$ is a pair $(s,u) \in S \times F(E)$ such that $s * u$ is defined, it is a *closed path* based on s, noted

$(s,u) \in C(s)$, if moreover $s * u = s$ (thus $\pi_1(T,s) = \{u \in F(E) \mid (s,u) \in C(s)\}$). Two closed paths $(\tilde{s},u)$ and $(\tilde{s}',u')$ of $\tilde{A}$ are *conjugate* in the covering $f : \tilde{A} \to A$ if they have the same f-image, i.e. $f(\tilde{s}) = f(\tilde{s}')$ and $u = u'$. Then a covering $f : \tilde{A} \to A$ between reversible automata is a Galois covering if and only if every path of $\tilde{A}$ conjugate to a closed path is closed.

The *Nerode equivalence* associated with a language $L \subseteq F(E)$ is $u \equiv v \Leftrightarrow [\forall w \in F(E) \quad u \cdot w \in L \Leftrightarrow v \cdot w \in L]$, i.e. $u \equiv v$ if and only if $u^{-1}L = v^{-1}L$. It is the greatest right congruence saturating L, i.e. the Nerode group $N_L = \{u \in F(E) \mid u^{-1}L = L\}$ is the greatest subgroup of $F(E)$ saturating L.

3.2 Representations of Reversible Automata

3.2.1 Extensions of Reversible Automata

Definition 3.2.1 *An* extension *of a reversible automaton $A = (S,E,T,s_0)$ is any complete reversible automaton (i.e. permutation automaton) isomorphic to some $A' = (E,S',T',s_0)$ such that $S \subseteq S'$ and $T = T' \cap (S \times E \times S)$. The inclusion of S into S' is termed a* full embedding *of A into A'.*

If $U \subseteq T$ is some spanning tree of a reversible automaton A, we let $\Delta(A,U)$ denote the following subset of $F(E)$:

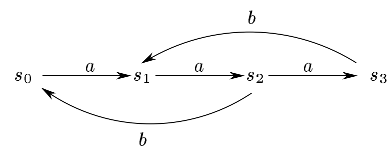
$$\Delta(A,U) = \{u_s \cdot u_{s'}^{-1} \mid \forall s,s' \in S \quad s \neq s'\} \cup \{u_s \cdot e \cdot u_{s'}^{-1} \mid \forall s,s' \in S \quad s \stackrel{e}{\not\to} s'\}$$

The word $u_s \cdot u_{s'}^{-1}$ is said to separate states s and s', and the word $u_s \cdot e \cdot u_{s'}^{-1}$ is said to inhibit the transition $s \stackrel{e}{\to} s'$. The following result gives a classification of the extensions of a reversible automaton, the conditions that a representing group should satisfy are analogues to the separation axioms introduced by Ehrenfeucht and Rozenberg for elementary transition systems. By Hrushovski's theorem [55] (see also [50]) we already know that any finite reversible automaton has a finite extension and the use of Hall's theorem in [5] in order to prove the existence of this finite extension was taken from the proof of Hrushovski's theorem given by Lascar and Herwig [51].

Proposition 3.2.2 *Every reversible automaton $A = (S,E,T,s_0)$ has an extension $A' = (E,S',T',s'_0)$ with the same fundamental group: $\pi_1(A') = \pi_1(A)$. If A is finite it has a finite extension. Up to isomorphism the extensions of a reversible automaton A are in bijective correspondance with the subgroups H of $F(E)$ such that $\pi_1(A) \subseteq H$ and $H \cap \Delta(A,U) = \emptyset$ where U is some spanning tree of A. The extension associated with H is $A_H = S(F(E),H,E)$ with full embedding $S \hookrightarrow H\backslash F(E) \ : \ s \mapsto Hu$ where $s_0 * u = s$. A can be fully embedded in a Cayley graph if and only if $N(\pi_1(A)) \cap \Delta(A,U) = \emptyset$ where $N(\pi_1(A))$ is the normaliser of the fundamental group in $F(E)$. Such a Cayley graph is then $C(G,E)$ where G is the group with presentation $G = \mathbf{gp}(E,B)$ where $B = \{c_t \mid t \in T\backslash U\}$ is the base of $\pi_1(A)$ derived from some spanning tree U of A.*

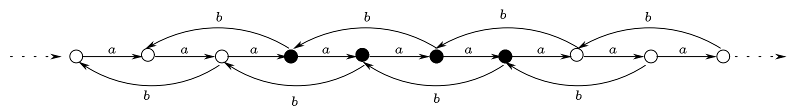
Corollary 3.2.3 $\pi_1(A) \cap \Delta(A,U) = \emptyset$.

Let A_1 be the following reversible automaton



Its fundamental group $\pi_1(A_1)$ is the subgroup of $F(E)$ generated by the elements aab and $aaaba^{-1}$ the normal closure of which is the group generated by the elements $uaabu^{-1}$ for $u \in F(E)$. Thus

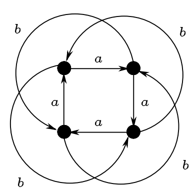
$$F(E)/N(\pi_1(A_1)) = \mathbf{gp}(\{a; b\}; aab) \cong \mathbb{Z}$$

$A_2 =$ 

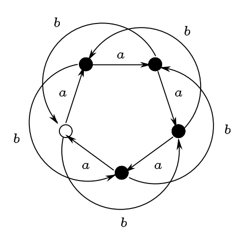
If U is the spanning tree of A, consisting of the a-labelled transitions,

$$\Delta(A_1, U) = \{a; a^2; a^3; a^4; ab; aba^{-1}; aba^{-2}; aba^{-3}; b; ba^{-1}; ba^{-2}; ba^{-3}\}$$

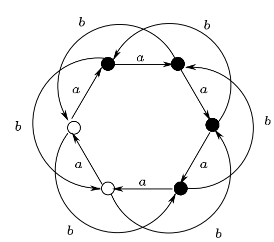
$\Delta(A_1, U) \cap N(\pi_1(A_1)) = \emptyset$ because none of the elements of $\Delta(A_1, U)$ maps to $0 \in \mathbb{Z}$. Thus the Cayley graph A_2 is an extension of A_1. If $X \subseteq F(E)$, we let $A_1(X)$ denote the automaton $S(F(E), H(X), E)$ where $H(X)$ is the normal closure of the subgroup of $F(E)$ generated by a^2b and the words in X. Then $A_1(a^6)$ is a finite extension of A_1.



$A_1(a^4)$



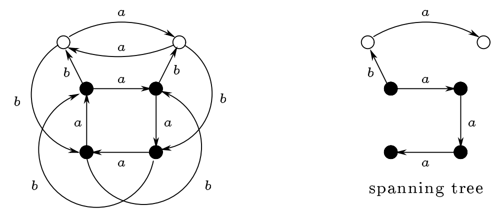
$A_1(a^5)$



$A_1(a^6)$

$A_1(a^4)$ is not an extension of A_1 because $a^4 \in \Delta(A_1, U)$ which corresponds to the adjunction of the extra transition $s_3 \xrightarrow{a} s_0$. Similarly $A_1(a^5)$ is not an extension of A_1 because $ba^{-3} = [a^{-2} \cdot (a^2 b) \cdot a^2] \cdot a^{-5} \in \Delta(A_1, U) \cap H(a^5)$, which corresponds to the adjunction of the transition $s_0 \xrightarrow{b} s_3$. Let A_3 the automaton obtained by adding the transition $s_3 \xrightarrow{a} s_0$ to A_1, then $\pi_1(A_3)$ is the subgroup of $F(E)$ generated by the elements aab, $aaaba^{-1}$ and a^4, and $\Delta(A_3, U) = \Delta(A_1, U) \setminus \{a^4\}$. A_3 does not fully embed in $S(F(E), N(\pi_1(A_3), E) = A_1(a^4)$ because of the two extra transitions $s_1 \xrightarrow{b} s_3$ and $s_0 \xrightarrow{b} s_2$ which correspond respectively to the elements $aba^{-3} = a^{-1} \cdot (a^2 b \cdot a^{-4}) \cdot a$ and $ba^{-2} = a^2 \cdot (a^{-4} \cdot a^2 b) \cdot a^{-2}$ of

$N(\pi_1(A_3)) \cap \Delta(A_3, U)$. Thus A_3 has no extension which is a Cayley graph. A finite extension of A_3 is the following:



It corresponds to the group H_3 generated by a^4, a^2b, a^3ba^{-1}, ba^2b^{-1}, $aba^{-1}b^{-1}$, b^2a^{-3}, and ab^2a^{-2} (associated with the chords of the spanning tree indicated in the figure).

3.2.2 Commutative Automata

The *Parikh mapping* is the (unique) morphism of groups $\psi : F(E) \to \mathbb{Z}^E$ such that $\pi(e)(e') = 1$ if $e = e'$ else 0. We write elements of the free abelian group as formal sums: $V = \sum V(e) \cdot e$, so that for instance $\psi(aba^{-2}) = -a + b$ and $\psi(ab^{-1}a^2b^2a) = 4a + b$. We term *commutative image* of an element $u \in F(E)$ or of a set $L \subseteq F(E)$ their respective images by the Parikh mapping, i.e. the vector $\psi(u) \in \mathbb{Z}^E$ and set of vectors $\psi(L) \subseteq \mathbb{Z}^E$ respectively. We let $V_A = \psi(L_A)$ denote the commutative image of the language of A.

Recall that the abelianization of a group G is the quotient of G by its commutator group $[G, G] = \{aba^{-1}b^{-1} | a, b \in G\}$. It is indeed an abelian group and the canonical projection $G \to G/[G, G]$ is universal among the morphisms $G \to A$ where A is abelian. If $G \subseteq F(E)$ is a subgroup of a free group its abelianization is isomorphic to its commutative image $\psi(G)$ which is a subgroup of $\mathbb{Z}^E$. Moreover if G is the subgroup of $F(E)$ generated by words $u_1, \ldots, u_n$ then $\psi(G)$ is the subgroup of $\mathbb{Z}^E$ generated by the vectors $\psi(u_1), \ldots, \psi(u_n)$; i.e. $\psi(G) = \{\sum_{i=1}^{n} \lambda_i \psi(u_i) | \lambda_i \in \mathbb{Z}\}$. The *first homology group* $H_1(A)$ of a reversible automaton is the abelianization $\psi(\pi_1(A))$ of its fundamental group. It therefore consists of the commutative images of the closed paths based on its initial state. But since the underlying graph is connected the initial state does not matter and the first homology group thus contains the commutative images of all closed paths. Notice that $V_A = \bigcup_{s \in S} \psi(u_s) + H_1(A)$.

Definition 3.2.4 *A reversible automaton A is a commutative automaton if and only if it satisfies the two following conditions:*

1. $\forall s, s' \in S \quad \forall u \in F(E) \qquad [s \xrightarrow{u} s' \ \wedge \ \psi(u) \in H_1(A)] \Rightarrow s = s'$
2. $\forall s, s' \in S \quad \forall u \in F(E) \quad \forall e \in E \qquad [s \xrightarrow{u} s' \ \wedge \ \psi(u) = e] \Rightarrow s \xrightarrow{e} s'$

We say that a reversible automaton A divises a reversible automaton B if there exists some reversible automaton C that covers A and fully embedds in B; in notation $A | B \Leftrightarrow \exists C \quad A \leq C \hookrightarrow B$.

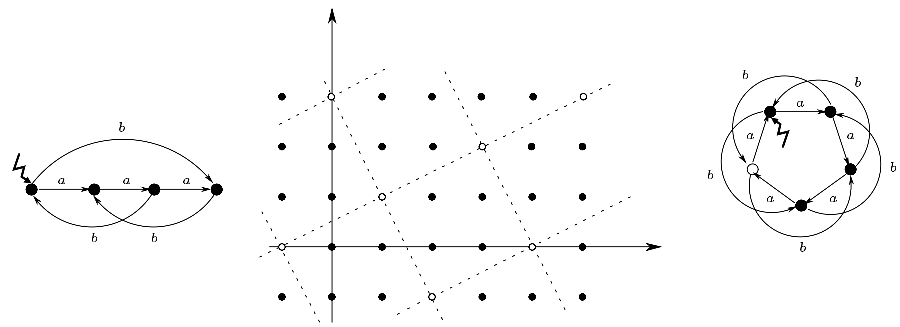


Figure 3.1: a commutative automaton

Proposition 3.2.5 *Let $A = (E, S, T, s_0)$ be a reversible automaton, then the following conditions are equivalent:*

1. *A is a commutative automaton,*
2. *A divides the Cayley graph of $\mathbb{Z}^E$,*
3. *A fully embeds in a Cayley graph of a finitely generated abelian group.*

Commutative automata with set of events E are in bijective correspondance with the pairs (V, H) where $V \subseteq \mathbb{Z}^E$ is a connected set of vectors and H is a subgroup of the group $I(V)$ of invariants of V: $H \subseteq I(V) = \{u \in \mathbb{Z}^E \mid u + V = V\}$. H is then the first homology group and V the commutative image of the language of the associated commutative automaton. A commutative automaton is reduced if and only if its first homology group coincides with its group of invariants.

In fact the conditions in Def. 3.2.4 state that the map $s \mapsto \psi(u_s) + H_1(A)$ is a full embedding of the automaton into the Cayley graph of $\mathbb{Z}^E/H_1(A)$. Consider the reversible automaton of Fig. 3.1, $H_1(A)$ is the group generated by $3a - b$ and $2a + b$, and $V_A = U_A + H_1(A)$ where $U_A = \{0; a; 2a; 3a\}$ (using the set of a-labelled transitions as spanning tree). The embedding of A into the Cayley graph of the group $\mathbb{Z}^E/H_1(A)$ takes a state s to the coset $\psi(u_s) + H_1(A)$; i.e. the orbit of $\psi(u_s)$ in $V_A \subseteq \mathbb{Z}^E$ for the action of $H_1(A)$. Now $H_1(A)$ is also generated by $5a$ and $3a - b$ (because $5a = (3a - b) + (2a + b)$) and thus $\mathbb{Z}^E/H_1(A)$ is isomorphic to the cyclic group $\mathbb{Z}/5\mathbb{Z}$ with a identified to 1 and b to $3 = -2$. The embedding of A into $\mathbb{Z}^E/H_1(A)$ is shown on the right of Fig. 3.1. We shall term the factor group $\mathcal{C}(A) = \mathbb{Z}^E/H_1(A)$ the *canonical group* of the reversible automaton $A = (E, S, T, s_0)$. The embedding of A into the Cayley graph of $\mathcal{C}(A)$ which takes a state s to the coset $\psi(u_s) + H_1(A)$ and an event e to its coset $e + H_1(A)$ is termed the *canonical representation* of A. We recall that any abelian group G has an explicit representation of the form $G \cong \mathbb{Z}/n_1\mathbb{Z} \times \ldots \times \mathbb{Z}/n_k\mathbb{Z} \times \mathbb{Z}^m$ where $1 \leq n_i | n_{i+1}$ and $m \geq 0$. The coefficients n_i and m are characteristic of K even though the isomorphism may not be uniquely determined. By abuse of notation we shall term *a canonical representation* for A any composition of its canonical representation with such an isomorphism. We describe in the next section how such canonical representations can be explicitely computed using some Smith normalization of a matrix associated with $\mathcal{C}(A)$.

3.2.3 Canonical Embeddings of a Commutative Automaton

If H is a subgroup of $\mathbb{Z}^n$ and $G = \mathbb{Z}^n/H$ is the factor group then G, as any commutative group, has a canonical decomposition $G \cong \mathbb{Z}/n_1\mathbb{Z} \times \ldots \mathbb{Z}/n_k\mathbb{Z} \times \mathbb{Z}^m$ where $1 \leq n_i | n_{i+1}$ and $m \geq 0$. These numbers called respectively the *torsion coefficients* and the *Betti number* are characteristic of G. They can be computed using the Smith normal form of a matrix associated with a presentation of G (see [64] page 140–150). Let us recall some facts about Smith normal forms of integral matrices [65]. Let $M \in \mathbb{Z}^{n,m}$ be a matrix of rank r, the whole numbers $f_0, f_1, \ldots, f_r$, where $f_0 = 1$ and f_k for $1 \leq k \leq r$ is the greatest common divisor of all nonzero determinants of k^{th} order submatrices of M are termed the *determinantal divisors* of M. Then $f_{k-1}|f_k$, the quotients q_k defined by $f_k = q_k f_{k-1}$ are the *invariant factors* of M. Matrix M is equivalent to a matrix S called its *Smith normal form* such that $S(i,i) = q_i$ for $i = 1, \ldots r$ and $S(i,j) = 0$ otherwise; i.e. M is of the form $M = RSC$ where S is in Smith normal form and R and C are elementary matrices corresponding respectively to a sequence of elementary row operations on M (interchanging rows or adding a multiple of a row to another row) and a sequence of elementary column operations on M. The Smith normal form of a matrix M is unique but not the elementary matrices R and C that may depend on the order in which the elementary operations needed to reach the normal form are performed.

Let $G = Ab(E, W)$ denote the group $\mathbb{Z}^E/H$ where H is the subgroup of $\mathbb{Z}^E$ generated by the elements of $W \subseteq \mathbb{Z}^E$. The pair (E, W) can be represented as a matrix, called the *relation matrix* of G, whose columns corresponds to the relators. We then compute the Smith normal form of this matrix. For instance the relation matrix for the group $G = Ab(a, b; 3a - b, 2a + b)$ associated with the commutative automaton of Fig. 3.1 is $M = \begin{pmatrix} 3 & 2 \\ -1 & 1 \end{pmatrix}$ and its Smith normal form is computed as follows:

$$\begin{aligned}
\begin{pmatrix} 3 & 2 \\ -1 & 1 \end{pmatrix} &= \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & -1 \\ 2 & 3 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \\
&= \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 2 & 1 \end{pmatrix} \begin{pmatrix} 1 & -1 \\ 0 & 5 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \\
&= \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 2 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 5 \end{pmatrix} \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \\
&= \begin{pmatrix} 2 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 5 \end{pmatrix} \begin{pmatrix} -1 & 1 \\ 1 & 0 \end{pmatrix}
\end{aligned}$$

Let $R_{i,j}$ and $R_{(i) \leftarrow (i)+c(j)}$ be the matrices the left multiplication by which amounts respectively to exchange rows i and j and to add c times row j to row i. The matrices $C_{i,j}$ and $C_{(i) \leftarrow (i)+c(j)}$ encoding elementary operations on columns are defined similarly. These matrices are nonsingular with inverses

$R_{i,j}^{-1} = R_{i,j}$ and $R_{(i)\leftarrow(i)+c(j)}^{-1} = R_{(i)\leftarrow(i)-c(j)}$ (similarly for the C matrices). In the previous exemple

$$R^{-1} = \begin{pmatrix} 1 & 0 \\ 2 & 1 \end{pmatrix}^{-1} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}^{-1} = \begin{pmatrix} 1 & 0 \\ -2 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & -2 \end{pmatrix}$$

and similarly $C^{-1} = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}$; and thus

$$\begin{pmatrix} 1 & 0 \\ 0 & 5 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & -2 \end{pmatrix} \begin{pmatrix} 3 & 2 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}$$

Therefore when computing the Smith normal form of a matrix we memorize the sequences of elementary operations performed on the sets of rows and columns from which the matrices R and C and their inverses can be obtained. Now the important property is the following.

Proposition 3.2.6 *Let $M \in \mathbb{Z}^{n,m}$ be the relation matrix of a group $G = Ab(E,W)$ (where $E = \{e_1, \ldots, e_n\}$ and $W = \{w_1, \ldots, w_m\}$) and $R \in \mathbb{Z}^{n,n}$ and $C \in \mathbb{Z}^{m,m}$ be respectively elementary row and column matrices. Then $N = RMC$ is the relation matrix of a group $G' = Ab(E', W')$ (where $E' = \{e'_1, \ldots, e'_n\}$ and $W' = \{w'_1, \ldots, w'_m\}$) isomorphic to G. The isomorphism takes the element $\sum_{i=1}^n \lambda_i e_i$ of G to the element $\sum_{i=1}^n \mu_i e'_i$ of G' where $\mu = R\lambda$.*

Let $M = RSC$ be the Smith normalisation of a relation matrix M of a group G, let $d_i = S(i,i)$ if $i \leq m$ and 0 if $i > m$ (this latter case can only occur if $m < n$ which can always be avoided up to the addition of redundant relators). Then by the preceding proposition we have an isomorphism $\varphi : \mathbb{Z}^n/G \to \mathbb{Z}/d_1\mathbb{Z} \times \ldots \mathbb{Z}/d_n\mathbb{Z}$ where $\psi(\sum_{i=1}^n \lambda_i e_i + G) = \sum_{i=1}^n (\mu_i \bmod d_i) e'_i$ with $\mu = R^{-1}\lambda$ and $e_i(j) = e'_i(j) = \delta_{i,j}$ (where $\delta_{i,j} = 1$ if $i = j$ and 0 otherwise). Notice that if $d = 1$ then the factor group $\mathbb{Z}/d\mathbb{Z}$ is the trivial group and may be omitted from the product and if $d = 0$ then factor group $\mathbb{Z}/d\mathbb{Z}$ is the infinite cyclic group $\mathbb{Z}$. The first k coefficients $d_1, \ldots, d_k$ (maybe $k = 0$) are equal to 1 and corresponds to the linear part that disappears, the last m coefficients are zeros where m is the Betti number, the intermediate values (distinct to 0 and 1) are the torsion coefficients. In our previous example we obtain an isomorphism $G \cong \mathbb{Z}/5\mathbb{Z}$ that takes $\lambda a + \mu b + H_1(A)$ to $\lambda - 2\mu \bmod 5$.

Corollary 3.2.7 *Let $\Psi : \mathcal{C}(A) \to \mathbb{Z}/d_1\mathbb{Z} \times \ldots \times \mathbb{Z}/d_n\mathbb{Z}$ be the isomorphism induced from some Smith normalisation of some relation matrix for the canonical group $\mathcal{C}(A) = \mathbb{Z}^E/H_1(A)$ of a reversible automaton A, and let $\lambda(s) = \Psi(\psi(u_s) + H_1(A))$, then the automaton A is commutative if and only if*

1. $\forall s, s' \in S \quad s \neq s' \Rightarrow \lambda(s) \neq \lambda(s')$,
2. $\forall s, s' \in S \;\; \forall e \in E \quad s \overset{e}{\not\to} s' \;\Rightarrow \lambda(s) + \lambda(e) \neq \lambda(s')$.

the embedding λ is then called a canonical representation *of the commutative automaton.*

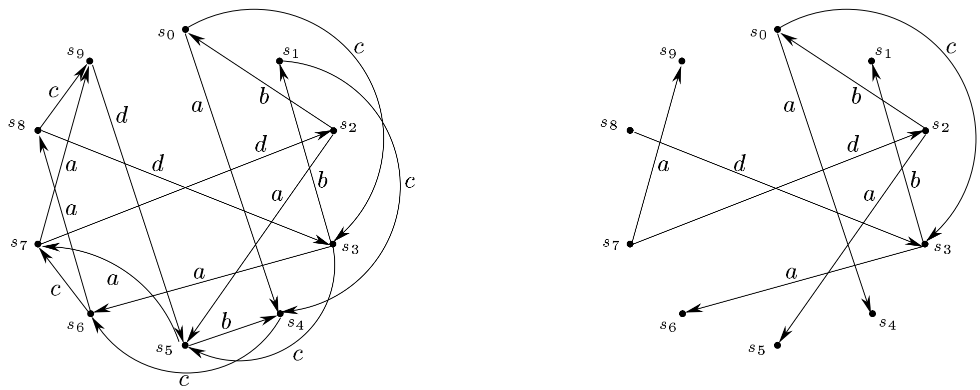


Figure 3.2: a reversible automaton and one of its spanning tree

3.2.4 Torsion-Free Commutative Automata

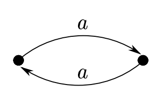
A commutative automaton A is termed *torsion-free* if its canonical group $\mathcal{C}(A) = \mathbb{Z}^E/H_1(A)$ is acyclic.

Observation 3.2.8 *If $s \xrightarrow{u} s$ is a closed path in a torsion-free commutative automaton whose commutative image is a multiple of some vector $V \in \mathbb{Z}^E$, i.e. $\psi(u) = \ell V$ where $\ell \in \mathbb{N} \setminus \{0\}$, then there exists a closed path $s \xrightarrow{v} s$ in A such that $\psi(v) = V$.*

Proposition 3.2.9 *The first homology group of a finite torsion-free commutative automaton coincides with its group of invariants: $H_1(A) = \{u \in \mathbb{Z}^E | u + V_A = V_A\}$.*

By Prop. 3.2.5, we deduce

Corollary 3.2.10 *Any finite torsion-free commutative automaton is reduced.*

Observe that the reversible automaton A= 

which is neither torsion-free nor reduced is such that V_A coincides with its group of linear invariants (isomorphic to $\mathbb{Z}$) while $H_1(A)$ is its subgroup $2\mathbb{Z}$.

3.2.5 An Example

Let us consider the reversible automaton of Fig. 3.2. The vectors $\psi_s = \psi(u_s)$ associated with the choice of spanning tree shown on the right of Fig. 3.2 are given in the following table.

	ψ_{s_0}	ψ_{s_1}	ψ_{s_2}	ψ_{s_3}	ψ_{s_4}	ψ_{s_5}	ψ_{s_6}	ψ_{s_7}	ψ_{s_8}	ψ_{s_9}
a	0	0	0	0	1	1	1	0	0	1
b	0	1	-1	0	0	-1	0	-1	0	-1
c	0	1	0	1	0	0	1	0	1	0
d	0	0	0	0	0	0	0	-1	-1	-1

The Parikh images of the fundamental cycles are given in the following table.

$t = s \xrightarrow{e} s'$	$\psi_t = \psi(c_t) = \psi_s + e - \psi_{s'}$
$s_1 \xrightarrow{c} s_4$	$-a+b+2c$
$s_3 \xrightarrow{c} s_5$	$-a+b+2c$
$s_4 \xrightarrow{c} s_6$	0
$s_5 \xrightarrow{b} s_4$	0
$s_6 \xrightarrow{c} s_7$	$a+b+2c+d$
$s_6 \xrightarrow{a} s_8$	$2a+d$
$s_8 \xrightarrow{c} s_9$	$-a+b+2c$
$s_9 \xrightarrow{d} s_5$	0

The relation matrix is therefore $M = \begin{pmatrix} -1 & 1 & 2 \\ 1 & 1 & 0 \\ 2 & 2 & 0 \\ 0 & 1 & 1 \end{pmatrix}$ and we compute its Smith normal form:

$$\begin{aligned}
\begin{pmatrix} -1 & 1 & 2 \\ 1 & 1 & 0 \\ 2 & 2 & 0 \\ 0 & 1 & 1 \end{pmatrix} &= R_{1,4} \cdot \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 2 & 2 \\ 2 & 1 & -1 \end{pmatrix} \cdot C_{1,3} \\
&= \underbrace{R_{1,4} \cdot R^{-1}_{(4)\leftarrow(4)-2(1)}}_{R_1} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 2 & 2 \\ 0 & -1 & -1 \end{pmatrix} \cdot \underbrace{C^{-1}_{(2)\leftarrow(2)-(1)} \cdot C_{1,3}}_{C_1} \\
&= R_1 \cdot R^{-1}_{(3)\leftarrow(3)-2(2)} \cdot R^{-1}_{(4)\leftarrow(4)+(2)} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \cdot C^{-1}_{(3)\leftarrow(3)-(2)} \cdot C_1
\end{aligned}$$

And thus

$$
\begin{aligned}
R^{-1} &= R_{(4)\leftarrow(4)+(2)} \cdot R_{(3)\leftarrow(3)-2(2)} \cdot R_{(4)\leftarrow(4)-2(1)} \cdot R_{1,4} \\
&= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}
\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -2 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}
\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -2 & 0 & 0 & 1 \end{pmatrix}
\begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \\
&= \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & -2 & 1 & 0 \\ 1 & 1 & 0 & -2 \end{pmatrix}
\end{aligned}
$$

The factor of the free abelian group $\mathbb{Z}^E \cong \mathbb{Z}^4$ by the first homology group of A has no torsion coefficient, its Betti number is 2. The canonical projection is given by the matrix Λ consisting of the last two rows of matrix R^{-1}:

$$\Lambda = \begin{pmatrix} 0 & -2 & 1 & 0 \\ 1 & 1 & 0 & -2 \end{pmatrix}$$

We introduce two symbols e and f associated with the dimensions of the quotient space, then matrix Λ gives a change of variables from the old alphabet $\{a; b; c; d\}$ to the new one $\{e; f\}$.

Λ	λ_a	λ_b	λ_c	λ_d
e	0	-2	1	0
f	1	1	0	-2

Similarly the states s are represented as $\lambda_s = \Lambda\psi_s$.

	λ_{s_0}	λ_{s_1}	λ_{s_2}	λ_{s_3}	λ_{s_4}	λ_{s_5}	λ_{s_6}	λ_{s_7}	λ_{s_8}	λ_{s_9}
e	2	1	2	1	2	0	1	2	-1	0
f	2	2	1	1	0	1	0	-1	1	0

Every state is represented by a distinct vector and thus λ gives an embedding, shown on Fig. 3.3, of the automaton in the Cayley graph $C(\mathbb{Z}^2, \{\lambda_e | e \in E\})$. We readily verify that λ is a full embedding as it happens that $\lambda_{s'} = \lambda_s + \lambda_e$ only if $s \xrightarrow{e} s'$.

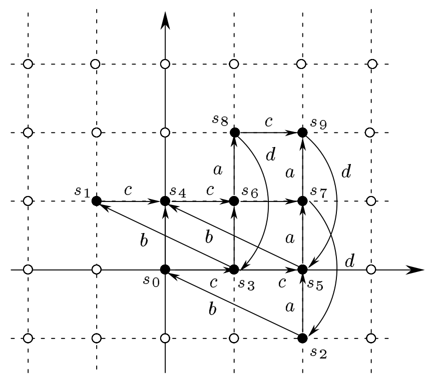


Figure 3.3: embedding of the automaton of Fig. 3.2 in $\mathbb{Z}^2$

Chapter 4

Vector Addition Systems

A place of a vector addition system is a synchronic constraint on the events occurring in the system, namely it constraints the relative frequence of execution of the transitions affecting that place. If the place is bounded its range of variation that is, the difference between its minimum and maximum values, is a measure of the reciprocal independence: higher values mean looser constraints. This is in agreement with the interpretation of a place as abstract resource shared by the transitions connected to that place.

This observation allows us to characterize the state graphs of vector addition systems as the maximal quotients of polyhedral automata. They are commutative automata and we conjecture that they are torsion-free, i.e. that their canonical representation does not contain torsion element. We give an algorithm for the synthesis of vector addition systems which relies on that conjecture.

The notion of synchronic distance was introduced, in the context of net theory, by C.A. Petri as a tool to measure the relative degree of freedom between sets of transitions in a concurrent system. It has been used in [21] to define a notion a *generalized regions* allowing to adapt the representation theorem of Ehrenfeucht and Rozenberg to the context of vector addition systems. A slightly different approach to synchronic relations in Petri nets was set out in [95], where linear programming techniques are used to compute upper bounds of synchronic invariants in a given net. In [6] the use of linear algebraic techniques provides a polynomial time algorithm solving the synthesis problem of vector addition systems.

4.1 State Graphs of Vector Addition Systems

4.1.1 Vector Addition Systems

A *vector addition system* [60] is a triple $N = (P, E, \mu)$ consisting of a finite set of *places* $P = \{p_1, \ldots, p_m\}$, a finite set of *events* $E = \{e_1, \ldots, e_n\}$, and a matrix $\mu : P \times E \to \mathbb{Z}$. A *marking* is any vector $M \in \mathbb{N}^m$, and the *state graph* of N is the transition system $N^* \subseteq \mathbb{N}^m \times E \times \mathbb{N}^m$ given by $(M, e, M') \in N^* \Leftrightarrow M' =$

$M + \mu \cdot e$. We often write $M[e>M'$ as an abbreviation for $(M, e, M') \in N^*$, and $M[e>$ when $M[e>M'$ for some M' in which case event e is said to be *enabled* in marking M.

The transition system induced by a reversible transition system $\mathbf{T} = (S, E, T)$ on a subset of states $S' \subseteq S$ is the reversible transition system defined by $\mathbf{T} \upharpoonright S' = (S', E, T \cap (S' \times E \times S')$.

Observation 4.1.1 *The state graph of a vector addition system is the reversible transition system induced by the Cayley graph of a power of $\mathbb{Z}$ on the set of vectors of non negative entries.*

But of course not all induced sub-transition systems of a Cayley Graph of $\mathbb{Z}^n$ are state graphs of vector addition systems. The main purpose of this chapter is to search for an effective procedure that decides whether a given finite reversible transition system is an induced sub-transition system of a Cayley Graph of $\mathbb{Z}^n$ and whether it is isomorphic to the state graph of some vector addition system.

If $s \in S$ is some state of a reversible transition system $\mathbf{T} = (S, E, T)$ we let $(\mathbf{T}, s)$ denote the reversible automaton whose transition relation is induced by T on the connected component of s. For instance if M_0 is some marking of a vector addition system $N = (P, E, \mu)$, the reversible automaton (N^*, M_0) will be termed the state graph of the marked vector addition system (P, E, μ, M_0).

4.1.2 Polyhedral Graphs

State graphs of (marked) vector addition systems are commutative automata. We introduce a class of automata, termed polyhedral, that are their abelian unfoldings.

A $\mathbb{Z}$-polyhedron is a set of vectors of $\mathbb{Z}^n$ defined by a finite set of affine inequalities. All polyhedra that we consider in this document contain the origin and therefore are of the form $\{x \in \mathbb{Z}^n \mid Ax + b \geq 0\}$ for some matrix $A \in \mathbb{Z}^{m,n}$ and vector $b \in \mathbb{N}^m$. Such a polyhedron may be viewed as a reversible transition system, termed a *polyhedral transition system*, whose states are the vectors of the polyhedron, whose events are given by $e_i(j) = 1 \quad \text{if } i = j \quad \text{else } 0$, and whose transition relation $T \subseteq S \times E \times S$ is given by $s \xrightarrow{e} s' \Leftrightarrow s' = s + e$.

A *polyhedral automaton* is a reversible automaton of the form $(\mathbf{T}, 0)$ where $\mathbf{T}$ is a polyhedral transition system, i.e. it is the connected component of the origin in some polyhedron of $\mathbb{Z}^n$. One should pay attention to the fact that polyhedral automaton is not synonymous to connected polyhedral transition system as it may happen, as shown in Fig. 4.1, that a connected component of the set of integral elements of a convex of $\mathbb{R}^n$ be strictly included in the set of integral elements of its convex hull. We call a *polyhedral graph* the undirected graph associated with a polyhedral transition system, i.e. it is an undirected graph whose vertices are the integral vectors of a polyhedron and such that any two vertices are connected by an edge if and only if their euclidean distance is 1. Thus the above example shows that a connected component of a polyhedral graph may not be a polyhedral graph.

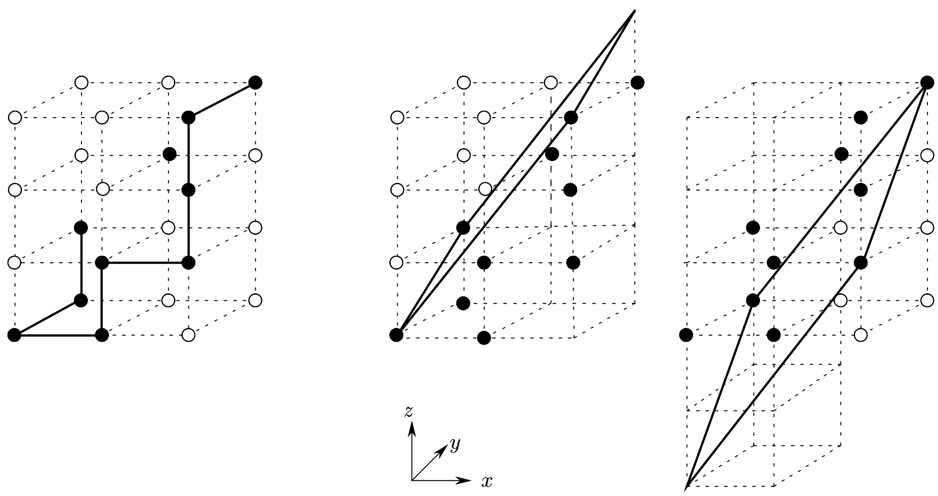


Figure 4.1: the set X of points of the three dimensional grid indicated by black dots is the set of integral elements of its convex hull K . As shown by the two right-hand side figures, this convex K is the polyhedron obtained by intersecting the paralleliped ($0 \leq x \leq 2$, $0 \leq y \leq 1$, and $0 \leq z \leq 3$) with two half spaces ($3x + 2y - 2z \geq 0$ and $-3x - 4y + 2z + 4 \geq 0$). Now set X has two connected components one of which is an isolated point that belongs to the convex hull of the other one

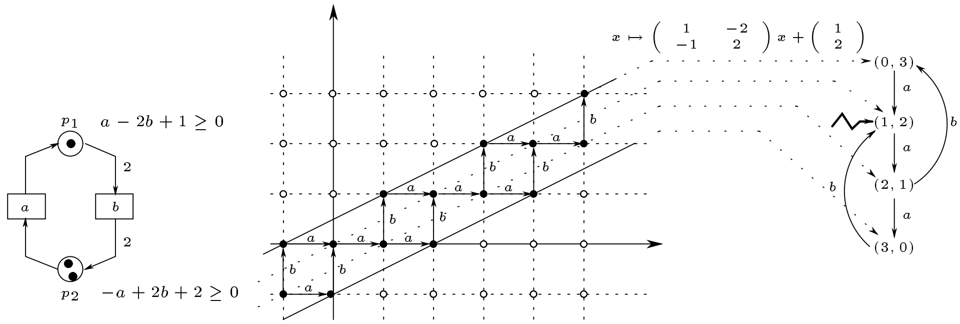


Figure 4.2: a vector addition system with two events a and b and two places p_1 and p_2 corresponding to the synchronic constraints $a - 2b + 1 \geq 0$ and $-a + 2b + 2 \geq 0$, its marking graph (shown on the right) is a quotient of the corresponding polyhedral automaton.

4.1.3 Places as Synchronic Constraints

A place $p \in P = \{p_1, \ldots, p_n\}$ of a (marked) vector addition system $N = (P, E, \mu, M_0)$ is a synchronic constraint on the language of N. To make that fact clear, let us recall that $M \xrightarrow{e} M'$, where M and M' are markings and $e \in E$ an event, if and only if $M' = M + \mu \cdot e$; from which it follows that $M \xrightarrow{u} M'$, where M and M' are markings and $u \in F(E)$ is a (reduced) word, if and only if $M' = M + \mu \cdot \psi(v)$ for all prefixes v of u. We can decompose the equation in matrix form $M' = M + \mu \cdot \psi(v)$ into the set of linear equations $M'(p_i) = M(p_i) + \mu_i \cdot \psi(v)$ indexed by places $p_i \in P$. Then $M_0 \xrightarrow{u} M$ for some marking M if and only if all prefixes v of u satisfies the set of linear constraints $\{M_0(p_i) + \mu_i \cdot \psi(v) \geq 0 \mid p_i \in P\}$ i.e. if and only if $0 \xrightarrow{u} \psi(u)$ in the polyhedral automaton with set of states $\{x \in \mathbb{Z}^m \mid \mu \cdot x + M_0 \geq 0\}$, where $E = \{e_1, \ldots, e_m\}$. Conversely each $\mathbb{Z}$-polyhedron presented as $\{x \in \mathbb{Z}^m \mid A \cdot x + b \geq 0\}$ where $A \in \mathbb{Z}^{m,n}$ and $b \in \mathbb{N}^n$ can be viewed as a vector addition system whose places are given by the synchronic constraints $A(i, \cdot) \cdot M + b(i) \geq 0$. To sum up:

a place of a vector addition system	$\equiv$	a synchronic constraint
	$\equiv$	an affine hyperplan

a vector addition system	$\equiv$	a set of synchronic constraints
	$\equiv$	a (presentation of) $\mathbb{Z}$-polyhedron

Now by the previous discussion, the map $x \mapsto \mu \cdot x + M_0$ which takes a vector of the $\mathbb{Z}$-polyhedron $\{x \in \mathbb{Z}^m \mid \mu \cdot x + M_0 \geq 0\}$, associated with the vector addition system $N = (P, E, \mu, M_0)$, to the vector of "distances" of x to each of the hyperplanes bording the polyhedron is a covering from the polyhedral automaton to the marking graph of N.

Proposition 4.1.2 *A reversible automaton is isomorphic to the state graph of a (marked) vector addition system if and only if it is a reduced commutative automaton the commutative image of whose language is a connected component of a polyhedral graph.*

4.1.4 An Algorithm

By Prop. 3.2.5 the state graph of a (marked) vector addition system is then characterized by the commutative image of its language, which is by Prop. 4.1.2 a connected component of a polyhedral graph, and its first homology group. If $V \subseteq \mathbb{Z}^n$, let $I(V)$ denote its group of invariants, i.e. $I(V) = \{u \in \mathbb{Z}^n | u + V = V\}$; and let $G(V) = \mathbb{Z}^n / I(V)$ denote the corresponding factor group.

Observation 4.1.3 *If V is the set of vertices of a polyhedral graph, then $G(V)$ is acyclic.*

Indeed, let u be a vector and ℓ a positive integer such that $\ell u \in I(V)$, then $V + \ell u = V$ and by convexity of V it follows that $v + u$ and $v - u$ belong to V for every $v \in V$ and thus $u + V = V$ which proves that $u \in I(V)$ and therefore that $G(V)$ is acyclic.

We conjecture that $G(V)$ is also acyclic if V is a connected component of a polyhedral graph, i.e.

Conjecture 4.1.4 *State graphs of vector addition systems are torsion-free commutative automaton.*

If A is the marking graph of a vector addition system (P, E, μ, M_0), then $u \in H_1(A)$ if and only if $\mu u = 0$ and there exists a path labelled u. Thus Conjecture 4.1.4 may be reformulated as

Conjecture 4.1.5 *If there exists a path between two vertices of a polyhedral graph of the form u and $u + \ell v$ where v is an invariant and ℓ is a positive integer, then there exists a path between u and $u + v$.*

Another conjecture (stronger than the previous ones) is the following:

Conjecture 4.1.6 *If V is a connected component of a polyhedral graph whose set of vertices $P = \{x \in \mathbb{Z}^n | \mu x + b \geq 0\}$ (where μ and b are a matrix and a vector with integral entries) is the set of integral elements of the convex hull of V, then V and P have the same group of invariants: $I(V) = I(P)$.*

Remark 4.1.7 *Since $K(u + V) = u + K(V)$ where $K(V)$ denote the convex hull of V we deduce that $I(V) \subseteq I(P)$; so we only have to prove the converse implication, i.e. $\mu \cdot u = 0 \Rightarrow \exists v_1, v_2 \in V \quad v_2 - v_1 = u$.*

Conjecture 4.1.6 indeed implies Conjecture 4.1.4 because if $u \in \mathbb{Z}^n$ and $\ell \in \mathbb{N}$ are such that $\ell u \in I(V)$, then $\ell u \in I(P)$ by the above remark, $u \in I(P)$ by Obs. 4.1.3, and $u \in I(V)$ by Conjecture 4.1.6.

Since finite torsion-free commutative automata are reduced a consequence of Conjecture 4.1.4 is

Corollary 4.1.8 *A finite reversible automaton is isomorphic to the state graph of a vector addition system if and only if it is a torsion-free commutative automaton the commutative image of whose language is a connected component of a polyhedron.*

Proposition 4.1.9 *Let $V \subseteq \mathbb{Z}^n$ be a set of vectors with invariants $I = \{u \in \mathbb{Z}^n \mid u + V = V\}$ and $G = \mathbb{Z}^n / I$ be the corresponding factor group. If G is acyclic then V is convex in $\mathbb{Z}^n$ if and only if its image V' is convex in $G \cong \mathbb{Z}^m$; moreover $H \subseteq \mathbb{Z}^n$ is a bounding hyperplane of V if and only if its image $H' \subseteq \mathbb{Z}^m$ is a bounding hyperplane of V'.*

As we already noticed however, the commutative image of the language of a vector addition system is not necessarily convex. If $X \subseteq \mathbb{Z}^n$ we let $K(X) \subseteq \mathbb{R}^n$ denotes the convex hull of X (in $\mathbb{R}^n$) and $K_I(X) = K(X) \cap \mathbb{Z}^n$ the set of integral elements of the convex hull of X.

Proposition 4.1.10 *Let Ψ be a canonical representation of a torsion-free commutative automaton A, then $K_I(V_A) = \Psi^{-1}(K_I(\Psi(V_A)))$.*

If state graph of vector addition systems prove to be torsion-free the following is an algorithm which decides whether a given finite reversible automaton is isomorphic to the marking graph of a vector addition system and which construct such a system when it exists.

1. Choose some spanning tree U of the graph of the automaton $A = (E, S, T, s_0)$. Let u_s denote the (reduced) word labelling the (unique) path in U from the initial state s_0 to state s, and let $c_t = u_s \cdot e \cdot u_{s'}^{-1}$ be the cycle associated with the chord $t = s \xrightarrow{e} s'$. Compute the commutative images of the chains u_s and of the cycles c_t: let $\psi_s = \psi(u_s)$ and $\psi_t = \psi(c_t)$.

2. Form the relation matrix M associated with the presentation $H_1(A) = Ab(E, \{\pi_t | t \in T \setminus U\})$ of the first homology group of the automaton given by the choice of the spanning tree; i.e. the columns of M are the commutative images of the fundamental cycles associated with U. Compute the Smith normal form of this relation matrix $M = RSC$ (we recall that matrix R^{-1} is computed along this normalization process).

3. If $G = \mathbb{Z}^E / H_1(A)$ has torsion coefficients (i.e. S contains entries distinct from 0 and 1) then the automaton is not isomorphic to the state graph of a vector addition system else proceed to the following steps.

4. If $n = |E|$ is the size of the alphabet and m is the Betti number of G, let $\Lambda \in \mathbb{Z}^{m,n}$ be the matrix consisting of the last m rows of matrix R^{-1}. The columns of Λ are indexed by the elements of the alphabet, let λ_e denote the column associated with $e \in E$. Compute $\lambda_s = \Lambda \psi_s$.

5. Check that the mapping λ represents the automaton as the subgraph of the Cayley graph $C(\mathbb{Z}^m, \{\lambda_e | e \in E\})$ induced on the subset of nodes $\lambda_S = \{\lambda_s | s \in S\}$. This amounts to verify the following two conditions

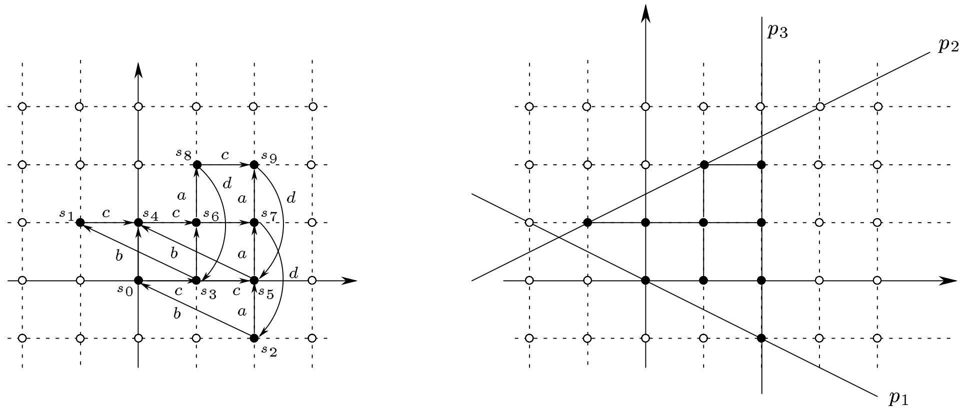


Figure 4.3: embedding of the automaton of Fig. 3.2 in $\mathbb{Z}^2$

(a) $\forall s, s' \in S \quad s \neq s' \Rightarrow \lambda_s \neq \lambda_{s'}$.

(b) $\forall s, s' \in S \;\; \forall e \in E \quad s \stackrel{e}{\not\to} s' \;\Rightarrow \lambda_s + \lambda_e \neq \lambda_{s'}$.

If one of the above conditions is not satisfied then the automaton is not isomorphic to the state graph of a vector addition system else proceed to the following steps.

6. Compute a polyhedral presentation $\{x \in \mathbb{Z}^m | \; \Pi x + b \geq 0\}$ (with $\Pi \in \mathbb{Z}^{r,m}$ and $b \in \mathbb{N}^m$) of the convex hull of the set λ_S. Its inverse image is the polyhedron $\{y \in \mathbb{Z}^n | \; \Pi\Lambda x + b \geq 0\}$ which is associated with the marked vector addition system $N = (P, E, \mu, M_0)$ with places $P = \{p_1, \ldots, p_r\}$ and such that $\mu(p, e) = p^t \Pi \Lambda e$, and $M_0(p) = p^t b$. Then the automaton A is isomorphic to the state graph of some vector addition vector systems if and only if it is isomorphic to the state graph of N if and only there is no element $v \in K(\lambda_S) \setminus \lambda_S$ of the form $v = \lambda_s + \lambda_e$ or $v = \lambda_s - \lambda_e$.

Let us consider the commutative automaton of section 3.2.5, it fully embedds in the Cayley graph of $\mathbb{Z}^2$ as shown on the left of Fig. 4.3. The image of this embedding is the set of integral points of the polytope delimited by three hyperplanes p_1, p_2 and p_3 as shown on the right of Fig. 4.3. This polytope is represented by the system

$$\left\{ \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \in \mathbb{R}^2 \;\middle|\; \begin{pmatrix} 1 & 2 \\ 1 & -2 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 0 \\ 3 \\ 2 \end{pmatrix} \geq \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \right\}$$

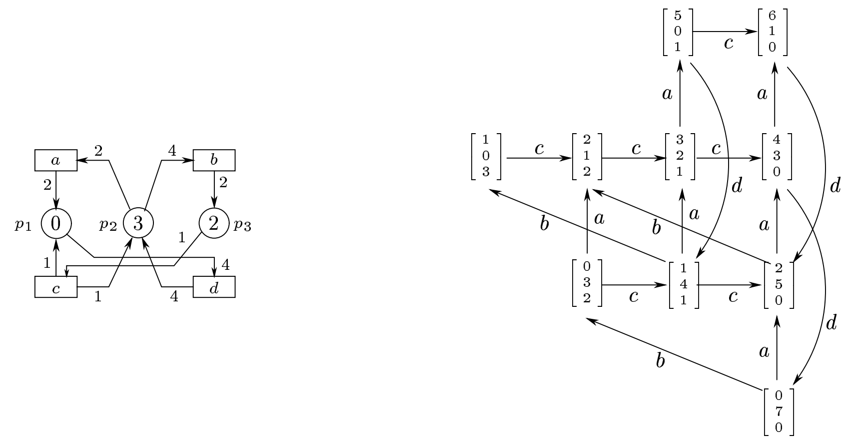


Figure 4.4: the vector addition system associated with the automaton of Fig. 3.2 and its state graph

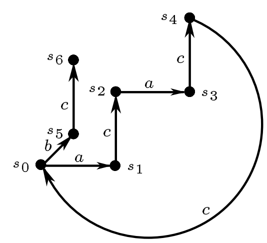


Figure 4.5: a reversible automaton

whose inverse image by λ is the polyhedron of $\mathbb{Z}^4$ given by

$$\left\{ \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix} \in \mathbb{R}^4 \;\middle|\; \begin{pmatrix} 2 & 0 & 1 & -4 \\ -2 & -4 & 1 & 4 \\ 0 & 2 & -1 & 0 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix} + \begin{pmatrix} 0 \\ 3 \\ 2 \end{pmatrix} \geq \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \right\}$$

Figure 4.4 gives the associated marked vector addition system together with its state graph.

Let us now consider the reversible automaton of Fig. 4.5. The vectors ψ_s associated with the choice of spanning tree obtained by discarding transition $s_4 \overset{c}{\to} s_0$ are given in the following table.

	ψ_{s_0}	ψ_{s_1}	ψ_{s_2}	ψ_{s_3}	ψ_{s_4}	ψ_{s_5}	ψ_{s_6}
a	0	1	1	2	2	0	0
b	0	0	0	0	0	1	1
c	0	0	1	1	2	0	1

The fundamental cycle associated with the chord is $2a+3c$. The Smith normalisation of the relation matrix is

$$\begin{pmatrix} 2 \\ 0 \\ 3 \end{pmatrix} = R \cdot \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

where $R = \begin{pmatrix} 2 & 0 & 1 \\ 0 & 1 & 1 \\ 3 & 0 & 1 \end{pmatrix}$ and $R^{-1} = \begin{pmatrix} -1 & 0 & 1 \\ 0 & 1 & 0 \\ 3 & 0 & -2 \end{pmatrix}$. The factor of the free abelian group $\mathbb{Z}^E \cong \mathbb{Z}^3$ by the first homology group of A has no torsion coefficient, its Betti number is 2. The canonical projection is given by the matrix Λ consisting of the last two rows of matrix R^{-1}:

$$\Lambda = \begin{pmatrix} 0 & 1 & 0 \\ 3 & 0 & -2 \end{pmatrix}$$

We introduce two symbols e and f associated with the dimensions of the quotient space, then matrix Λ gives a change of variables from the old alphabet $\{a; b; c; d\}$ to the new one $\{e; f\}$.

Λ	λ_a	λ_b	λ_c
e	0	1	0
f	3	0	-2

Similarly the states s are represented as $\lambda_s = \Lambda\psi_s$.

	λ_{s_0}	λ_{s_1}	λ_{s_2}	λ_{s_3}	λ_{s_4}	λ_{s_5}	λ_{s_6}
e	0	0	0	0	0	1	1
f	0	3	1	4	2	0	-2

Every state is represented by a distinct vector and thus λ gives an embedding, shown on the left of Fig. 4.6, of the automaton in the Cayley graph $C(\mathbb{Z}^2, \{\lambda_e | e \in E\})$. We readily verify that λ is a full embedding as it happens that $\lambda_{s'} = \lambda_s + \lambda_e$ only if $s \xrightarrow{e} s'$. The convex closure of the image of this embedding is a polytope of $\mathbb{Z}^2$ which contains an integral element that is not in the image of the embedding (the grey state in Fig. 4.6). However this element is not of the form $\lambda_s + \lambda_e$ or $\lambda_s - \lambda_e$ and therefore the automaton of Fig. 4.5 is isomorphic to the state graph of a vector addition system. For computing such a vector addition system we represent the polytope by the system

$$\left\{ \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \in \mathbb{R}^2 \;\middle|\; \begin{pmatrix} 1 & 0 \\ -1 & 0 \\ -4 & -1 \\ 2 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 4 \\ 0 \end{pmatrix} \geq \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \right\}$$

whose inverse image by λ is a polyhedron of $\mathbb{Z}^3$ given by

$$\left\{ \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} \in \mathbb{R}^3 \;\middle|\; \begin{pmatrix} 0 & 1 & 0 \\ 0 & -1 & 0 \\ -3 & -4 & 2 \\ 3 & 2 & -2 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 4 \\ 0 \end{pmatrix} \geq \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \right\}$$

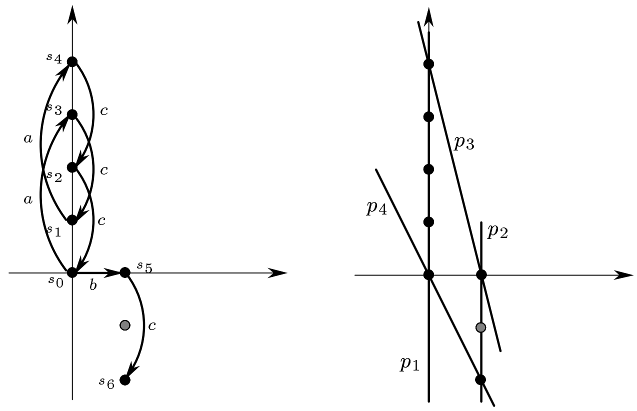


Figure 4.6: embedding of the automaton of Fig. 4.5 in $\mathbb{Z}^2$

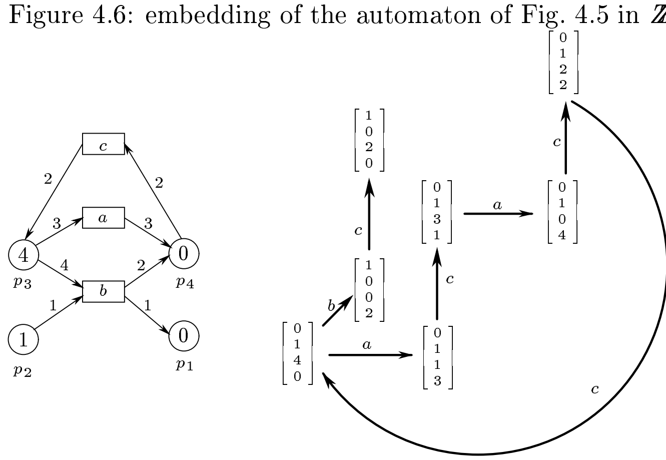


Figure 4.7: the vector addition system associated with the automaton of Fig. 4.5 and its state graph

Figure 4.7 gives the associated marked vector addition system together with its state graph.

In order to conclude let us give some hints on the complexity of the algorithm. The two costy parts of the algorithm are the computation of the Smith normal form of the relation matrix of the canonical group and the computation of the convex hull of the representation of the state space.

Kannan and Bachem [59] give a polynomial time algorithm for the computation of the Smith normal form of an integer matrix. They indeed prove that both the number of involved algebraic operations and the number of binary digits of all the intermediate numbers are bounded by polynomials in the length of the input data encoded in binary. Another solution is proposed by Rayward-Smith in [84]. Chou and Collins [32] and then Iliopoulos [56] improve the result of Kannan and Bachem by giving an $O(s^5 M(s^2))$ elementary operation algorithm for

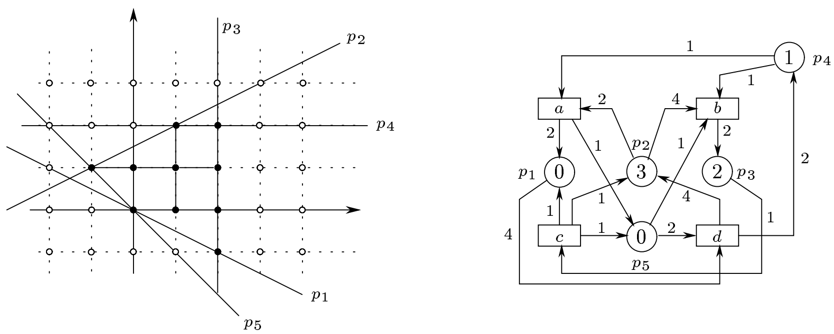


Figure 4.8: a convex hull for the representation of the state space of the automaton of Fig. 4.3 with a non minimal number of facets leading to a vector addition system with two redundant places as compared to the vector addition system indicated in Fig. 4.4

computing the Smith normal form of an integer matrix where $M(n)$ denotes an upper bound on the number of elementary operations required for the multiplication of two integers of length n bits, and where the size s of an $m \times n$ matrix A is the number $m + n + log\|A\|$ with $\|A\| = max_{i,j}\{|a_{i,j}|\}$. Computing the convex hull of a finite set of points in an euclidean space is one of the most fundamental problem of computational geometry [40, 41]. There exist algorithms running in time $O(n \log n)$ in dimension 2 or 3. General algorithms construct the convex hull of a set of n points in $\mathbb{Z}^d$ in $O(n \log n + n^{\lfloor (d+1)/2 \rfloor})$ in time and $O(n^{\lfloor d/2 \rfloor})$ in space. Since d corresponds to the size of the alphabet minus the dimension of the space of cycles, we see that this worst case complexity is exponential in the size of the automaton. However the points in the representation of the state space of a commutative automaton are not randomly distributed: they are "almost" all the integral points of a polyhedron, and the convex hull may be computed from its extremal points only. A comparison of the number n of integral points of a d dimensional polyhedron with $m^{\lfloor (d+1)/2 \rfloor}$ where m is the number of its extremal points may give a better hint to the complexity of our algorithm. In the same spirit, Seidel [92] and Swart [99] give the complexity of the convex hull construction in term of the output of the algorithm, namely the number of facets that it computes. Seidel gives an algorithm which has running time $O(n^2 + F \log n)$ where F is the number of facets produced by the algorithm. As we shall see in the next section, only a polynomial number of places need to be synthetized when solving the synthesis problem for vector addition systems: there exists a convex hull with a number of facets $F \leq n \times ((n-1) \times 2 \times p)$ where n is the number of states and p the number of events of the automaton when this automaton is isomorphic to the state graph of a vector addition system. However as shown in Fig. 4.8 the construction of the convex hull of the representation of the state space may lead to a polyhedron with a non minimal number of facets; and we have no indication on the number of facets that may be produced for an arbitrary automaton. Nevertheless it suggests that we can hope than on average the number of facets be polynomially bounded by the size of the automaton.

The tool SYNET [28] implements the algorithm of [6] that we describe in the next section, and provides a computer assisted solution to the distribution of protocols. When the automaton decribing a protocol fails to be isomorphic to the state graph of a vector addition system informations are provided in order to assist the designer to modify the specification of the protocol. In that respect, the algorithm described above may give further informations. In particular the canonical representation of a commutative automaton may give useful informations for the distribution of the automaton even if it fails to be isomorphic to the state graph of a vector addition system.

4.2 The Synthesis of Vector Addition Systems

In this section we adapt the representation theorem of Ehrenfeucht and Rozenberg to the context of vector addition systems and we describe a polynomial time algorithm for the synthesis of vector addition systems from their sequential state graphs.

4.2.1 Generalized Regions

Definition 4.2.1 *a (generalized)* [1] *region of a reversible automaton* $A = (S, E, T, s_0)$ *with alphabet* $E = \{e_1, \ldots, e_n\}$ *is any closed half space of the form* $R = \{x \in \mathbb{Z}^n | u \cdot x + u_0 \geq 0\}$ *where* $u \in \mathbb{Z}^n$ *and* $u_0 \in \mathbb{N}$ *that contains the commutative image* $V_A = \psi(L(A))$ *of the language of* A*.*

Each place p of a vector addition system $N = (P, E, \mu, M_0)$ induces a region $R_p = \{x \in \mathbb{Z}^n | \mu(p, -) \cdot x + M_0 \geq 0\}$ of its state graph. R_p is called the *extension* of place p, see Fig. 4.9. We shall write $R = \mathbf{region}(u, u_0)$ the region $R = \{x \in \mathbb{Z}^n | u \cdot x + u_0 \geq 0\}$ presented by $u \in \mathbb{Z}^n$ and $u_0 \in \mathbb{N}$. A minimal region (w.r.t. set inclusion) is therefore a bording half space of V_A i.e. a closed half space containing V_A such that there exists at least one element of V_A in its bording hyperplane. We recall that if U is some spanning tree of A, the commutaive image of the language of A is given by $V_A = \bigcup_{s \in S}[\psi(u_s) + H_1(A)]$ it follows that a half space $R = \{x \in \mathbb{Z}^n | u \cdot x + u_0 \geq 0\}$ associated with $u \in \mathbb{Z}^n$ and $u_0 \in \mathbb{N}$ is a region for A if and only if $H_1(A) \subseteq ker(u)$ (i.e. $\forall v \in H_1(A) \quad u \cdot v = 0$) and $u \cdot \psi(u_s) + u_0 \geq 0$ for all $s \in S$. Conversely if these two conditions hold we can define a map (also denoted R) $R : S \to \mathbb{N}$ by letting $R(s) = u \cdot \psi(v) + u_0$ where v is an arbitrary path from s_0 to s (two such paths differ by an element of $\pi_1(A)$). This map satisfies the following condition

$$[s_1 \xrightarrow{e} s_1' \quad \wedge \quad s_2 \xrightarrow{e} s_2'] \Rightarrow R(s_1') - R(s_1) = R(s_2') - R(s_2)$$

[1] we shall drop the epithet "generalized" as long as there is no confusion with the region associated with elementary net systems.

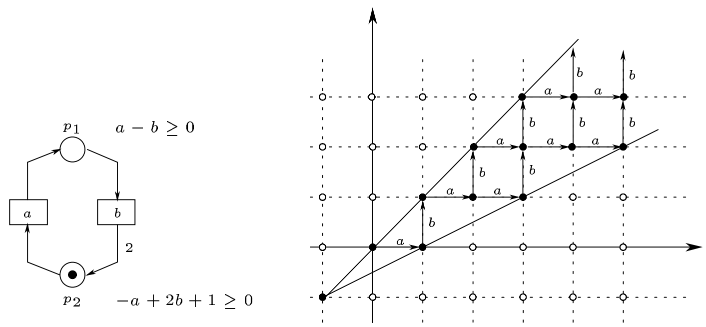


Figure 4.9: extensions of places are regions

A map $R : S \to \mathbb{N}$ which satisfies the above condition is termed a *potential* on A. A potential R induces a map $R : E \to \mathbb{Z}$ (also denoted $R!$) which when construed as a vector $u \in \mathbb{Z}^n$ satisfies $\forall v \in H_1(A) \quad u \cdot v = 0$ because the elements of $H_1(A)$ are the commutative images of the closed paths and the variation of potential along any closed path is null. Conversely such a vector that we term *coboundary* determines the potential up to an additive constant, and it does derive from a potential if and only if $u_0 = \mathbf{min}\{-u \cdot \psi(u_s) | s \in S\}$ exists (here again this does not upon the choice of the spanning tree since any two paths from s_0 to s differ from an element of $\pi_1(A)$) and $R = \mathbf{region}(u, u_0)$ is then a region which determines the potential from which this coboundary derives. To sum up we have a bijective correspondence between regions and potentials, and if the automaton is finite (in which case the above minimal bound certainly exists) we have a bijective correspondence between minimal regions and coboundaries. Observe that the latter in general does not hold for an infinite automaton. For instance the first homology group of the state graph of the vector addition system of Fig. 4.9 is trivial and thus every vector of $\mathbb{Z}^2$ is a co-boundary whereas there is only two minimal regions which are associated with the vectors $a - b$ and $-a + 2b$.

4.2.2 A Galois connection between Nets and Automata

As we have already mentioned a presentation of a polyhedron $P = \{x \in \mathbb{Z}^n | \mu \cdot x + b \geq 0\}$ where $\mu \in \mathbb{Z}^{m,n}$ and $b \in \mathbb{N}^n$ can be identified with the vector addition system with representation matrix μ and initial marking b. If we make no distinction between vector addition systems which differ only in the order in which the inequalities appear in the presentation, then the set $\mathbf{VAS}(E)$ of vector addition systems on the set of events E is a complete lattice whose order relation is given by the inclusion of the presentations viewed as multi-sets [2] of inequations. For sake of simplicity we will admit in this section that a vector addition system may have an infinite number of places. The above order relation is reflected by the opposite set-theoretic inclusion between the polyhedra they represent. The

[2] we have no reason to exclude presentations with replicated inequations.

least upper bound $\bigvee_{i\in I} N_i$ of a family of vector addition systems is thus the vector addition system whose presentation is the union of the presentations of the N_i's. A vector addition system is termed *atomic* if it consists of only one place, then any vector addition system with set of places $P = \{p_i | i \in I\}$ is of the form $N = \bigvee_{i\in I} N_i$ where N_i is the atomic net associated with the place p_i. Observe that regions can be viewed as atomic vector addition system.

Dually if we make no distinction between automata that differ only in the identity of states (i.e. which are isomorphic: identical up to a bijective renaming of their sets of states), then the set $\mathbf{Aut}(E)$ of reversible automata with alphabet E is a complete lattice whose order relation is given by

$$A \sqsubseteq A' \Leftrightarrow \exists f : S \to S' \quad [f(s_0) = s'_0 \quad \wedge \quad s_1 \stackrel{e}{\to} s_2 \Rightarrow f(s_1) \stackrel{e}{\to} f(s_2)]$$

Indeed such a mapping f when it exists is unique (because A and A' are reversible automata), and this relation is a preorder whose induced equivalence is the isomorphism of automata. The greatest lower bound of reversible automata is given by their synchronized product. We remind the reader that the synchronized product $\bigwedge_{i\in I} A_i$ of a family of automata $A_i = (S_i, E, T_i, s_{0,i})$ indexed by $i \in I$ is the automaton (S, E, T, s_0) with components as follows: $s_0 = (s_{0,i})_{i\in I}$, S is the connected component of s_0 w.r.t. the *synchronized transition rule*

$$(s_i)_{i\in I} \stackrel{e}{\to} (s'_i)_{i\in I} \quad \textbf{iff} \quad \forall i \in I \quad (s_i \stackrel{e}{\to} s'_i) \in T_i$$

and T is the set of occurrences of this rule at states $(s_i)_{i\in I} \in S$.

We recall that the state graph N^* of a vector addition system associated with the polyhedron $P = \{x \in \mathbb{Z}^n | \mu \cdot x + b \geq 0\}$ where $\mu \in \mathbb{Z}^{m,n}$ and $b \in \mathbb{N}^n$ is the automaton with set of states $S = \{y \in \mathbb{N}^m | \exists x \in P \quad y = \mu \cdot x + b\}$ with initial sate $s_0 = b$ and transition relation $y \stackrel{e_i}{\to} y' \Leftrightarrow y' = y + \mu \cdot e_i$. By definition of state graphs of vector addition systems $N^* = \bigwedge_{1\leq i\leq m} N_i^*$ where $N = \bigvee_{1\leq i\leq m} N_i$ is the decomposition of N into atomic elements. The above remarks can be extended to the case where A is a matrix with an infinite number of rows, i.e. when N is a vector addition system with infinitely many places. Dually we let $A^* = \bigvee\{N | N \text{ is a region of } A\}$ stands for the vector addition system associated with the reversible automaton A.

Observation 4.2.2 *For any atomic vector addition system N, and reversible automaton A, $A \sqsubseteq N^*$ if and only if N is a region of A and thus:*

$$A \sqsubseteq N^* \Leftrightarrow N \sqsubseteq A^*$$

Proposition 4.2.3 *The two $()^*$ operators, mapping respectively the automaton A to the dual vector addition system A^* and the vector addition system N to its state graph N^*, constitute a Galois connection between the ordered sets $\mathbf{VAS}(E)$ and $\mathbf{Aut}(E)$: $A \sqsubseteq N^* \Leftrightarrow N \sqsubseteq A^*$ for $A \in \mathbf{Aut}(E)$ and $N \in \mathbf{VAS}(E)$.*

The relations $A_1 \sqsubseteq A_2 \Rightarrow A_2^* \sqsubseteq A_1^*$ (for $A_1, A_2 \in \mathbf{Aut}(E)$) and $N_1 \sqsubseteq N_2 \Rightarrow N_2^* \sqsubseteq N_1^*$ (for $N_1, N_2 \in \mathbf{VAS}(E)$) follow immediately from the Galois connection. Another property of Galois connections is to produce *closure operators* by composition of the dual operators. Recall that an operator $\overline{()}$ on $(X, \leq)$, mapping x to $\overline{x}$, is a closure operator if it is increasing ($x_1 \leq x_2 \Rightarrow \overline{x_1} \leq \overline{x_2}$), extensive ($x \leq \overline{x}$), and idempotent ($\overline{\overline{x}} = x$). The double dual operators $()^{**}$ acting respectively on the ordered sets $(\mathbf{Aut}(E), \sqsubseteq)$ and $(\mathbf{VAS}(E), \sqsubseteq)$ are therefore closure operators. A reversible automaton A equal to its closure A^{**} is said to be *separated*, while a vector addition system N equal to its closure N^{**} is said to be *saturated*. Owing to the Galois connection, the lattices of separated reversible automata and saturated vector addition systems are dually order-isomorphic (i.e. isomorphic up to reversing the order).

Thus a reversible automaton A is isomorphic to the state graph of a vector addition system if and only if it is separated and its dual is such a vector addition system (in fact the greatest one). However if A has any region, it will have an infinite number of regions and therefore its dual A^* is an infinite vector addition system. We shall see that any separated finite reversible automaton is isomorphic to the state graph of a finite vector addition system. For that purpose we introduce the notion of an *admissible set of regions* which is the analog of the homonymous notion introduced by Desel and Reisig [37] for elementary transition systems.

Definition 4.2.4 *a set $\mathcal{R}$ of regions of a reversible automaton is termed admissible if $A \cong \bigwedge\{N^* | N \in \mathcal{R}\}$, it is termed complete if $A^{**} \cong \bigwedge\{N^* | N \in \mathcal{R}\}$*

Thus when synthesizing a vector addition system from a reversible automata one can restrict the search space to any complete set of regions and stop the synthesis as soon as we have reach an admissible set of regions. The state graph N^* of a vector addition system is separated because $N^* \cong N^{***}$ follows from the Galois connection. In fact, the atomic components of N form an admissible set of regions for N^*. The convex hull of the commutative image of the language of A corresponds to a complete set of regions. Therefore minimal regions (which correspond to bording half spaces) constitute a complete set of regions. This observation is the counterpart of the fact noticed by Bernardinello [19] that the minimal regions of an elementary transition system is a complete set of regions.

4.2.3 Representation Result

Ehrenfeucht and Rozenberg gave a characterization of those automata which are isomorphic to the marking graphs of elementary nets in term of two *separation axioms* the first of which states that there exists sufficiently many regions to distinguish every pair of distinct states in the automaton. The second axiom of separation states that for every action and every state at which this action is not enabled there exists a region which "inhibits" this action in this state. Similarly the following criterion may be used to recognize admissible set of regions, and consequently separated automata. We recall that a region R of $A = (S, E, T, s_0)$ determines (and is determined by) two maps: the potential $R : S \to \mathbb{N}$ and the coboundary $R : E \to \mathbb{Z}$.

Proposition 4.2.5 *A set $\mathcal{R}$ of regions of a reversible automaton $A = (S, E, T, s_0)$ is admissible if and only if the following separation properties are satisfied for all state $s, s' \in S$, and for every event $e \in E$:*

- *(**ssp**): $s \neq s' \Rightarrow \exists R \in \mathcal{R} \quad Rs \neq Rs'$*
- *(**essp**):* $\begin{cases} s \stackrel{e}{\not\to} & \Rightarrow \quad \exists R \in \mathcal{R} \quad Rs + Re < 0 \\ \stackrel{e}{\not\to} s & \Rightarrow \quad \exists R \in \mathcal{R} \quad Rs - Re < 0 \end{cases}$

When both properties are satisfied, $A \cong (A^*_{\mathcal{R}})^*$, where $A^*_{\mathcal{R}}$ is the sub-system of A^* with restricted set of places $\mathcal{R}$, called the vector addition system synthesized from $\mathcal{R}$.

Indeed, $(A^*_{\mathcal{R}})^* \cong \bigwedge\{R^* | R \in \mathcal{R}\}$ and the morphism $A \to \bigwedge\{R^* | R \in \mathcal{R}\}$ that takes a state s to the vector $(R(s); R \in \mathcal{R})$ is injective if and only if A satisfies the state separation property **ssp** and it is a covering if and only if A satisfies the event-state separation property **essp**. Now isomorphisms coincide with the injective coverings.

Thus if we are looking for the representation of a reversible automaton by a vector addition system up to language equivalence we simply drop the separation property **ssp** in the above proposition. If we are looking for a representation up to the direct language equivalence (words labelling oriented paths in the automaton) then we further drop the second condition in **essp**.

4.2.4 A polynomial Time Algorithm

The set of coboundaries of a finite reversible automaton A, i.e. the set of vectors $u \in \mathbb{Z}^E$ such that $\forall v \in H_1(A) \quad u \cdot v = 0$, is the kernel of the linear transformation defined by the transpose of the relation matrix of the canonical group of A, $\mathcal{C}(A) = \mathbb{Z}^n / H_1(A)$. The algorithm of von zur Gathen and Sieveking (see [91]), given the relation matrix as input, produces in polynomial time a set of generators $\{u_1, \ldots, u_k\}$ for the group of co-boundaries. For instance the first homology group of the automaton depicted in Fig. 4.10 is generated by the vectors $a + b + c$ and $a' + b' + c'$. The group of co-boundaries thus consist of those vectors $u : E \to \mathbb{Z}$ such that:

$$u(a) + u(b) + u(c) = 0 \quad \text{and} \quad u(a') + u(b') + u(c') = 0$$

It is therefore the subgroup of $\mathbb{Z}^E$ generated by

$$u_1 = a - c \;\; ; \;\; u_2 = b - c \;\; ; \;\; u_3 = a' - c' \;\; \text{and} \;\; u_4 = b' - c'$$

We recall that, since A is finite, coboundaries are in bijective correspondence with minimal regions: the minimal region $R = \mathbf{region}(u, u_0)$ induced by the co-boundary u is such that $u_0 = \mathbf{min}\{-u \cdot \psi(u_s) | s \in S\}$, and $R(s) = u_0 + u \cdot \psi(u_s)$ and $R(e) = u(e)$. Moreover the minimal regions constitute a complete set of regions and

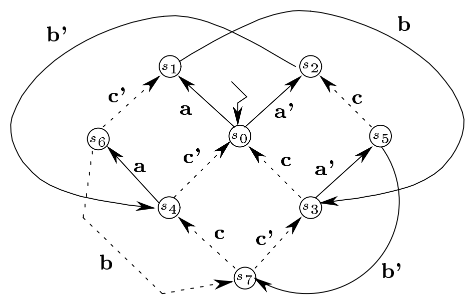


Figure 4.10: an automaton with one of its spanning trees (in solid lines)

Table 4.1: states $s \in S$ represented by vectors $(u_i \cdot \psi(u_s)_i$ indexed by the generating co-boundaries u_i

$u_i \cdot \psi(u_s)$	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7
u_1	0	1	0	1	0	1	1	1
u_2	0	0	0	1	0	1	0	1
u_3	0	0	1	0	1	1	1	1
u_4	0	0	0	0	1	0	1	1

Proposition 4.2.6 *The set of minimal regions induced by a set C of co-boundaries is an admissible set of regions if and only if the following separation properties are satisfied for all state $s, s' \in S$, and for every event $e \in E$:*

- *(**ssp**): $s \neq s' \Rightarrow \exists u \in C \quad u \cdot \psi(u_s) \neq u \cdot \psi(u_{s'})$*
- *(**essp**):* $\begin{cases} s \overset{e}{\not\to} & \Rightarrow \quad \exists u \in C \ \ \forall s' \in S \quad u \cdot \psi(u_s) + u \cdot e - u \cdot \psi(u_{s'}) < 0 \\ \overset{e}{\not\to} s' & \Rightarrow \quad \exists u \in C \ \ \forall s \in S \quad u \cdot \psi(u_s) + u \cdot e - u \cdot \psi(u_{s'}) > 0 \end{cases}$

As far as the first separation property is concerned, it suffices to check that every state s has a distinct representation as $\llbracket s \rrbracket = (u_i \cdot \psi(u_s); 1 \leq i \leq k)$ where $\{u_1, \ldots, u_k\}$ is a generating set of co-boundaries. For instance if we consider the previous example, the commutative images of the paths leading from s_0 to s in the spanning tree are

$$\begin{array}{llll} \psi(u_{s_0}) = 0 & \psi(u_{s_1}) = a & \psi(u_{s_2}) = a' & \psi(u_{s_3}) = a + b \\ \psi(u_{s_4}) = a' + b' & \psi(u_{s_5}) = a + b + a' & \psi(u_{s_6}) = a' + b' + a & \psi(u_{s_7}) = a + b \\ & & & \quad +a' + b' \end{array}$$

The corresponding scalar products $u_i \cdot \psi(u_s)$ are tabulated in Table 4.1 which shows that the first separation property is satisfied as all the columns are different.

Let $\{u_1, \ldots, u_k\}$ be a set of generating co-boundaries, and let $\alpha_i(s, e, s') = u_i \cdot \psi(u_s) + u_i(e) - u_i \cdot \psi(u_{s'})$. The second separation property then reads as follows:

If $s \stackrel{e}{\not\to}$ then find $u = \sum_{i=1}^{k} x_i \cdot u_i$ such that $\forall s' \in S \quad \sum_{i=1}^{k} \alpha_i(s,e,s') \cdot x_i < 0$; and similarly
if $\stackrel{e}{\not\to} s'$ then find $u = \sum_{i=1}^{k} x_i \cdot u_i$ such that $\forall s \in S \quad \sum_{i=1}^{k} \alpha_i(s,e,s') \cdot x_i > 0$.
Thus an instance of the second separation property write down as a system of linear inequations $MX < 0$ where $M \in \mathbb{Z}^{|S|,k}$. Now such a system may equivalently be written as the system

$$MX \leq (-1)^n \tag{4.1}$$

where $(-1)^n = (-1, \ldots, -1)$ $(\in \mathbb{Z}^n)$. Such a system has an integral solution if and only if it has a rational solution. The method of Khachiyan (see [91] p.170) may be used to decide upon the feasability of (4.1) and to compute a rational solution, if it exists, in polynomial time. Thus, every instance of the second separation property is solved up to a multiplicative factor, or shown unfeasible, in time polynomial in $|S|$ and $|E|$. In our running example, the system of linear inequations which expresses the separation problem associated with $s_2 \stackrel{a}{\not\to}$ is the following:

s	$\sum_i \alpha_i(s_2, a, s) \cdot x_i < 0$
s_0	$x_1 + x_3 < 0$
s_1	$x_3 < 0$
s_2	$x_1 < 0$
s_3	$x_3 - x_2 < 0$
s_4	$x_1 - x_4 < 0$
s_5	$-x_2 < 0$
s_6	$-x_4 < 0$
s_7	$-x_2 - x_4 < 0$

This system is solvable, and admits in particular the solution $x_1 = x_3 = -1$ and $x_2 = x_4 = 1$. Therefore, $u = -u_1 + u_2 - u_3 + u_4 = -a + b - a' + b'$ solves the corresponding instance of the separation problem. The automaton of Fig. 4.3 is separated: every instance of the separation properties can be solved by a co-boundary in Table. 4.2. The vector addition system synthesized from this set of regions is shown in Fig. 4.11.

Table 4.2: minimal regions $R_u = \textbf{region}(u, u_0)$ induced by separating set of co-boundaries u

u	a	b	c	a'	b'	c'	u_0
$-u_1$	-1	0	1	0	0	0	1
u_2	0	1	-1	0	0	0	0
$-u_3$	0	0	0	-1	0	1	1
u_4	0	0	0	0	1	-1	0
$u_1 - u_2$	1	-1	0	0	0	0	0
$u_3 - u_4$	0	0	0	1	-1	0	0
$u_2 + u_4 - u_1 - u_3$	-1	1	0	-1	1	0	1

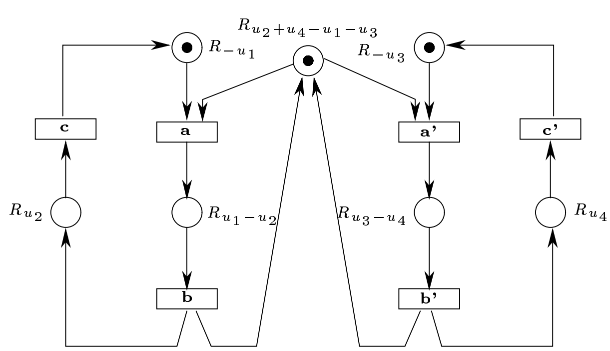


Figure 4.11: the net synthesized from the admissible set of minimal regions given in table 4.2

Chapter 5

Duality between Nets and Automata

In this chapter we describe dualities between nets and automata. These categorical dualities are induced by schizophrenic objects and are parametric on the type of nets. The advantage of this presentation is two-fold. First it shows that the region based dualities between nets and automata are close analogues of the classical representation theorems, like Birkhoff and Stone representation theorems, most of which arise from concrete dualities induced by schizophrenic objects. Second, it gives a uniform presentation for all (or almost all) dualities between nets and automata encountered in the litterature, in particular it can be extended to higher dimensional automata in order to take into account the concurrent behaviour of net systems.

5.1 Dual Adjunctions Induced by Schizophrenic Objects

The simplest example of duality induced by a schizophrenic object is Birkhoff's duality between finite distributive lattices and finite partial orders that we now recall. Assume a *schedule* is given as a finite ordered set of *tasks*, where $a \leq b$ means that task a should be performed before b. A *configuration* is a downward closed set of tasks: it consists of the tasks that have been performed in a particular state of the system. This set of configurations ordered by inclusion is a finite distributive lattice since meet and join are given by the set-theoretic union and intersection. We call this lattice the lattice of configurations of the ordered set. The problem is now to know whether a given finite distributive lattice is isomorphic to the lattice of configurations of some finite ordered set. The *extension* $[a[$ of a task a in the lattice of configurations, i.e. the set of configurations in which that task is reported, is a prime filter of the lattice of configurations (see Fig. 5.1). Indeed, it is a filter (non empty upper-set closed by meet) because the

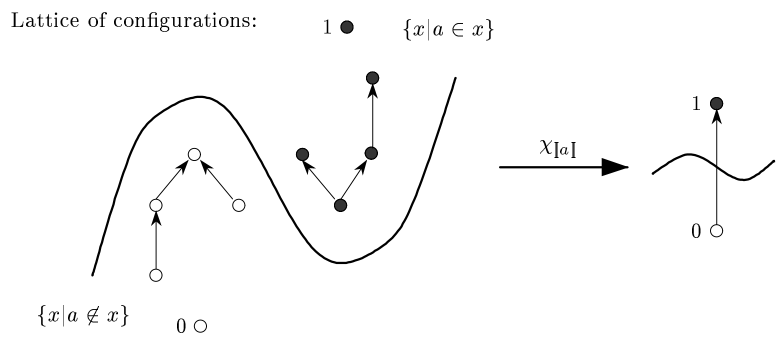


Figure 5.1: prime filters as tokens of the representation

whole set belongs to $\llbracket a \llbracket$, $a \in x \wedge x \subseteq y \Rightarrow a \in y$ and $[a \in x \wedge a \in y] \Rightarrow a \in x \cap y$. By symmetry its complement is an ideal (non-empty down-set closed by join), thus it is a prime filter. Moreover $a \leq b \Rightarrow \llbracket a \rrbracket \supseteq \llbracket b \rrbracket$, thus a candidate for representing a finite distributive lattice is the set of its prime filters ordered by reverse inclusion. Let us call this ordered set the schedule of the lattice. Birkhoff's theorem asserts that

> *any finite ordered set is isomorphic to the schedule of its lattice of configurations and any finite distributive lattice is isomorphic to the lattice of configurations of its schedule.*

Birkhoff's duality between finite distributive lattices and finite partial orders relies on the schizophrenic object $\mathbf{2} = \{0; 1\}$, viewed as a lattice and as an ordered set where $0 \leq 1$. The dual L^* of a distributive lattice L, i.e. its schedule, is the ordered set of its prime filters F whose characteristic functions are the lattice morphisms $\chi_F : L \to \mathbf{2}$. The dual E^* of an ordered set E, i.e. its lattice of configurations, is the lattice of its downwards closed subsets x whose characteristic functions are the morphisms of ordered sets $\chi_x : E \to \mathbf{2}$.

The dual adjunction more precisely asserts that for any ordered set E and finite distributive lattice L the set of monotone maps from E to L^* is in bijective correspondence with the set of lattice morphisms from L to E^*. In fact they are both in bijective correspondence with the relations $\models \subseteq E \times L$ such that

$$\begin{array}{l} a \not\models 0 \qquad a \models 1 \\ a \models x \wedge y \quad \Leftrightarrow \quad (a \models x \text{ and } a \models y) \\ a \models x \vee y \quad \Leftrightarrow \quad (a \models x \text{ or } a \models y) \\ (a \leq b \text{ and } b \models x) \quad \Rightarrow \quad a \models x \end{array}$$

These conditions are indeed equivalent to the fact that the map $a \mapsto \{x | a \models x\}$ is a monotone map from E to L^* or to the fact that the map $x \mapsto \{x | a \models x\}$ is a lattice morphism from L to E^*.

If $L = E^*$ is the set of configurations of E, the ordered set E is isomorphic to its double dual $E \cong E^{**}$ where $a \in E$ is identified with $ev_a \in E^{**}$ such that $\chi_{ev_a}(x) = \chi_x(a)$ for every down-set $x \in E^*$. Symmetrically, if $E = L^*$ the lattice L is isomorphic to its double dual $L \cong L^{**}$ where $x \in L$ is identified with $ev_x \in L^{**}$ such that $\chi_{ev_x}(F) = \chi_F(x)$ for every prime filter $F \in L^*$. Thus both units of the dual adjunction are morphisms whose underlying maps are the evaluation maps.

Birkhoff's duality between finite ordered sets and finite distributive lattices is an instance of concrete dualities induced by schizophrenic objects that we now describe.

Definition 5.1.1 *A* Set-category *(or category over* **Set***) is a pair* $<\mathcal{C}, U>$ *where* $\mathcal{C}$ *is a category and* $U : \mathcal{C} \to \mathbf{Set}$ *is a functor called the underlying functor. It is a* concrete category *if* U *is faithful.*

In the sequel, the underlying functor is left implicit and we use the uniform notation $|C|$ and $|f|$ for respectively the underlying set of an object C and the underlying map of an arrow f. In a Set-category $\mathcal{C}$, a *structured source* is an indexed family of pairs $\{C_i; f_i : X \to |C_i|\}$, where the C_i's are objects of $\mathcal{C}$ and the f_i's are maps from a fixed set X to the underlying sets of the C_i's. A *lift* of a structured source is an indexed family $\tilde{f}_i : C \to C_i$ of arrows of $\mathcal{C}$ such that $|\tilde{f}_i| = f_i$, and hence $|C| = X$. An *initial lift* of a structured source is a lift such that, if $g_i : C' \to C_i$ is another lift and there exists a map $f : |C'| \to X$ such that $|g_i| = f_i \circ f$ for all indices, then there exists a unique arrow $\tilde{f} : C' \to C$ such that $|\tilde{f}| = f$ and $g_i = f_i \circ \tilde{f}$ for all indices. The following definition is an adaptation from [81].

Definition 5.1.2 *A schizophrenic object between two Set-categories* $\mathcal{A}$ *and* $\mathcal{B}$ *is a pair of objects* $<K_\mathcal{A}, K_\mathcal{B}> \in |\mathcal{A}| \times |\mathcal{B}|$ *with the same underlying set* $K = |K_\mathcal{A}| = |K_\mathcal{B}|$ *and such that*

1. *for every object* A *in* $\mathcal{A}$*, the family* $\{K_\mathcal{B}; ev_A(a) : \mathcal{A}(A, K_\mathcal{A}) \to K\}_{a \in |A|}$ *of evaluation maps* $ev_A(a)(f) = |f|(a)$ *has an initial lift* $\{\epsilon_A(a) : A^* \to K_\mathcal{B}\}_{a \in |A|}$

2. *for every object* B *in* $\mathcal{B}$*, the family* $\{K_\mathcal{A}; ev_B(b) : \mathcal{B}(B, K_\mathcal{B}) \to K\}_{b \in |B|}$ *has an initial lift* $\{\epsilon_B(b) : B^* \to K_\mathcal{A}\}_{b \in |B|}$.

A^*, called the *dual* of A, is therefore an object of the category $\mathcal{B}$ whose underlying set is the set of $\mathcal{A}$-morphisms from A to the *classifying object* $K_\mathcal{A}$. If $K = \{0, 1\}$ and $\mathcal{A}$ is concrete, then the elements of the underlying set of the dual of A can be identified with subsets of the underlying set of A: $|A^*| \subseteq 2^{|A|}$ and $|A^{**}| \subseteq 2^{2^{|A|}}$. In any case, A and A^{**} are linked by an evaluation morphism $Ev_A : A \to A^{**}$ according to the following statement.

Lemma 5.1.3 *Let* $<K_\mathcal{A}, K_\mathcal{B}>$ *be a schizophrenic object between two Set-categories* $\mathcal{A}$ *and* $\mathcal{B}$*. The initial lift* $\{\epsilon_A(a) : A^* \to K_\mathcal{B}\}_{a \in |A|}$ *of the evaluation maps, viewed as a mapping* $\epsilon_A : |A| \to \mathcal{B}(A^*, K_\mathcal{B})$*, is the underlying map of an arrow* $Ev_A : A \to A^{**}$.

As an initial lift, the dual A^* of A is only defined up to an isomorphism. However, once an arbitrary representative A^* is fixed for each class of isomorphic objects, the operator $(-)^*$ gives rise to a functor according to the following statement.

Lemma 5.1.4 *Let* $<K_{\mathcal{A}}, K_{\mathcal{B}}>$ *be a schizophrenic object between two Set-categories* $\mathcal{A}$ *and* $\mathcal{B}$. *For every morphism* $f : A_1 \to A_2$ *in* $\mathcal{A}$, *the map "composing with* f*" given by* $f^\bullet : \mathcal{A}(A_2, K_{\mathcal{A}}) \to \mathcal{A}(A_1, K_{\mathcal{A}})$: $g \mapsto g \circ f$ *is the underlying map of an arrow* $f^* : A_2^* \to A_1^*$ *in* $\mathcal{B}$ *such that the functoriality laws* $(1_A)^* = 1_{A^*}$ *and* $(f \circ g)^* = g^* \circ f^*$ *are satisfied.*

The following proposition tells us that the two functors $(-)^*$ induced from a schizophrenic object are in fact dual adjoints.

Proposition 5.1.5 *Let* $<K_{\mathcal{A}}, K_{\mathcal{B}}>$ *be a schizophrenic object between two Set-categories* $\mathcal{A}$ *and* $\mathcal{B}$. *The following identities, where* $f : A \to B^*$ *and* $g : B \to A^*$, *define a bijective correspondence* $\mathcal{A}(A, B^*) \cong \mathcal{B}(B, A^*)$:

$$g = f^* \circ Ev_B \quad \text{and} \quad f = g^* \circ Ev_A$$

i.e. the functors $(-)^*$ *are adjoint to the right with the evaluations as units.*

In the particular case where $\mathcal{A}$ and $\mathcal{B}$ are concrete categories, the above correspondence may be presented as matrix transposition. Actually, in this special case, $\mathcal{A}(A, B^*) \cong \mathbf{Span}_K(A, B) \cong \mathcal{B}(B, A^*)$ where $\mathbf{Span}_K(A, B)$ is the set of matrices $|A| \times |B| \to K$ whose rows are underlying maps of morphisms $\varphi_a : B \to K_{\mathcal{B}}$ (for $a \in |A|$), or equivalently whose columns are underlying maps of morphisms $\varphi^b : A \to K_{\mathcal{A}}$ (for $b \in |B|$). In such a matrix, the set of rows determines a unique morphism from A to B^*, and the set of columns φ^b determines a unique morphism from B to A^*.

We can turn the set of $\mathcal{K}$-spans into a category whose objects are triples (A, φ, B) where $\varphi \in \mathbf{Span}_{\mathcal{K}}(A, B)$ (equivalently (A, f, B) where $f \in \mathcal{A}(A, B^*)$ or $(A, f^\sharp, B)$ where $f^\sharp \in \mathcal{B}(B, A^*)$) and whose morphisms are pairs of *reindexing* morphisms $\alpha \in \mathcal{A}(A_1, A_2)$, and $\beta \in \mathcal{B}(B_2, B_1)$ such that

- $\forall a \in A_1 \ \ \forall b \in B_2 \quad \varphi_1^{|\beta| b} = \varphi_2^b \circ \alpha$ and $(\varphi_2)_{|\alpha| a} = (\varphi_1)_a \circ \beta$,
- or equivalently $\beta^* \circ f_1 = f_2 \circ \alpha$,
- or equivalently $\alpha^* \circ f_2^\sharp = f_1^\sharp \circ \beta$.

For concrete categories this condition on morphisms reduces to :

$$\forall a \in A_1 \ \ \forall b \in B_2 \quad \varphi_1(a, |\beta| b) = \varphi_2(|\alpha| a, b)$$

Now the projection $\pi_1 : \mathbf{Span}_{\mathcal{K}} \to \mathcal{A}$ has a right-inverse left-adjoint $\rho_1 : \mathcal{A} \to \mathbf{Span}_{\mathcal{K}}$ which takes an object A of $\mathcal{A}$ to the unit $Ev_A : A \to A^{**}$ and takes the arrow $f : A \to A'$ to the pair (f, f^*). Symmetrically, the second projection $\pi_2 : \mathbf{Span}_{\mathcal{K}} \to \mathcal{B}^{op}$ has a right-inverse right-adjoint $\rho_2 : \mathcal{B}^{op} \to \mathbf{Span}_{\mathcal{K}}$ which takes an object B of $\mathcal{B}$ to the unit $Ev_B : B \to B^{**}$ and takes the arrow $g : B' \to B$ to the pair (g^*, g). The original dual adjunction $()^* \dashv (()^*)^{op} : \mathcal{A} \to \mathcal{B}^{op}$ can be recovered as the composite of the adjunctions $\rho_1 \dashv \pi_1$ (a co-reflection) and $\pi_2 \dashv \rho_2$ (a reflection). $\mathcal{A}$ and $\mathcal{B}^{op}$ are co-reflective (respectively reflective) full subcategories of $\mathbf{Span}_{\mathcal{K}}$. The *kernel* of the adjunction is the full subcategory consisting of those spans $\varphi \in \mathbf{Span}_{\mathcal{K}}(A, B)$ such that $B \cong A^*$ and $A \cong B^*$, it is equivalent to the respective full subcategories of $\mathcal{A}$ and $\mathcal{B}^{op}$ consisting of those objects for which units are isomorphisms. This yields a duality between the respective subcategories of $\mathcal{A}$ and $\mathcal{B}$.

The notion of span appear in the literature under various names. G. Birkhoff [23] call them *polarities*, but this term is sometimes used as a synonymous of Galois correspondence. S. Vickers [102] call *Topological Systems* the spans associated with the dual adjunction between topological spaces and frames. R. Wille [103] (see also [34]) considers that a concept has a dual nature; its extent is the set of all objects which exhibit the concept and its intent (or meaning) is the set of all properties shared by these objects. R. Wille call a *context* a span between instances and meaning. Spans are also called *Chu spaces* [82] or *classifications* [17].

5.2 Galois Connections

Birkhoff's dual adjunction between finite ordered sets and finite distributive lattice is a duality (i.e. the kernel of the adjunction is isomorphic to the whole of each category). This is not generally the case and it may be difficult to identify the kernel. An interesting case of dual adjunction (not necessarily induced by a schizophrenic object) is when the kernel coincides with the respective images of both categories, more precisely we recall the following:

Definition 5.2.1 (Galois Connections) *Let $\mathcal{A}(A, B^*) \cong \mathcal{B}(B, A^*)$ be a dual adjunction with units $\{E_A : A \to A^{**}\}_{A \in |\mathcal{A}|}$ and $\{E_B : B \to B^{**}\}_{B \in |\mathcal{B}|}$. We further let $\mathcal{B}^*$ (the image of $\mathcal{B}$) denote the full subcategory of $\mathcal{A}$ consisting of those objects B^* for $B \in |\mathcal{B}|$. The image of $\mathcal{A}$ is defined similarly. Then the dual adjunction is a* Galois Connection *whenever one of the following equivalent conditions is satisfied :*

1. *It restricts to a duality between the images: $\mathcal{B}^* \overset{op}{\cong} \mathcal{A}^*$,*
2. *the arrows $(E_A)^*$ are isomorphisms for $A \in |\mathcal{A}|$,*
3. *their left-inverses E_{A^*} are isomorphisms,*
4. *the arrows $(E_B)^*$ are isomorphisms for $B \in |\mathcal{B}|$,*
5. *their left-inverses E_{B^*} are isomorphisms,*
6. *the arrows $\{E_A : A \to A^{**}\}_{A \in |\mathcal{A}|}$ constitute a reflection of $\mathcal{A}$ into $\mathcal{B}^*$,*
7. *the arrows $\{E_B : B \to B^{**}\}_{B \in |\mathcal{B}|}$ constitute a reflection of $\mathcal{B}$ into $\mathcal{A}^*$.*

For instance the dual adjunction between topological spaces and frames is a Galois correspondance induced by the schizophrenic object $\mathbf{2}$, viewed as a boolean algebra and as a discrete topological space [58, 34]. Recall that a frame is a complete lattice with the generalized distributivity law (finite meets distribute over arbitrary joins: $f \wedge \bigvee_i f_i = \bigvee_i (f \wedge f_i)$). For any frame F, let $pt(F)$ be the set of *points* x of F defined as frame morphisms $x : F \to \mathbf{2}$. The dual F^* of F is the topological space $(pt(F),\Omega)$ whose open sets are the sets $O_f = \{x : F \to \mathbf{2} | \ x(f) = 1\}$ for f ranging over F. Conversely, the dual X^* of a topological space (X,Ω) is the frame of its open sets $O \in \Omega$, whose characteristic functions χ_O are the continuous maps from (X,Ω) to the Sierpinski space $\mathbf{2}$ (with open sets $\{0,1\}$, $\{1\}$, and $\emptyset$). Frames and topological spaces are connected by a dual adjunction $\mathbf{Frame}(F,X^*) \cong \mathbf{Top}(X,F^*)$.

By restricting this adjunction at both sides on its kernel, one obtains a duality $\mathbf{Top}^* \overset{op}{\cong} \mathbf{Frame}^*$ between the subcategory $\mathbf{Top}^*$ of *spatial* frames and the subcategory $\mathbf{Frame}^*$ of *sober* spaces. So, a frame F is isomorphic to its double dual F^{**} if and only if F is a spatial frame. Now, spatial frames are characterized by two conditions very similar to our separation conditions for automata, when regions are replaced by morphisms $x : F \to \mathbf{2}$. Namely, a frame F is spatial if and only if the following conditions are satisfied for all $f, f' \in F$, where $f \leq f' \Leftrightarrow f = f \wedge f'$:

$$\begin{array}{lccl} \textit{(i)} & f \neq f' & \Rightarrow & \exists x : F \to \mathbf{2} \ : \ x(f) \neq x(f') \\ \textit{(ii)} & f \not\leq f' & \Rightarrow & \exists x : F \to \mathbf{2} \ : \ x(f) = 1 \ \wedge \ x(f') = 0 \end{array}$$

Condition *(i)* is the analogue of our state separation condition SSP. Condition *(ii)* is the counterpart of our event state separation condition ESSP, when the structure of labelled transition system is replaced by the structure of partial order.

We say that a dual adjunction is of *depth* 0 if it is a duality, and it is of depth $n+1$ if its restriction to the images is a dual adjunction of depth n. For instance, a Galois connection is a dual adjunction of depth 1.

5.3 Dual Adjunction between Nets and Transition Systems

The classical dualities recalled above are concerned with points $x \in X$, properties $p \in P$, and a binary relation $\models \subseteq X \times P$ valued in the underlying set of the schizophrenic object, i.e. $\{0,1\}$. When this relation is given a matrix form, duality appears as matrix transposition. Dualities between transition systems and nets fit exactly in the same pattern: the points are the transitions $s \xrightarrow{e} s'$, the properties are the regions (σ,η), and the evaluation matrix given by $ev(s \xrightarrow{e} s', (\sigma,\eta)) = \sigma(s) \xrightarrow{\eta(e)} \sigma(s')$ describes the local effect of the transitions on the places (σ,η) of the dual net. This construction is parametric on the *type of nets* [10].

5.3.1 Types of Nets

Definition 5.3.1 *A type of nets is a deterministic transition system $\tau = (LS, LE, LT)$, where LS and LE are the respective sets of local states and local events, and $LT \subseteq LS \times LE \times LS$ defines the partial action of local events on local states.*

Definition 5.3.2 *A net of type τ is a triple $N = (P, E, W)$ where P is a set of places, E is a set of events, and $W : P \times E \to LE$ is the weight matrix. A marking is a mapping $M : P \to LS$. A net system of type τ is a structure $\mathcal{N} = (P, E, W, M_0)$ where M_0, the initial marking, is a marking of the underlying net $N = (P, E, W)$.*

A net or net system is place simple if all rows of the weight matrix are different; it is event simple if all columns of the weight matrix are different.

A net may be seen as an undirected complete bipartite graph whose edges are weighted by local events. As such, nets are of a static nature, but types (of nets) define their dynamics: the partial actions of events on markings may be inferred from the partial actions of local events on local states, using the weight matrix to control products of local events. The following definition extends in this way the usual *sequential firing rule.*

Definition 5.3.3 *Given a net $N = (P, E, W)$, of type $\tau = (LS, LE, LT)$, the (sequential) marking graph of N is the transition system $N^* = (LS^P, E, T)$ with set of transitions T defined by $(M \xrightarrow{e} M') \in T$ if and only if $\forall p \in P \;\; (M(p) \xrightarrow{W(p,e)} M'(p)) \in LT$. Given a net system $\mathcal{N} = (P, E, W, M_0)$, the (sequential) marking graph of $\mathcal{N}$ is the (dual) automaton $\mathcal{N}^* = (S, E, T_S, M_0)$ where S is the connected component* [1] *of M_0 in T, and $T_S = T \cap (S \times E \times S)$.*

Thus an event has concession at marking M if and only if for every place p, the local event $W(p, e)$ is enabled at the local state $M(p)$ in the transition system τ (defining the type of the net). Elementary nets are nets of type τ_{EN} shown in Fig. 5.2. Vector addition systems are nets of type $\tau_{vas} = (\mathbb{N}, \mathbb{Z}, T)$ such that

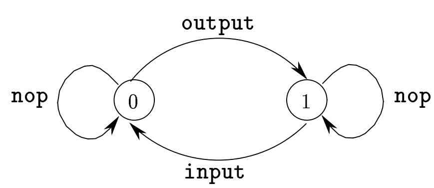


Figure 5.2: the type τ_{EN} of elementary nets

$n \xrightarrow{z} n'$ if and only if $n' = n + z$, i.e. τ_{vas} is the full subgraph of the Cayley graph of $\mathbb{Z}$ induced by the restriction on the subset of nodes in $\mathbb{N}$. The type of Petri nets is the transition system $\tau_{PN} = (\mathbb{N}, \mathbb{N} \times \mathbb{N}, T)$ such that $n \xrightarrow{(p,q)} n'$ if and only if $n \geq p$ and $n' = (n - p) + q$. Petri nets are set in bijective correspondence with nets of type τ_{PN} by the relation $W(p, e) = (F(p, e), F(e, p))$. With this correspondence, the firing rule stated in Def. 5.3.3 reads actually as

$$M[e{>}M' \;\;\Leftrightarrow\;\; \forall p \in P \;\; M(p) \geq F(p, e) \;\wedge\; M'(p) = M(p) - F(p, e) + F(e, p)$$

[1] we could instead have consider the inductive closure of $\{M_0\}$ w.r.t. forward transitions in T, however since we are mainly interested in the representation of reversible automata by nets we prefer to consider the forward and backward closure relation

The above Petri nets are a particular instance of the generalized Petri nets studied in [38]. In this paper, Droste and Shortt parametrize the classical definition of Petri nets, in which $\mathbb{N}$ is substituted for by the positive part G^+ of a partially ordered abelian group G. These authors further classify types of Petri nets over a fixed group G by the set of pairs $((F(p,e),F(e,p)) \in G^+ \times G^+$ occurring in associated subclasses of nets. For instance, condition-event nets are obtained by restricting nets over $\mathbb{Z}$ to the pairs $((F(p,e),F(e,p)) \in \{(0,0),(0,1),(1,0),(1,1)\}$. Note that the types of nets which we have considered so far can similarly be obtained as induced subgraphs of Cayley graphs. For instance, τ_{EN} is the Cayley graph of $\mathbb{Z}/_{3\mathbb{Z}}$ restricted on nodes 0 and 1, with $\texttt{nop} = 0$, $\texttt{output} = 1$, and $\texttt{input} = 2$.

5.3.2 Regions as Morphisms

The firing rule for nets stated in Def. 5.3.3 tells us that for every place p in a net of type τ, the pair of maps (σ_p, η_p) defined by $\sigma_p(M) = M(p)$ and $\eta_p(e) = W(p,e)$ is a morphism of transition systems from the marking graph of the net to the type τ. Therefore, if we forget the internal structure of states in the marking graph, identified with any isomorphic transition system (S,E,T), and if we identify a place p with its extension (σ_p, η_p), we can rediscover the places of the net (and also discover implicit places) as morphisms $(\sigma,\eta) : (S,E,T) \to \tau$. This motivates the following definition of regions for arbitrary types of nets.

Definition 5.3.4 *Given a transition system* $\mathbf{T} = (S,E,T)$ *and a type of nets* $\tau = (LS, LE, LT)$*, the set* $\mathcal{R}_\tau(\mathbf{T})$ *of* τ*-type regions in* $\mathbf{T}$ *is the set of morphisms from* $\mathbf{T}$ *to* τ*.*

As already noticed in Chapter 2 (see Fig. 2.1) regions introduced by Ehrenfeucht and Rozenberg indeed coincide with morphisms of transition systems from $\mathbf{T}$ to τ_{EN}. Similarly, generalized regions (see (4.2.1)) coincide with morphisms of transition systems from $\mathbf{T}$ to τ_{vas}, see Fig. 5.3.

5.3.3 Schizophrenic Object

Let **Trans** be the category of transition systems (S,E,T) whose morphisms $(\sigma,\eta) : (S,E,T) \to (S',E',T')$ are the pairs of maps $\sigma : S \to S'$ and $\eta : E \to E'$ such that $s \stackrel{e}{\to} s'$ (in T) entails $\sigma(s) \stackrel{\eta(e)}{\to} \sigma(s')$ (in T'). **Trans** is a Set-category with forgetful functor $U : \mathbf{Trans} \to \mathbf{Sets}$ given by $U(S,E,T) = T$ and $U(\sigma,\eta)(s \stackrel{e}{\to} s')$ $= (\sigma(s) \stackrel{\eta(e)}{\to} \sigma(s'))$. Say that a family of applications $f_i : X \to X_i$ is *jointly monic* if $\forall i \quad f_i(x) = f_i(y) \Rightarrow x = y$, i.e. there exists an injection $j : X \hookrightarrow \prod_i X_i$ such that $f_i = \pi_i \circ j$ where $\pi_i : \prod_i X_i \to X_i$ is the i^{th} projection. **Trans** has an initial lift of a jointly monic structure source $\{(S_i,E_i,T_i); T \subseteq \prod_i T_i \stackrel{f_i}{\to} T_i\}$ given by the product $(\prod_i S_i, \prod_i E_i, T \subseteq \prod_i T_i)$.

Let **Nets** be the category of event-simple nets (P,E,W) of type τ, thus $W : P \times E \to LE$ has all columns distinct, where a morphism $(\beta,\eta) : (P,E,W) \to (P',E',W')$ is a pair of maps $\beta : P \to P'$ and $\eta : E' \to E$ such that $W(p,\eta(e')) =$

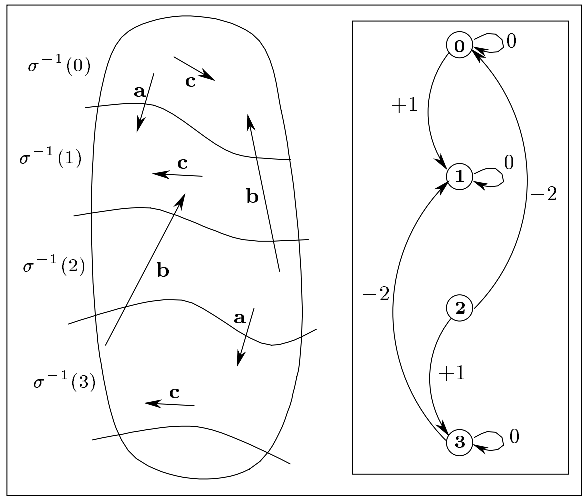


Figure 5.3: a generalized region as a morphism: $\mathbf{T} \to \tau_{vas}$

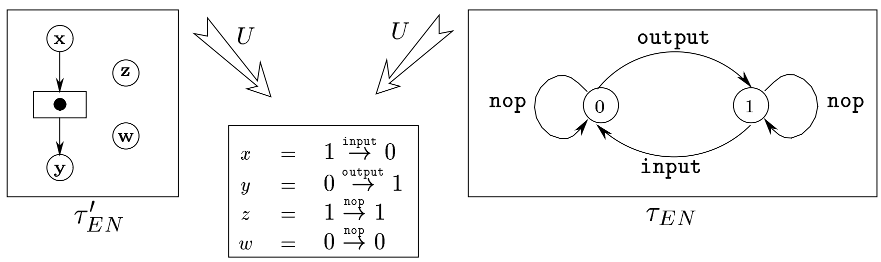


Figure 5.4: the schizophrenic object for elementary nets

$W(\beta(p), e')$. Owing to the assumption of event-simpleness, β determines η in any morphism (β, η), and **Nets** is a concrete category with forgetful functor $U : \mathbf{Nets} \to \mathbf{Sets}$ given by $U(P, E, W) = P$ and $U(\beta, \eta) = \beta$. **Nets** has initial lifts of arbitrary structure sources $\{(P_i, E_i, W_i); \beta_i : P \to P_i\}$ given by $(P, \coprod_i E_i, W)$ where $W : P \times E_i \to LE$ is the "coproduct" of the matrices W_i in the sense that

$$\forall p \in P \quad \forall a_i \in A_i \qquad W_i(\beta_i(p), a_i) = W(p, in_i(a_i))$$

Let $\tau' = (LT, \{\bullet\}, W) \in \mathbf{Nets}$ be the net with the unique event $\bullet$ such that $W(\ell s \stackrel{\ell e}{\to} \ell s') = \ell e$ for every place $\ell s \stackrel{\ell e}{\to} \ell s' \in LT$. Thus $U\tau' = LT = U\tau$. Figure 5.4 displays the net τ'_{EN} corresponding to the type τ_{EN} of elementary nets.

Proposition 5.3.5 *The pair (τ, τ') is a schizophrenic object between the categories* **Trans** *and* **Nets**, *inducing a dual adjunction* $\mathbf{Trans}(\mathbf{T}, N^*) \cong \mathbf{Nets}(N, \mathbf{T}^*)$.

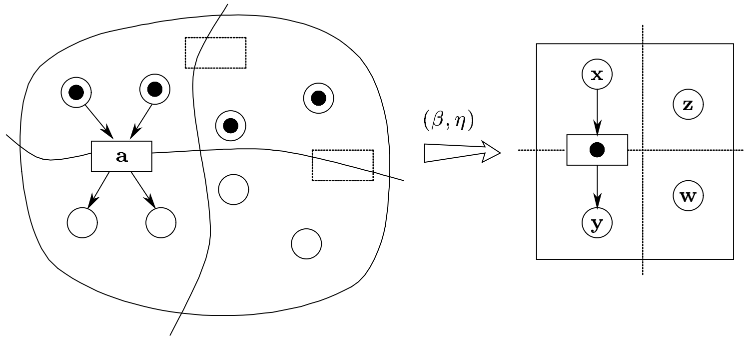


Figure 5.5: firings as net morphisms

It remains to interpret $\mathbf{T}^*$ and N^* in more familiar terms. For any transition system $\mathbf{T} = (S, E, T)$, the homset $\mathbf{Trans}(T, \tau)$ is the set $\mathcal{R}_\tau(\mathbf{T})$ of its τ-regions. The evaluation $ev_{\mathbf{T}}(s \stackrel{e}{\rightarrow} s')(\sigma, \eta) = (\sigma(s) \stackrel{\eta(e)}{\rightarrow} \sigma(s'))$ classifies therefore the transitions $t = (s \stackrel{e}{\rightarrow} s') \in T$ according to their local effect on each region. By definition, $\mathbf{T}^*$ is the net resulting from the initial lift of the family of evaluation maps $ev_A(t)$ for $t \in \mathbf{T}$. The following proposition shows that $\mathbf{T}^*$ coincides with the net synthesized from the set of regions $\mathcal{R}_\tau(\mathbf{T})$ up to the confusion of indiscernible events.

Proposition 5.3.6 $\mathbf{T}^*$ *is isomorphic to the net* $(P, E_\equiv, W)$ *where* $P = \mathcal{R}_\tau(\mathbf{T})$ *is the set of regions of* $\mathbf{T}$, $\equiv$ *is the equivalence relation on* E *such that* $e \equiv e'$ *when* $\eta(e) = \eta(e')$ *for every region* (σ, η), *and* $W((\sigma, \eta), [e]_\equiv) = \eta(e)$.

Now, for any net $N = (P, E, W)$, the homset $\mathbf{Nets}(N, \tau')$ is in bijective correspondence with the set of transitions of the marking graph of N. In the case of elementary nets (see Fig. 5.5), each morphism $(\beta, \eta) : N \rightarrow \tau'_{EN}$ induces the transition $\beta^{-1}(\{x, z\}) \xrightarrow{\eta(e)} \beta^{-1}(\{y, z\})$, and conversely, each firing $M[e\!>\!M'$ induces the morphism (β, η) such that $\eta(\bullet) = e$ and for every place p,

$$\beta(p) = \begin{cases} x & \textbf{if} \quad p \in M \setminus M' \\ y & \textbf{if} \quad p \in M' \setminus M \\ z & \textbf{if} \quad p \in M \cap M' \\ w & \textbf{if} \quad p \notin M \cup M' \end{cases}$$

Therefore, the evaluation $ev_N(p)(\beta, \eta) = \beta(p)$ classifies the places of N according to the local transition they undergo in each global firing of the net. By definition, N^* is the transition system resulting from the initial lift of the family of evaluation maps $ev_N(p)$ for $p \in P$.

Proposition 5.3.7 N^* *is isomorphic to the marking graph of* N.

5.4 Dual Adjunction between Nets Systems and Automata

The category **Aut** of automata is the category whose objects are pairs $A = (\mathbf{T}, s_0)$ where $\mathbf{T} = (S, E, T)$ is a connected transition system and $s_0 \in S$ is an initial state, and whose morphisms $(\sigma, \eta) : A \to A'$ are morphisms of transition systems such that $\sigma(s_0) = s'_0$. We let the dual of an automaton be the net system $A^* = (\mathbf{T}^*, M_0)$ whose initial marking is given by $M_0(p) = p(s_0)$ for every place $p \in \mathcal{R}_\tau(\mathbf{T})$ of $\mathbf{T}^*$. Net systems (of a given type) are turned into a category **NetSys** whose morphisms $(\beta, \eta) : (P, E, W, M_0) \to (P', E', W', M'_0)$ are the morphisms of the underlying nets (i.e. pairs of maps $\beta : P \to P'$ and $\eta : E' \to E$ such that $W(p, \eta(e') = W(\beta(p), e')$ for all $p \in P$ and $e' \in E'$) such that $M_0 = M'_0 \circ \beta$.

Proposition 5.4.1 *The dual adjunction between transition systems and nets induces a dual adjunction of depth* 2 *between automata and net systems. By restriction on its kernel it provides a duality* $\mathbf{Aut}^{**} \overset{op}{\cong} \mathbf{NetSys}^{**}$ *between separated automata and saturated net systems. An automaton is separated if, and only if, it satisfies the following separation properties where* $\mathcal{R}$ *is the set of regions of the underlying transition system:*

- *(***esp***):* $e \neq e' \Rightarrow \exists R = (\sigma, \eta) \in \mathcal{R} \quad \eta(e) \neq \eta(e')$
- *(***ssp***):* $s \neq s' \Rightarrow \exists R = (\sigma, \eta) \in \mathcal{R} \quad \sigma(s) \neq \sigma(s')$
- *(***essp***):* $\begin{cases} s \overset{e}{\not\to} \;\Rightarrow\; \exists R = (\sigma, \eta) \in \mathcal{R} \quad \sigma(s) \overset{\eta(e)}{\not\to} \; \text{in } \tau \\ \overset{e}{\not\to} s \;\Rightarrow\; \exists R = (\sigma, \eta) \in \mathcal{R} \quad \overset{\eta(e)}{\not\to} \sigma(s) \; \text{in } \tau \end{cases}$

Indeed if $\mathcal{N} = (N, M_0)$ and $A = (\mathbf{T}, s_0)$, since $\mathbf{T}$ is connected and $\mathcal{N}^*$ is the connected component of M_0 in N^* the morphisms $f : A \to \mathcal{N}^*$ coincide with the morphisms $f : \mathbf{T} \to N^*$ such that $f(s_0) = M_0$ which are in bijective correspondence with $f^\sharp : N \to \mathbf{T}^*$ such that $f^\sharp(p)(s_0) = M_0(p)$, i.e. the morphisms $f^\sharp : \mathcal{N} \to A^*$. Hence a derived dual adjunction $\mathbf{Aut}(A, \mathcal{N}^*) \cong \mathbf{NetSys}(\mathcal{N}, A^*)$. The evaluation map $Ev_A : A \to A^{**}$ represents a state $s \in S$ by the vector of local states $(\sigma(s); R = (\sigma, \eta) \in \mathcal{R})$, an event $e \in E$ by the vector of local events $(\eta(e); R = (\sigma, \eta) \in \mathcal{R})$, and a transition $t = s \overset{e}{\to} e'$ by the vector of local transitions $(\sigma(s) \overset{\eta(e)}{\to} \sigma(s'); R = (\sigma, \eta) \in \mathcal{R})$ hence it is an isomorphism if and only if the three separation properties are satisfied. Extensions of places of a net system are sufficient to establish the separation properties **ssp** and **essp** of its state graph but not necessarily **esp**. Thus $Ev_{\mathcal{N}^*}$ is not an isomorphism for every net system $\mathcal{N}$, i.e. the dual adjunction is not a Galois connection. However it holds for those nets whose state graphs verify **esp**:

Proposition 5.4.2 *An automaton verifying* **esp** *is isomorphic to the state graph of a net system if, and only if, it satisfies* **ssp** *and* **essp**.

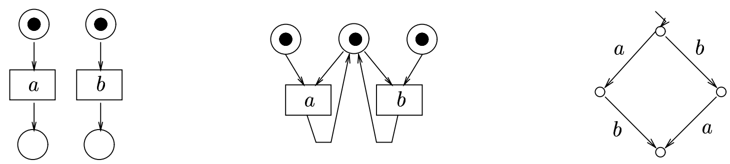


Figure 5.6: two safe Petri nets with identical marking graph but different independence relations

5.5 Extension to Higher Dimensional Automata

Nielsen and Winskel [73] and Mukund [67] solved the synthesis problem for respectively safe Petri nets and (general) Petri nets, thus allowing side conditions. With those impure nets the concurrency exhibited by the synthesized net is no longer reflected in the structure of the automaton. The simplest illustration of this phenomenon appears in Fig. 5.6 where the two safe Petri nets have the same marking graph shown on the right, but the actions a and b are independent in the first case and not in the second case. Actions in a safe net are independent if and only if they have disjoint domains, where the domain of an action is the set of conditions which are connected to it (i.e. its input and output conditions). The information about the independence of transitions in the marking graph of a safe net is then totally captured by a binary relation of independence on the set of actions (which are the labels of the transitions). Enriched with this relation, the marking graph becomes what is known as an *Asynchronous Transition System* [18, 93]. For the converse direction, Nielsen and Winskel have defined a variant of region, called *condition*, in asynchronous transition systems (with the restriction that for every pair of independent actions there exists at least one state in which these actions are both enabled). Conditions of such an asynchronous transition system constitute the places of a safe net of which the asynchronous transition system is the marking graph as soon as both separation axioms are satisfied. By supplying safe nets and asynchronous transition systems with adequate notions of morphisms they established a coreflection between the subcategory of asynchronous transition systems verifying the separation axioms and the category of safe nets.

Actions in a (general) Petri net are independent in a given marking if there exists enough tokens in their input places so that they can fire concurrently. For instance the three Petri nets in Fig. 5.7 have the same marking graph shown on the right; however the three actions a, b and c are independent in the first net (at the indicated marking) whereas they are pairwise independent but not independent in the second case. The third example (borrowed from [54]) is even more involved: at the indicated marking the maximal sets of independently enabled actions are $\{a, c\}$ and $\{b, c\}$; a and b are not independent but they become independent once action c has been fired. Thus the independence relation of actions is now state dependent; moreover even at a given marking the information about the independence of actions does not reduce to a binary

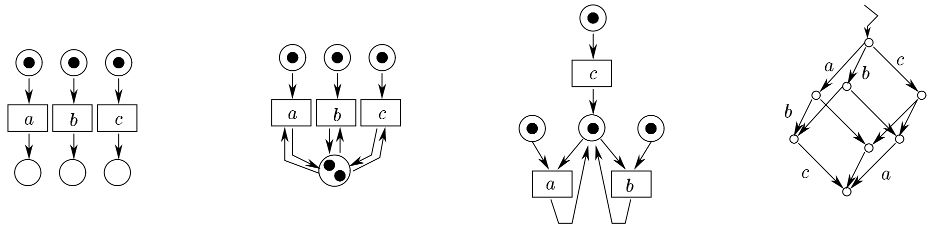


Figure 5.7: three Petri nets with identical marking graph but different independence relations

relation: actions as in the second example can be pairwise independent but not globally independent. Moreover it seems natural to allow *autoconcurrency*, i.e. several instances of the same action can fired concurrently if sufficiently many ressources are available in their input places. Enriched with the data of the multisets of actions independently enabled at each marking, the marking graph of a Petri net becomes a so-called *step transition system*. This counterpart of asynchronous transition systems introduced by Mukund in [67] are transition systems in which one state can be transformed to another one in a single step consisting of a finite multiset of concurrently enabled actions. Mukund has introduced a variant of regions in step transition systems that we shall called here *generalized conditions*. The generalized conditions of a step transition system constitute the places of a Petri net of which this step transition system is the marking graph as soon as both separation axioms are satisfied. By supplying Petri nets and step transition systems with morphisms between them, Mukund has obtained a coreflection between the subcategory of step transition systems verifying the separation axioms and the category of Petri nets.

Definition 5.5.1 *A* step transition system (S, M, T) *over an abelian monoid* M *consists of a set of states* S *and a deterministic transition relation* $T \subseteq S \times M \times S$*, with distinguished* empty steps*:* $s \stackrel{0}{\to} s'$ *iff* $s = s'$ *such that* $s_1 \stackrel{\alpha}{\to} s_2$*,* $s_2 \stackrel{\beta}{\to} s_3$*, and* $s_1 \stackrel{\alpha+\beta}{\to} s_4$ *implies* $s_3 = s_4$*. A* step automaton *is a connected step transition system* (S, M, T, s_0) *with an initial state* $s_0 \in S$*.*

This definition of step transition systems extends slightly Mukund's original definition, which was restricted to free abelian monoids. The extension allows to accomodate the idea of regions as morphisms to step transition systems which do not necessarily present the *intermediate state* property: $s \stackrel{\alpha+\beta}{\to} s' \Rightarrow \exists s'' \in S$ $s \stackrel{\alpha}{\to} s'' \wedge s'' \stackrel{\beta}{\to} s'$. The definition of regions in step transition systems is parametric on enriched types of nets defined as follows.

Definition 5.5.2 *An enriched type of nets is a (deterministic) step transition system* $\tau = (LS, LE, LT)$*, where* LE *is an abelian monoid* $(LE, +, 0)$*.*

For instance, the enriched type of Petri nets is just the type $\tau_{PN} = (\mathbb{N}, \mathbb{N} \times \mathbb{N}, LT)$, where $n \stackrel{(i,j)}{\longrightarrow} n' \in LT$ if and only if $n \geq i$ and $n' = n - i + j$, enriched with the operation of componentwise addition in $I\!N \times I\!N$. Each type of nets determines a specific concurrent firing rule and thus a specific construction of concurrent marking graphs.

Definition 5.5.3 *Given a net $N = (P, E, W)$ with (enriched) type $\tau = (LS, LE, LT)$, the concurrent marking graph of N is the step transition system $(LS^P, <E>, T)$ with set of transitions T defined by :*

$$(M \xrightarrow{\alpha} M') \in T \;\Leftrightarrow\; \forall x \in P \;\; (M(x) \xrightarrow{W(x,\alpha)} M'(x)) \in LT \tag{5.1}$$

where $W(x, e_1 + \cdots + e_n) = W(x, e_1) + \cdots + W(x, e_n)$. Given a net system $\mathcal{N} = (P, E, W, M_0)$, the concurrent marking graph of $\mathcal{N}$ is the step automaton $\mathcal{N}^* = (S, <E>, T_S, M_0)$ where S is the connected component of M_0 in T and $T_S = T \cap (S \times <E> \times S)$.

Definition 5.5.4 *A morphism of step transition systems from $\mathbf{T} = (S, M, T)$ to $\mathbf{T}' = (S', M', T')$ is a pair (σ, η), made of a map $\sigma : S \to S'$ and a monoid morphism $\eta : M \to M'$, such that $s \xrightarrow{\alpha} s' \Rightarrow \sigma(s) \xrightarrow{\eta(\alpha)} \sigma(s')$. The* morphisms of step automata *from A to A' are the morphisms from $\mathbf{T}$ to $\mathbf{T}'$ that preserve the initial state.*

Definition 5.5.5 *Given a step transition system $\mathbf{T} = (S, M, T)$ and an enriched type of nets $\tau = (LS, LE, LT)$, the set $\mathcal{R}_\tau(\mathbf{T})$ of τ-type regions of $\mathbf{T}$ is the set of morphisms of step transition systems from $\mathbf{T}$ to τ.*

By specializing this definition to the type τ_{PN}, one retrieves exactly the regions defined by Mukund in step transition systems over a free abelian monoid [67]. Special attention may be paid to the class of step transition systems $\mathbf{T} = (S, M, T)$ derived from asynchronous transition systems $(S, E, \|, T')$ as follows: $M = <E>$ is the free abelian monoid generated by E (the elements of M are finite multisets of elements of E), $s \xrightarrow{\alpha} s'$ in T if and only if α is a sub*set* of pairwise independent events $\{e_1, \ldots, e_n\} \subseteq E$ (hence there is no auto-concurrency) and there exists in T' a sequence of transitions $s \xrightarrow{e_1} s_1 \xrightarrow{e_2} s_2 \ldots s_{n-1} \xrightarrow{e_n} s_n$ such that $s' = s_n$ (such sequences exist therefore for all permutations of $\{e_1, \ldots, e_n\}$). For this class of step transition systems, the regions $(\sigma, \eta) : A \to \tau_{PN}$ which are *safe* in the sense that $\sigma(s) \in \{0, 1\}$ for all $s \in S$ are in bijective correspondence with the regions defined by Nielsen and Winskel for asynchronous transition systems [73].

If as in section 5.3.3 we let τ' denote the net whose places are the local transitions of τ,

Proposition 5.5.6 *The pair (τ, τ') is a schizophrenic object between the category* **STS** *of step transition systems and the category* **Nets** *of nets, inducing a dual adjunction* $\mathbf{STS}(\mathbf{T}, N^*) \cong \mathbf{Nets}(N, \mathbf{T}^*)$.

We have an analogue of Prop. 5.4.1 where events are replaced by steps (notice that the concurrent marking graph of a net is a step transition system over a *free* abelian monoid). Mukund's characterization of Petri net transition systems,

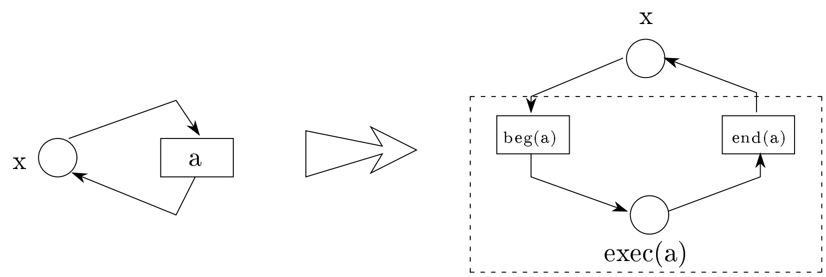


Figure 5.8: transformation of a net with side conditions into a pure one by splitting of actions

established in [67], follows directly from this proposition applied to the type τ_{PN}. Nielsen and Winskel's characterization of separated asynchronous automata, established in [73], follows therefrom as the subcase met when imposing on regions $(\sigma, \eta) \in \mathcal{R}$ the constraint that $\sigma(s) \in \{0,1\}$ for every state s. The synthesis of nets systems with inhibitor arcs from their concurrent state graphs [79, 80] is another instance of our construction.

It emerges from the discussion above that the behaviour of pure nets is given by ordinary automata (all information about concurrency is encoded in the structure of the automata) whereas the behaviour of (possibly) impure nets is given by *Higher-Dimensional Automata*. Higher-dimensional automata are geometrical models for automata with concurrency relations on transitions. There were first proposed by Pratt and van Glabbeek [83, 46] (see also [94]) and are extensively studied in Goubault's thesis [48]. An higher-dimensional automaton consists of states of various dimensions, where an n-dimensional state is interpreted as a situation in which n independent events are concurrently executed. An n-dimensional state of a step transition system over a free abelian monoid is a pair (s, α) where α is a step enabled in s. Such an automaton can be viewed as an ordinary automaton, called its *discretization*, defined on the splitted alphabet (each event a is splitted into the *begining of* a, $beg(a)$, and the *end of* a, $end(a)$) by $(s,\alpha) \stackrel{beg(a)}{\rightarrow} (s,\beta)$ and $(s,\beta) \stackrel{end(a)}{\rightarrow} (s',\alpha)$ where $s \stackrel{\alpha}{\rightarrow} s'$ and $\beta = \alpha + a$. Dicretization of higher-dimensional automata is closely connected to the usual transformation of an impure net into a pure one by splitting the actions of the original net (see e.g. [47]). This transformation is illustrated in Fig. 5.8: each input place of action a becomes an input place of $beg(a)$ with the same weight, similarly each ouput place of a becomes an output place of $end(a)$. The place $exec(a)$ which is an output place of $beg(a)$ and an input place of $end(a)$ (with single weights) witnesses the activity of action a in case of a safe net, or counts the number of instances of actions a which are currently executed in case of a (general) Petri net. A marking of the splitted net can then be viewed as an n-dimensional state where n is the total number of tokens in places of the form $exec(a)$. More precisely, the marking graph of the splitted net, which is an ordinary automaton on the splitted alphabet, appears as a discretization of the

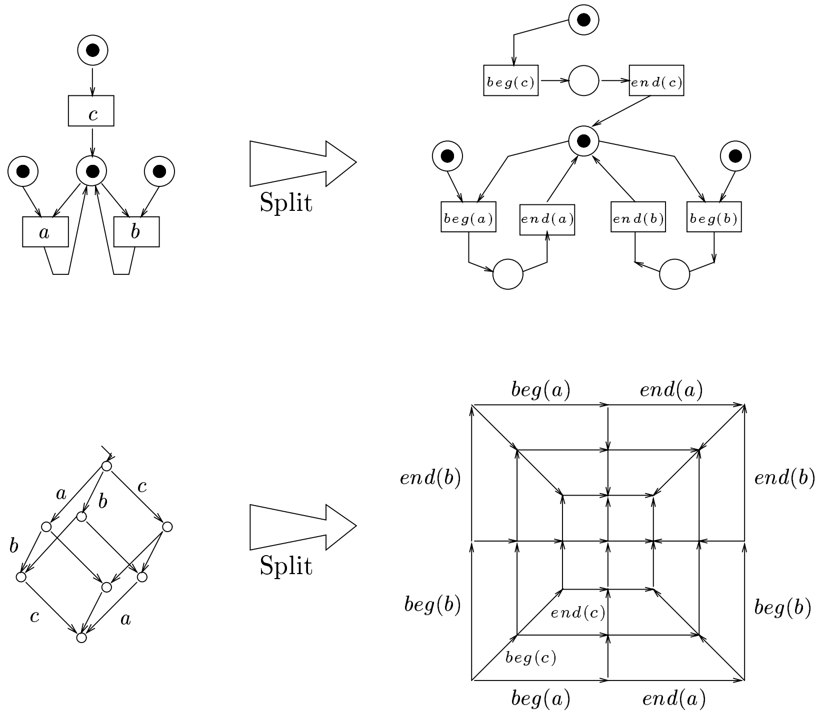


Figure 5.9: the marking graph of the splitted net as the discretization of the higher-dimensional automaton associated with the original net

higher-dimensional automaton associated with the original net (see Fig. 5.9). Moreover

Proposition 5.5.7 ([3]) *An asynchronous transition system is the behaviour of some safe Petri net (i.e. satisfies the separation axioms w.r.t. its set of conditions) if and only if its discretization is the marking graph of some elementary net system (i.e. is separated by regions). Similarly a step transition system is the behaviour of some Petri net (i.e. is separated by generalized conditions) if and only if its discretization is the marking graph of some pure Petri net (i.e. is separated by generalized regions).*

Chapter 6

Conclusion

Let me first mention some miscellaneous results that have not been presented in this document. The polynomial algorithm of [6], presented in Section 4.2.4, has been adapted to the synthesis of Petri net from its sequential or concurrent marking graph [12] to the synthesis of Petri nets from languages [6, 33] and to the synthesis of stratified Petri nets [13], a weaker form of Valk's self-modifying nets [100, 101]. In the survey [14] we hint some connections of the theory of regions with the cutset representation of graphs on which the synthesis of marked graphs [68] is founded. We also give a classification of safe nets, i.e. nets with set theoretic regions (local states are 0 and 1), among which one can find besides elementary net systems the *trace nets* of [8, 9] (equivalently the chart nets of [61]) and the *flip flop nets* of [89].

The theory of regions have come so far in two areas of application, the state encoding problem for asynchronous circuits using elementary regions [30] and the distribution of protocols based on generalized regions [28]. We may consider using regions in order to synthesize distributed controllers for discrete event systems and using the synthesis of stratified Petri nets for analysing cooperative systems in order to identify their various modes of operations.

Another goal of research is to consider relaxed versions of the synthesis problem where systems are derived from specifications given by languages or logical formulas rather than by an explicit automaton. In particular it could be useful to synthesize net systems from partial service specifications. In that direction [33] describes an algorithm that decides whether there exists a Petri net whose language contains a regular language expressing service requirements and is disjoint from another regular language expressing safety conditions.

The duality between automata and net systems illustrates the dual nature of dicrete event systems according to which an event may be described extensionally by its transition relation or intensionally by the way it modifies local properties. It is then tempting to view markings as complete theories for a logic algebra given by the places. For instance it is shown in [20] that the set of regions of an elementary transition system ordered by inclusion is an orthomodular poset whose ultrafilters correspond to markings. This is similar to the logic of theory change [1] in which a state is described by a maximal coherent theory (the analogue of an ultrafilter), a change is given by a modification of some properties and is treated as a revision function on theories: a theory is amended to account for the propositions being eliminated or added due to the occurrence of the event.

Bibliography

[1] Alchourrón, C.E., Gärdenfors, P., and Makinson, D., *On the Logic of Theory Change: Partial Meet Contraction and Revision Functions.* The Journal of Symbolic Logic, volume 50, Number 2 (1985) 510–530.

[2] Badouel, E., *Models of concurrency.* Fifth European Summer School in Logic Language and Information, Faculdade de Letras, Universidade de Lisboa, 1993.

[3] Badouel, E., *Splitting of Actions, Higher-Dimensional Automata and Net Synthesis.* Inria Research Report No 3013 (1996).

[4] Badouel, E., *Réseaux de Petri à structure dynamiques.* Ecole d'été MOVEP (modélisation et vérification des processus parallèles), Nantes (juillet 1998).

[5] Badouel, E., *Representations of Reversible Automata and State Graphs of Vector Addition Systems.* Inria Research Report no 3490 (1998).

[6] Badouel, E., Bernardinello, L. and Darondeau, Ph., *Polynomial algorithms for the synthesis of bounded nets*, Proceedings Caap 95, Lecture Notes in Computer Science 915 (1995) 647–679.

[7] Badouel, E., Bernardinello, L. and Darondeau, Ph., *The synthesis problem for elementary net systems is NP-complete*, Theoretical Computer Science 186 (1997) 107–134.

[8] Badouel, E., and Darondeau, Ph., *Trace Nets.* REX workshop, Beekbergen "Semantics: Foundation and Applications", Springer-Verlag Lecture Notes in Computer Science, vol. 666 (1993) 21–50.

[9] Badouel, E., and Darondeau, Ph., *Trace nets and process automata*, Acta Informatica 32 (1995) 647–679.

[10] Badouel, E., and Darondeau, Ph., *Dualities between Nets and Automata induced by Schizophrenic Objects*, 6^{th} International Conference on Category Theory and Computer Science, Cambridge, volume 953 of Lecture Notes in Computer Science (1995) 24–43.

[11] Badouel, E., and Darondeau, Ph., *A Survey on Net Synthesis*, Proceedings of cesa'96 imacs Multiconference, Lille, France (1996)

[12] Badouel, E., and Darondeau, Ph., *On the Synthesis of General Petri Nets.* Inria Research Report No 3025 (1996).

[13] Badouel, E., and Darondeau, Ph., *Stratified Petri Nets.* Proceedings of fct'97, volume 1279 of Lecture Notes in Computer Science, Springer Verlag (1997) 117–128.

[14] Badouel, E., and Darondeau, Ph., *Theory of Regions.* Third Advance Course on Petri Nets, Dagstuhl Castle. To appear in the Lecture Notes in Computer Science (1997).

[15] Badouel, E., Darondeau, Ph., and Raoult, J.-C., *Context-Free Event Domains are Recognizable.* Proceedings Amast 95 (Algebraic Methodology and Software Technology), Montréal. volume 936 of Lecture Notes in Computer Science (1995) 214–229. To appear in *Information and Computation.*

[16] Badouel, E., and Oliver, J., *Reconfigurable Nets, A Class of High Level Petri Nets Supporting Dynamic Changes.* Proceedings of the workshop *Workflow Management: Net-based Concepts, Models, Techniques and Tools*, Lisbon (1998).

[17] Barwise, J., and Seligman, J., *Information Flow, the Logic of Distributed Systems.* Cambridge Tracts in Computer Science 44, Cambridge University Press (1997).

[18] Bednarczyk, M. A., *Categories of Asynchronous Systems.* Ph. D. Thesis, University of Sussex (1988).

[19] Bernardinello, L., *Synthesis of Net Systems.* Application and Theory of Petri Nets, Lecture Notes in Computer Science 691 (1993) 89–105.

[20] Bernardinello, L., *Propriétés algébriques et combinatoires des régions dans les graphes, et leurs applications à la synthèse de réseaux.* Thèse Université de Rennes I (1998).

[21] Bernardinello, L., De Michelis, G., Petruni, K., and Vigna, S., *On the Synchronic Structure of Transition Systems*, in J. Desel (Ed.), Structures in Concurrency Theory (strict), May 1995, Springer (1996) 11–31.

[22] Berthelot, G., *Transformations and Decomposition of Nets.* In [24] 359–376.

[23] Birkhoff, G.D., *Lattice Theory.* Colloquium Publications. American Mathematical Society, Providence, R.I. (1940).

[24] Brauer, W., Reisig, W., and Rozenberg, G., (eds): *Petri Nets: Central Models and their Properties, Advances in Petri Nets 1986, Part I*, Lecture Notes in Computer Science vol. 254 (1987)

[25] Bryant, R., *Graph-Based Algorithms for Boolean Function Manipulation.* ieee Transactions on Computers, vol. C-35, no. 8 (1986) 677–691.

[26] Bryant, R., *Symbolic boolean manipulation with ordered binary decision diagrams.* acm Computing Surveys, vol. 24, no. 3 (1992) 293–318.

[27] Busi, N., and pinna, G.M., *Synthesis of nets with inhibitor arcs.* Proceedings of Concur'97, Lecture Notes in Computer Science vol. 1243 (1997), 151–165.

[28] Caillaud, B., synet : *un outil de synthèse de réseaux de Petri bornés, applications* Irisa Research Report no 1101 (1997).

[29] Cortadella, J., Kishinevsky, M., Lavagno, L., and Yakovlev, A., *Synthesizing Petri Nets from State-Based Models.* Proceedings of iccad'95 (1995) 164–171.

[30] Cortadella, J., Kishinevsky, M., Kondratyev, A., Lavagno, L., and Yakovlev, A., *Complete state encoding based on the theory of regions.* Proceedings of the 2nd International Workshop on Advanced Research in Asynchronous Circuits and Systems (1996) 36–47.

[31] Chand, D.R., and Kapur, S.S., *An algorithm for convex polytopes.* Journal of the Association for Computing Machinery, vol. 17, No 1 (1970) 78–86.

[32] Chou, T.J., and Collins, G.E., *Algorithms for the solution of systems of linear Diophantine equations*, siam Journal of Computing, 11 (1982) 687–708.

[33] Darondeau, Ph., *Deriving unbounded Petri nets from formal languages.* Proceedings of concur'98, Springer-Verlag Lecture Notes in Computer Science Vol. 1466 (1998) 533–548.

[34] DAVEY, B.A., and PRIESTLEY, H.A., *Introduction to Lattices and Order.* Cambridge University Press, (1990).

[35] DESROCHERS, A.A., and AL-JAAR, R., *Applications of Petri Nets in Manufacturing Systems. Modeling, Control, and Performance Analysis.* IEEE Press, New York (1995).

[36] DESEL, J., and ESPARZA, J., *Free Choice Petri Nets.* Volume 40 of Cambridge Tracts in Theoretical Computer Science. Cambridge University Press (1995).

[37] DESEL, J., REISIG, W., *The Synthesis Problem of Petri Nets.* Acta Informatica vol. 33 (1996) 297–315.

[38] DROSTE, M., and SHORTT, R.M., *Petri Nets and Automata with Concurrency Relations – an Adjunction.* in "Semantics of Programming Languages and Model Theory", M. Droste and Y. Gurevich eds(1993) 69–87.

[39] DROSTE, M., and SHORTT, R.M., *From Petri Nets to Automata with Concurrency.* Unpublished draft (1996).

[40] EDELSBRUNNER, H., *Algorithms in Combinatorial Geometry.* EATCS Monographs in Theoretical Computer Science vol. 10, Springer, Heidelberg (1987).

[41] EDELSBRUNNER, H., *Geometric Algorithms.* In P.M. Gruber and J.M. Wills (Eds.) Handbook of Convex Geometry vol. A, North-Holland (1993) 699–735.

[42] EHRENFEUCHT, A., and ROZENBERG, G., *Partial 2-structures ;* Part I : *Basic Notions and the Representation Problem*, Acta Informatica, vol. 27 (1990) 315–342.

[43] EHRENFEUCHT, A., and ROZENBERG, G., *Partial 2-structures ;* Part II :*State Spaces of Concurrent Systems*, Acta Informatica, vol. 27 (1990) 343–368.

[44] FERIGATO, C., *Note su Alcune Propietà Algebriche, Logiche e Topologiche della Concorrenza.* Dottorato di Ricerca in Informatica, Roma (1996).

[45] GAREY, M.R., and JOHNSON, D.S., *Computer and Intractability. A guide to the theory of NP-Completeness.* W.H. Freeman and Company (1979).

[46] VAN GLABBEEK, R., *Bisimulation Semantics for Higher Dimensional Automata.* Technical report, Stanford University (1991).

[47] VAN GLABBEEK, R., and VAANDRAGER, F., *Petri net models for algebraic theories of concurrency.* In Proceedings PARLE'87, vol. II: Parallel Languages, Eindhoven, volume 259 of Lecture Notes in Computer Science (1987) 224–242.

[48] GOUBAULT, E., *The Geometry of Concurrency.* Ph. D. Thesis, Ecole Normale Supérieure, Paris (1995).

[49] GROSS, J.L., and TUCKER, T.W., *Topological Graph Theory*, Wiley-Interscience, New York (1987).

[50] HERWIG, B., *Extending partial isomorphisms on finite structures*, Combinarotica 15 (1995) 365–371.

[51] HERWIG, B., and LASCAR, D., *Extending partial automorphism and the profinite topology on the free groups.* Draft (1997).

[52] HIRAISHI, K., *Some complexity results on transition systems and elementary net systems.* Theoretical Computer Science 135 (1994) 361–376.

[53] HOOGERS, P.W., KLEIJN, H.C.M., and THIAGARAJAN, P.S., *A Trace Semantics for Petri Nets.* Lecture Notes in Computer Science vol. 623 (1992) 595–604.

[54] HOOGERS, P.W., KLEIJN, H.C.M., and THIAGARAJAN, P.S., *An Event Structure Semantics for General Petri Nets*. Theoretical Computer Science, volume 153 (1996) 129–170.

[55] HRUSHOVSKI, E. *Extending partial isomorphisms of graphs*, Combinatorica 12 (1992) 204–218.

[56] ILIOPOULOS, C.S., *Worst-case complexity bounds on algorithms for computing the canonical structure of finite abelian groups and the Hermite and Smith normal forms of an integer matrix*. SIAM Journal on Computing, Vol. 18, No 4 (1989) 658–669.

[57] JENSEN, K., *Coloured Petri Nets. Basic Concepts, Analysis Methods and Pratical Use, Volume 1, Basic Concepts*. EATCS-Monographs, Springer Verlag (1992).

[58] JOHNSTONE, P.T., *Stone Spaces*. Cambridge University Press, (1982).

[59] KANNAN, R., and BACHEM, A., *Polynomial Algorithms for Computing the Smith and Hermite Normal Forms of an Integer Matrix*. SIAM Journal on Computing, Vol. 8, No 4 (1979) 499–507.

[60] KARP, R.M., and MILLER, R.E., *Parallel program schemata*. Journal of Computer and System Sciences vol. 3 (1969) 147–195.

[61] KISHINEVSKY, M., CORTADELLA, J., KONDRATYEV, A., LAVAGNO, L., TAUBIN, A., and YAKOVLEV, A., *Place Chart Nets and their Synthesis*. Technical Report 96-2-003 Department of Computer Hardware, University of Aizu (1996).

[62] KISHINEVSKY, M., CORTADELLA, J., KONDRATYEV, A., LAVAGNO, L., and YAKOVLEV, A., *Synthesis of General Petri Nets*. Technical Report 96-2-004, Department of Computer Hardware, University of Aizu (1996).

[63] LYNDON, R.C., and SCHUPP, P.E., *Combinatorial Group Theory*. Ergebnisse Vol. 89, Springer (1977).

[64] MAGNUS, W., KARRASS, A., and SOLITAR, D., *Combinatorial Group Theory*. Wiley, New York (1966).

[65] MARCUS, M., and MINC, H., *A Survey of Matrix Theory and Matrix Inequalities*. Dover Publications, New York (1992).

[66] MEMMI, G., and ROUCAIROL, G., *Linear algebra in net theory*. Proceedings of Net Theory and Applications, volume 84 of the Lecture Notes in Computer Science, Springer Verlag (1980) 213–223.

[67] MUKUND, M., *Petri Nets and Step Transition Systems*. International Journal of Foundation of Computer Science, vol 3, n^o 4 (1992) 443–478.

[68] MURATA, T., *Circuit Theoretic Analysis and Synthesis of Marked Graphs*. IEEE Transations on Circuits and Systems, Vol. CAS-24, No. 7 (1977) 400–405.

[69] MURATA, T., *Petri Nets: Properties, Analysis and Applications*. Proceeding of the IEEE, 77(4) (1989) 541–580.

[70] NIELSEN, M., ROZENBERG, G., AND THIAGARAJAN, P.S., *Elementary Transition Systems*. Theoretical Computer Science, vol. 96 (1992) 3–33.

[71] NIELSEN, M., ROZENBERG, G., AND THIAGARAJAN, P.S., *Transition Systems, Event Structures and Unfoldings*. DAIMI PB-353 Aarhus (1991).

[72] NIELSEN, M., ROZENBERG, G., AND THIAGARAJAN, P.S., *Elementary Transition Systems and Refinement*. Acta Informatica 29 (1992) 555–578.

[73] NIELSEN, M., and WINSKEL, G., *Models for Concurrency*, Handbook of Logic for Computer Science, vol. 3, Oxford University Press (1994) 100–200. Appeared also as Research Report DAIMI PB-463, Aarhus University, November 1993.

[74] PETERSON, J.L., *Petri Net Theory and the Modelling of Systems*, Englewood Cliffs, NJ, Prentice-Hall (1981).

[75] PETRI, C. A., *Kommunikation mit Automaten*, Schriften des IIM Nr. 2, Institut für Instrumentelle Mathematik, Bonn (1962). *English translation*: Technical Report RADC-TR-65-377, Griffiths Air Force Base, New York, vol. 1, suppl. 1 (1966).

[76] PETRI NETS TOOL DATABASE, CPN group at University of Aarhus, Denmark, http://www.daimi.aau.dk/PetriNets/tools/

[77] PETRI NET MAILING LIST, CPN group at University of Aarhus, Denmark, http://www.daimi.aau.dk/ petrinet/

[78] PETRICH, M., *Inverse Semigroups*, Wiley, New York (1984).

[79] PIETKIEWICZ-KOUTNY, M., *Transition systems of elementary net systems with inhibitor arcs.* Proceedings of ICATPN'97. Springer-Verlag Lecture Notes in Computer Science Vol. 1248 (1997) 310–327.

[80] PIETKIEWICZ-KOUTNY, M., *Synthesis of* ENI-*systems Using Minimal regions.* Proceedings of CONCUR'98, Springer-Verlag Lecture Notes in Computer Science Vol. 1466 (1998) 565–580.

[81] PORST, H.-E., and THOLEN, W., *Concrete Dualities.* In "Category Theory at Work", H. Herrlich, and H.-E. Porst (eds.), Heldermann Verlag Berlin (1991) 111-136.

[82] PRATT, V.R., *The Stone Gamut: A Coordinatization of Mathematics*, Proceedings of the 10^{th} Symposium on Logics in Computer Science, IEEE Computer Society (1995) 444–454.

[83] PRATT, V., *Modelling Concurrency with Geometry.* Proceedings of the 18^{th} ACM Symposium on Principles of Programming Languages, Orlando, ACM Press (1991) 311–322.

[84] RAYWARD-SMITH, V.J., *On computing the Smith normal form of an integer matrix.* ACM Transaction on Mathematical Software, Vol. 5. No 4 (1979) 451–456.

[85] REISIG, W., *Petri Nets.* EATCS Monographs on Theoretical Computer Science, Vol. 4, Springer Verlag (1985).

[86] REISIG, W., *Elements of Distributed Algorithms, Modeling and Analysis with Petri Nets.* Springer (1998).

[87] REUTENAUER, CH., *Aspects mathématiques des réseaux de Petri.* Masson, Paris (1989). English translation by I. Craig: *The Mathematics of Petri Nets.* Prentice-Hall, New York (1990).

[88] SANCHEZ-LEIGHTON, V., *Quelques relations entre les réseaux de Petri, la logique dynamique et les méthodes de la sémantique dénotationnelle.* Thèse de l'Université de Paris 6 (1983).

[89] SCHMITT, V., *Flip-Flop Nets*, Proceedings of Stacs 96, Lecture Notes in Computer Science vol. 1046 (1996) 517–528.

[90] SCHMITT, V., *Représentations finies de comportements concurrents.* Thèse de l'Université de Rennes I (1997).

[91] SCHRIJVER, A., *Theory of Linear and Integer Programming.* John Wiley (1986).

[92] SEIDEL, R., *Constructing Higher-Dimensional Convex Hulls at Logarithmic Cost per Face.* Proceedings Annual ACM Symposium on Theory of Computing 18 (1986) 404–413.

[93] SHIELDS, M.W., *Concurrent machines*, Computer Journal, vol. 28 (1985) 449–465.

[94] SHIELDS, M.W., *Deterministic asynchronous automata.* In *Formal Methods in Programming*, North Holland (1985).

[95] SILVA, M., and COLOM, J.M., *On the Computation of Structural Synchronic Invariants in P/T nets.* In "Advances in Petri Nets 1988", G. Rozenberg (Ed.), volume 340 of Springer Verlag Lecture Notes in Computer Science (1988) 386–417.

[96] SILVA, M., TERUEL, E., and COLOM, J.M., *Linear Algebraic Techniques for the Analysis of Net Systems.* Third Advance Course on Petri Nets, Dagstuhl Castle. To appear in the Lecture Notes in Computer Science (1997).

[97] STALLINGS, J.R., *Topology of Finite Graphs.* Inventiones Mathematicae. 71 (1983) 551–565.

[98] STILLWELL, J., *Classical Topology and Combinatorial Group Theory*, Springer Verlag, New York (1980).

[99] SWART, G., *Finding the Convex Hull Facet by Facet.* Journal of Algorithm 6 (1985) 17–48.

[100] VALK, R., *Self-Modifying Nets, a Natural Extension of Petri Nets.* Proceedings of Icalp'78, Lecture Notes in Computer Science vol. 62 (1978) 464–476.

[101] VALK, R., *Generalizations of Petri Nets.* Proceedings of MFCS'81, Lecture Notes in Computer Science vol. 118 (1981) 140–155.

[102] VICKERS, S., *Topology via Logic.* Cambridge Tracts in Theoretical Computer Science 5, (1989) Cambridge University Press.

[103] WILLE, R., *Restructuring Lattice Theory: an Approach based on Hierarchies of Concepts.* In I. Rival (ed.) *Ordered Sets*, NATO ASI Series 83,Reidel, Dordrecht (1982) 445–470.