Chapter 1
Abelian Invertible Automata

Klaus Sutner

1.1 Motivation

Historically, the idea of reversible computation had its roots in physics rather than logic: at the fundamental level, the laws of physics are reversible. Since computing devices can obviously be realized within the context of these laws, it is plausible that computation itself should be amenable to reversibility: there ought to be a way to make the requisite logical operations reversible [4, 17, 16]. Perhaps surprisingly, this idea turns out to be of some practical importance, since reversible computation can be carried out without any thermodynamical cost [5], at least as a matter of principle. Morita has given many ingenious examples of reversible computation in the context of discrete dynamical systems, and in particular cellular automata [21, 18, 19, 20]. As these examples show, and contrary to what was tacitly assumed till the 1960s, reversible computation is a rich and endlessly challenging area of computability theory. We now even have the beginnings of a more structural approach to reversible computation [1], following roughly Girard's Geometry of Interaction.

It can be argued that in the realm of algebra it is the concept of a group that best captures the notion of reversibility: any action by a group element g can be undone by the action associated with g^{-1}. It is thus natural to ask whether reversible computation might have any direct connections with group theory. A first step in this direction was taken by Serre, albeit in a different context: he suggested to study subgroups of the full automorphism group $\mathsf{Aut}(\mathbf{2}^\star)$ of the infinite binary tree $\mathbf{2}^\star$ [28]. The topological group $\mathsf{Aut}(\mathbf{2}^\star)$ is profinite, and thus Hausdorff, compact and totally disconnected. In some interesting cases, subgroups can be described by certain finite state machines that are naturally reversible. More precisely, there are Mealy automata $\mathcal{A}$ over a binary

Carnegie Mellon University, Pittsburgh, USA

alphabet that have associated inverse automata $\mathcal{A}'$ such that the composition of their respective transductions is the identity. The transductions in question are length-preserving and thus do not admit universal computation. Yet their associated groups are surprisingly complicated and there are many challenging questions associated with these automata. In fact, groups defined by invertible automata have become a standard source of examples and counterexamples in group theory [22, 29]. A case in point is a 5-state automaton due to Grigorchuk [10] that defines a subgroup of $\mathsf{Aut}(\mathbf{2}^\star)$ with intermediate growth, answering a question by Milnor. Grigorchuk goes so far as to describe the discovery of very small automata associated with complicated groups as one of the "wonderful phenomena in modern mathematics." If one considers semigroups rather than groups, there is even a 2-state non-invertible Mealy automaton that exhibits intermediate growth [2].

We are here interested in a slightly different perspective: the computational complexity of the discrete dynamical systems defined by invertible Mealy automata. Given an automorphism f defined by some automaton $\mathcal{A}$, one would like to understand the orbits $x\, f^\star$ of words $x \in \mathbf{2}^\star$ under f. In particular, one would like to analyze the computational complexity of the question whether a word appears in the orbit of another (Orbit Problem) and where in the orbit it appears (Timestamp Problem). These questions are quite difficult in general, so it makes sense to restrict one's attention to a scenario that is of little interest from the group theory perspective: all the groups in question are free Abelian. In this limited setting, one can give a description of the automorphism f in terms of the algebraic integers in an algebraic number field associated with the automaton. Nonetheless, it requires some amount of effort to characterize the associated automata. The existence of numeration systems for algebraic integers then produces a convenient normal form for the automorphisms and is helpful in classifying the relevant automata and tackling the orbit problems just mentioned. For example, somewhat surprisingly it turns out that, in some cases, the Orbit Problem can be solved by a finite state machine, despite the fact that the orbits have exponential length. Alas, little is known about the general situation.

We will here refrain from giving detailed proofs and confine ourselves to simple sketches. We refer the reader to the literature for details, in particular [22, 10, 29, 30, 32, 33, 24].

1.2 Transducers and Automorphisms of the Binary Tree

For our purposes, an invertible transducer is a type of Mealy automaton $\mathcal{A} = \langle\, Q, \mathbf{2}, \tau \,\rangle$ where $\tau : Q \times \mathbf{2} \to \mathbf{2} \times Q$. In the customary arrow notation, all transitions are of the form $p \xrightarrow{a/\pi_p(a)} q$; here π_p is a permutation of the alphabet $\mathbf{2}$ depending on the source state p; for background see [27, 6, 8]. Our automata have no initial and final states; Eilenberg referred to these devices as output modules [7]. In order to obtain a transduction $\mathcal{A}(p)$ from $\mathbf{2}^\star$ to $\mathbf{2}^\star$ we select an arbitrary initial state p in $\mathcal{A}$. To lighten notation, we write $\underline{p}$ for this map whenever the automaton is clear from context. Note that all our automorphisms naturally extend to maps $\mathbf{2}^\omega \to \mathbf{2}^\omega$. We refer to states such that $\pi_p = I$ as copy states, and as toggle states otherwise. We will write application of our automorphisms as a right action on finite or infinite words, $x\, f$.

Any interesting invertible automaton must have at least one copy state and one toggle state. Surprisingly, the 2-state machine in figure 1.1, with one toggle and copy state each, already generates the lamplighter group, in perfect keeping with the Grigorchuk's observation from above.

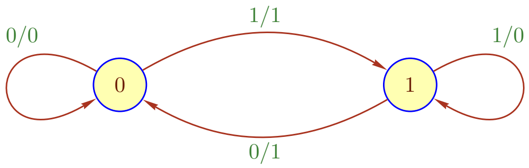

Fig. 1.1 The smallest interesting invertible automaton.

The transduction semigroup generated by all the $\mathcal{A}(p)$, $p \in Q$, under composition will be written $\mathcal{S}(\mathcal{A})$; and $\mathcal{G}(\mathcal{A})$ the corresponding group. While it is convenient to admit infinite automata, the situation where the Mealy automaton has finite state set is by far the most interesting. As it turns out, many interesting groups admit such a representation: free groups, free Abelian groups, certain nilpotent groups, the lamplighter group, and in particular Grigorchuk's group. Computationally, it is straightforward to construct automata that represent the elements of $\mathcal{S}(\mathcal{A})$ using the standard product construction for the composition of rational transductions described in [8]. Of course, the size of these machines grows exponentially, so the construction is only feasible in rather limited circumstances. To handle the group $\mathcal{G}(\mathcal{A})$, one usually also needs the inverse automaton $\mathcal{A}'$ that is obtained by interchanging the input/output bits for all toggle states. Note that we may have $\mathcal{S}(\mathcal{A}) = \mathcal{G}(\mathcal{A})$, in which case there is no need for the inverse automaton; we call $\mathcal{A}$ group-like in this situation. It is easy to see that $\mathcal{S}(\mathcal{A})$ is Abelian if, and only if, $\mathcal{G}(\mathcal{A})$ is Abelian.

We will transfer standard notions from semigroup and group theory to the corresponding automata. For example, we may refer to an automaton as being torsion-free Abelian.

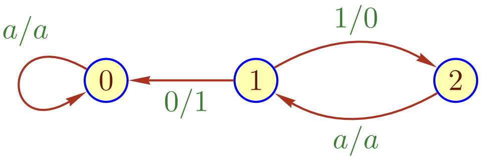

Fig. 1.2 The successor automaton Suc_2 of rank 2, generating the free monoid $\mathbb{N}^2$.

Here is are some simple examples. The automaton Suc_2 in figure 1.2, the successor automaton of rank 2, generates the monoid $\mathbb{N}^2$. If the loop has length n rather than 2 as in figure 1.2 we obtain the general successor automaton of rank n, see below for an algebraic definition. In the case where $n = 1$ these machines are also referred to as adding machines or odometers; regrettably, for $n \geq 2$ the notion of sausage automaton appears in the literature. It is not hard to see that all orbits of our invertible binary automata have length a power of 2. For Suc_2, the orbit of a word x of length $2k$ under transduction $\underline{1}$ has length 2^k. To see why, let $u, v \in \mathbf{2}^k$ and write shf for the perfect shuffle operation. Then

$$\mathsf{shf}(u, v)\,\underline{1} = \mathsf{shf}(u\,f, v)$$

where f is the truncated successor function in reverse binary (another automorphism of $\mathbf{2}^\star$, defined by the analogous successor automaton where state 1 has a self-loop under 1/0).

By contrast, the machine $\mathcal{A}$ in figure 1.3 generates $\mathbb{Z}^2$ as a semigroup. To see why, note that $\underline{1}^{\,-1} = \underline{2}$, and similarly $\underline{3}^{\,-1} = \underline{4}$. This is an example of an automaton that admits a skew-symmetry φ: a transition $p \xrightarrow{a/b} q$ is mapped to $\varphi(p) \xrightarrow{\overline{a}/\overline{b}} \varphi(q)$. Thus, state $\varphi(p)$ defines the inverse function of state p and the automaton is obviously group-like. As in the previous example, the orbit of a word of length $2k$ under $\underline{1}$ has length 2^k and has a similar description in terms of shuffle. Alas, this time there is no simple description for the associated map f. Note that $\underline{1}^{\,k}$ similarly produces orbits of length 2^k for odd k, whereas even powers of $\underline{1}$ produce shorter orbits.

The example also shows that full edge labels in the diagrams lead to visual clutter. It is preferable to relabel transitions as follows; this convention will also be useful in section 1.4. Henceforth, a toggle transition $p \xrightarrow{0/1} q$ will be written as $p \xrightarrow{1} q$, a toggle transition $p \xrightarrow{1/0} q$ as $p \xrightarrow{\overline{1}} q$ and a copy transition $p \xrightarrow{a/a} q$ as $p \xrightarrow{0} q$ or even simply as $p \longrightarrow q$. The labels $\mathbf{2}_s = \left\{\overline{1}, 0, 1\right\}$ will be referred to as trits to emphasize similarity to balanced ternary numeration systems. Numerically we will interpret $\overline{1}$ as -1.

Here is our last example of an automaton, using this convention. The 3-state machine is group-like and indeed generates $\mathbb{Z}^2$, but this time there is

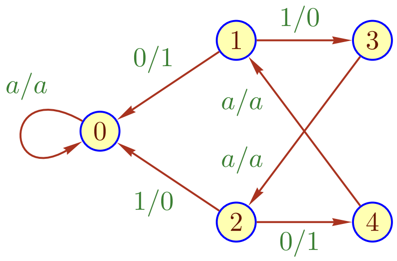

Fig. 1.3 An automaton that uses a successor-like function to generate the free Abelian group of rank 2.

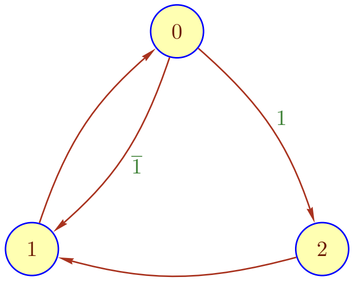

Fig. 1.4 Another automaton generating the free Abelian group of rank 2, albeit it in a less obvious manner.

no obvious reason for this. As it turns out, the identity $\underline{0}^{\,2}\underline{1}^{\,2}\underline{2} = I$ holds, from which observation our claim can easily be derived (to avoid confusion, we always write I for the identity automorphism of $\mathbf{2}^\star$). Again, the orbit of words of length $2k$ under $\underline{0}$ has length 2^k, but this time the situation is a bit more complicated: one can show that for any fixed orbit under $\underline{0}$ and any $u \in \mathbf{2}^k$ there is precisely one $v \in \mathbf{2}^k$ such that $\mathsf{shf}(u, v)$ lies in the orbit. Hence the orbits of even length words under $\underline{0}$ have the form $\{\, \mathsf{shf}(u, v\, f) \mid u \in \mathbf{2}^k \,\}$ for some function finite transduction f.

The description of automorphisms of $\mathbf{2}^\star$ in terms of Mealy automata is convenient since standard algorithms from automata theory can be exploited in the study of these automorphisms. For example, we may safely assume that the Mealy automata are minimal in the sense that no two states have the same behavior, if we consider them as acceptors over $\mathbf{2} \times \mathbf{2}$ in the obvious manner. All the sample automata we have seen so far are indeed minimal. Thus we may safely assume that all the basic maps $\underline{k}$, $k \in Q$, are distinct.

Of course, it is also important to have a more algebraic description available. To begin with, note that any automorphism f of $\mathbf{2}^\star$ can be written in the recursive form $f = (f_0, f_1)s$ where $s \in \mathfrak{S}_2$, the symmetric group on two letters: s describes the action of f on $\mathbf{2}$, construed as the first level of $\mathbf{2}^\star$, and f_0 and f_1 are the automorphisms induced by f on the two subtrees of the root (which are naturally isomorphic to the whole tree). Thus there are residuation maps $\partial_a : \mathsf{Aut}(\mathbf{2}^\star) \to \mathsf{Aut}(\mathbf{2}^\star)$, $a \in \mathbf{2}$, and a parity map $\mathrm{par} : \mathsf{Aut}(\mathbf{2}^\star) \to \mathfrak{S}_2$ that produce the corresponding decomposition. Here we write σ for the transposition in $\mathfrak{S}_2$ and suppress the identity in this context. We refer to an automorphism of the form $f = (f_0, f_1)\sigma$ as odd, all others $f = (f_0, f_1)$ as even. In other words, f is even if $a\,f = a$ for $a \in \mathbf{2}$, and odd otherwise. Clearly, $\underline{p}$ is odd if, and only if, p is a toggle state. We can describe the full automorphism group as a wreath product:

$$\mathsf{Aut}(\mathbf{2}^\star) \simeq (\mathsf{Aut}(\mathbf{2}^\star) \times \mathsf{Aut}(\mathbf{2}^\star)) \rtimes \mathfrak{S}_2.$$

The group operation in the wreath product has the form

$$(f_0, f_1)s\,(g_0, g_1)t = (f_0 g_{s(0)}, f_1 g_{s(1)})\,st.$$

In the context of sequential functions, residuals were first introduced by Raney [26] and correspond exactly to the recursive components in the wreath decomposition. Note that a subgroup G of $\mathsf{Aut}(\mathbf{2}^\star)$ may not be closed under residuation; if it is, we call G self-similar or state-closed. In this case, the wreath characterization of the full automorphism group carries over: $G \simeq (G \times G) \rtimes \mathfrak{S}_2$.

For legibility, we will occasionally write k^- rather than $k-1$, and k^+ rather than $k+1$. As an example, using wreath notation, the successor automaton Suc_n of rank n, with a loop of length n rather than just 2 as in figure 1.2, has the form

$$\underline{0} = (\underline{0}, \underline{0}) \qquad \underline{1} = (\underline{0}, \underline{n})\,\sigma \qquad \underline{k} = (\underline{k^-}, \underline{k^-}), \quad 2 \le k \le n.$$

Using the shuffle characterization from above, it is not hard to show that Suc_n generates the free Abelian monoid $\mathbb{N}^n$, but not a group. The automaton in figure 1.3 can be generalized like so. A cycle-cum-chord transducer is given by

$$\underline{0} = (\underline{n^-}, \underline{m^-})\sigma \qquad \underline{k} = (\underline{k^-}, \underline{k^-}), \quad 1 \le k < n.$$

where $1 \le m \le n$. We will write CC^n_m for this transducer. The diagram of CC^6_4 is shown in figure 1.5. One can show that CC^n_m generates the free Abelian group of rank $n - \gcd(n, m)$. As we will see, these are in a sense the most basic invertible automata generating free Abelian groups.

Self-similar subgroups of $\mathsf{Aut}(\mathbf{2}^\star)$ can always be translated into Mealy automata, though not necessarily finite ones. For suppose G is self-similar. We can construct the complete group automaton for G, in symbols $\mathfrak{C}_G$, as follows: the automaton has G as state set and the transitions $f \xrightarrow{a/a\,f} \partial_a f$.

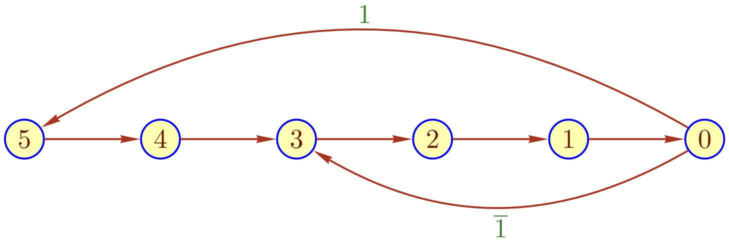

Fig. 1.5 A cycle-cum-chord automaton that generates the free Abelian group of rank 4.

In general, this invertible transducer will be infinite, but certainly $\mathcal{S}(\mathfrak{C}_G)$ is a group and isomorphic to G. Note that the complete automaton always admits the skew-symmetry mentioned above. As already mentioned, more interesting is a representation of G in terms of a finite automaton. To this end, call G is finite-state if for all $f \in G$ the number of residuals $\partial_x f$ is finite. If G is self-similar, finite-state and finitely generated, we can construct the group automaton $\mathfrak{A}_G$, a subautomaton of $\mathfrak{C}_G$, just like the complete group automaton, but with state set restricted to the collection of all residuals of the generators of G. The group generated by $\mathfrak{A}_G$ is isomorphic to G, but the semigroup may be different as is the case for successor automata. Note that $\mathfrak{A}_G$ is minimal by construction. To recover parts of $\mathfrak{C}(\mathcal{A})$ computationally from $\mathfrak{A}_G$ we can use the standard product machine construction combined with minimization to obtain a machine for each automorphism $f \in G$. Unless $\mathfrak{A}_G$ is group-like, some of the components in these products will be copies of the inverse automaton $\mathfrak{A}'_G$. The complete automaton is then the limit of these automata. Note that the product machine construction combined with minimization is directly related to questions of growth, so one should in general expect no simple descriptions of the resulting automata [3].

1.3 Abelian Automata

Given a nontrivial, state-closed group G acting on $\mathbf{2}^\star$, it is clear that the collection of even elements forms a subgroup H of index 2. Moreover, restricted to H, the residuation maps are group homomorphisms. Correspondingly, one can define an action of a group G on $\mathbf{2}^\star$, given a subgroup $H \leq G$ of index 2 and a homomorphism $\Phi : H \to G$. Fix coset representatives $h_0 = 1$ and $h_1 \in G - H$ and define the action via

$$ax\, f = b\,(x\, \Phi(h_b^{-1} f h_a))$$

where b is determined by the condition that $h_b^{-1} f h_a \in H$. Unfortunately, this action may fail to be faithful and the conditions under which it is are slightly complicated, see [22, 23]. In the Abelian case, no problems arise and one can rewrite this characterization using additive notation in the form

$$ax\,f = \begin{cases} a\,(x\,\Phi(f)) & \text{if } f \in H, \\ \overline{a}\,(x\,\Phi(f + (-1)^a g)) & \text{if } f \notin H \end{cases}$$

where g is a suitably chosen coset representative, $g \in G - H$. As an example, consider $G = \mathbb{Z}^2$ with generators $\mathbf{e}_1$ and $\mathbf{e}_2$ and $H = \langle\, 2\mathbf{e}_1, \mathbf{e}_2 \,\rangle$. We can set $\Phi(2a, b) = (b, a)$ and let $g = \mathbf{e}_1$. Then, for example, $0^\omega(4, 3) = 01001^\omega$ and $1^\omega(4, 3) = 101001^\omega$.

It is fairly easy to check whether a given invertible Mealy automaton generates an Abelian group. Suppose $G \leq \mathsf{Aut}(\mathbf{2}^\star)$ is self-similar. For any automorphism $f \in G$ define its gap to be $\gamma_f = (\partial_0 f)(\partial_1 f)^{-1} \in G$, so that $\partial_0 f = \gamma_f\, \partial_1 f$. An easy induction using wreath representation shows the following.

Lemma 1. *A self-similar group $G \leq \mathsf{Aut}(\mathbf{2}^\star)$ is Abelian if, and only if, all even elements of G have gap value I, and all odd elements have the same gap value.*

The conditions of the lemma naturally carry over to an automaton generating the group G. So suppose $\mathcal{A}$ is an invertible automaton that satisfies the gap conditions so that $\mathcal{S}(\mathcal{A})$ is Abelian. To avoid tedious special cases, let us note that the transduction monoid and group are Boolean precisely when every toggle state has gap I. By minimality, this means that every such state has out-degree 1 (we consider the transition diagram to be a simple graph rather than a multi-graph). In the other case we obtain a free Abelian monoid and group. From now on, we will always assume that the we are in the second case. We refer to $\gamma_(\underline{p})$ for any toggle state p as the gap value of $\mathcal{A}$. It follows easily from minimality that any state p has at most one predecessor copy state. For toggle predecessors note that for any $a \in \mathbf{2}$ there can be at most one q such that $q \xrightarrow{a/\overline{a}} p$. However, it may well happen that there are distinct predecessors q_0 and q_1 such that $q_a \xrightarrow{a/\overline{a}} p$. Thus, every state has indegree at most 3 in a minimal invertible automaton, and it is not hard to show that this bound is tight.

So suppose $\mathcal{A}$ is a minimal invertible automaton with n states and n_0 toggle states. We may safely assume $n_0 < n$. It is clear that one can test whether $\mathcal{A}$ is Abelian in polynomial time by checking directly that all the maps $\underline{p}$ commute. This requires a product machine construction $\mathcal{A}_p \otimes \mathcal{A}_q$ as described in [8] and a test that $\mathcal{A}_p \otimes \mathcal{A}_q$ and $\mathcal{A}_q \otimes \mathcal{A}_p$ are behaviorally equivalent. The latter property can be handled in time $\widetilde{O}(n^2)$ using the standard algorithm in [11]. Using the gap characterization, we can instead check that all the copy states have out-degree 1 and check that all product automata $\mathcal{A}_p \otimes \mathcal{A}'_p$ where p is a toggle state have the same behavior.

Lemma 2. *Given a minimal invertible automaton on n states, one can test commutativity in $\widetilde{O}(n_0 n^2)$ steps.*

Note that the test is trivial for an invertible automaton that contains just a single toggle state p. In this case, the automaton will be Abelian if, and only if, it consists of a copy chain, a directed path of copy states, ending in a toggle state of the form

$$q_k \xrightarrow{0} q_{k-} \xrightarrow{0} \ldots \xrightarrow{0} q_1 \xrightarrow{0} q_0$$

plus two back transitions starting at the toggle state. We refer to this part of the complete automaton as the copy chain at q_0 of length k. As an example, consider the cycle-cum-chord automaton in figure 1.5. Note that copy chains of arbitrary length always exist; in fact, we can construct an infinite copy chain at any toggle state, see section 1.5 for an application of this idea.

For the two transitions emanating from the toggle state q_0, there are two possibilities. First, they may both end at two copy states in the chain. We may safely assume that one of the transitions leads back to q_k, so we are dealing with a cycle-cum-chord transducer. The other possibility is that one of these transitions leads to the identity state (recall that we assume minimality). In this case we obtain a successor automaton Suc_n, see [22].

Consider a group $G \leq \mathsf{Aut}(\mathbf{2}^\star)$. Following Nekrashevych and Sidki [23], we will refer to G as an m-lattice if G is state-closed and free Abelian of finite rank m. Suppose $\mathcal{A}$ generates an m-lattice. As we have seen, the complete automaton associated with $\mathcal{A}$ is a computable structure. In particular, we can effectively construct a finite subautomaton for any automorphism $f \in \mathcal{G}(\mathcal{A})$. The reference shows that a computationally preferable representation of the complete automaton can be obtained by using $\mathbb{Z}^m$ directly as state set. Call $u \in \mathbb{Z}^m$ even if its first component u_1 is even, and odd otherwise. Then the transition function τ of the complete automaton can be described in terms of a residuation matrix $\mathsf{A} \in 1/2\,\mathbb{Z}^{m \times m}$ and an odd residuation vector $e \in \mathbb{Z}^m$ by

$$\tau(u, d) = \mathsf{A}(u + de) \tag{1.1}$$

where $d = 0$ whenever u is even, and $d = \pm 1$ otherwise (recall our labeling convention from above). Writing c_i for the ith column of A, we have $e = \mathsf{A}^{-1}(c_1 + v)$ where v is integral. The matrices A in question are non-singular and 1/2-integral: $\mathsf{A}^{-1}(\mathbb{Z}^m) \cap \mathbb{Z}^m$ is a sublattice of $\mathbb{Z}^m$ of index 2. As a matter of fact, using similarity with respect to $\mathsf{GL}_m(\mathbb{Z})$, we can safely assume that the matrix A has the form

$$\mathsf{A} = \begin{pmatrix} \frac{a_{11}}{2} & a_{12} & \dots & a_{1m} \\ \frac{a_{21}}{2} & a_{22} & \dots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{a_{m1}}{2} & a_{m2} & \dots & a_{mm} \end{pmatrix} \tag{1.2}$$

where all the coefficients a_{ij} are integral. One can verify that the characteristic polynomials of these matrices have the form

$$\chi(z) = z^m + 1/2\big(g_{m-1}z^{m-1} + \ldots + g_1 z + g_0\big) \tag{1.3}$$

where all the coefficients g_i are integral; in particular $g_0 = \pm 1$. In the case of interest to us when the action induced by A is faithful, it is shown in the reference that $\chi(z)$ is irreducible, a property we will tacitly assume from now on. Computational evidence suggests that most matrices A have but one $\mathsf{GL}_m(\mathbb{Z})$ class [24, 23]. In this case we can further assume that the matrix A has the form of a companion matrix with fractional components again only appearing in the first column, and all other columns being unit vectors. One property of these characteristic polynomials $\chi(z)$ that is crucial for our purposes is the fact that all their roots have modulus strictly less than 1. Thus any residuation matrix A has spectral radius strictly less than 1, and A is a contraction. As elements of the corresponding algebraic number field, all roots have denominator 2.

The complete automaton $\mathfrak{C}(\mathsf{A}, e)$ now takes the following simple form and is obviously computable: the state set is $\mathbb{Z}^m$ and the transition function is given by equation (1.3). Unlike the product automata mentioned earlier, this infinite Mealy automaton is always reduced in the sense that any two distinct states have distinct behavior. In the following we will always interpret the complete automaton in this manner.

1.3.1 Canonical and Principal Automata

As in the last section, consider the complete automaton $\mathfrak{C}(\mathsf{A}, e)$ for some m-lattice G. We are interested in finite subautomata of $\mathfrak{C}(\mathsf{A}, e)$ that generate the same lattice. Following [22], define the nucleus $\mathcal{N}$ of the action as the following subset of G:

$$\mathcal{N} = \bigcup_{g \in G} \bigcap_{n \in \mathbb{N}} \{\, \partial_x g \mid |x| \geq n \,\}$$

Thus, $\mathcal{N}$ consists of all states of the complete automaton that are reachable from a cycle. Note that $\mathcal{N}$ naturally defines a subautomaton of $\mathfrak{C}(\mathsf{A}, e)$ and it is shown in [22] that this automaton generates the lattice. The action is called

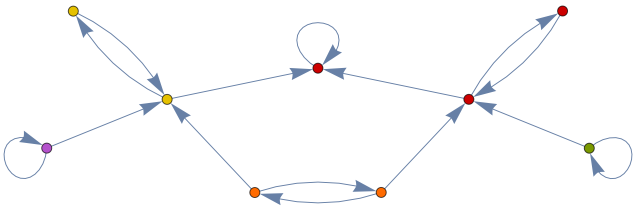

Fig. 1.6 The nucleus automaton associated with the sausage automaton of rank 2.

contracting if $\mathcal{N}$ is finite and one can show that all m-lattices are contracting in this sense. As a consequence, it is decidable whether two Abelian automata generate the same lattice. The algorithm given in the reference relies on the claim that the nucleus can be effectively generated; see below for a plausible method. An image of the nucleus of the successor automaton Suc_2 of rank 2 is shown in figure 1.6. Note the extraneous strongly connected components that could be removed without changing the generated group (the subautomaton colored red already generates the group).

Select some anchor point $u \in \mathbb{Z}^m$, $u \neq 0$, and form the closure under the transition function defined by equation (1.3). The elements of the closure will have the form

$$\mathsf{A}^k \cdot u + \left(a_k \mathsf{A}^k + a_{k-1} \mathsf{A}^{k-1} + \ldots + a_1 \mathsf{A}\right) \cdot e \tag{1.4}$$

where all the coefficients a_i are trits, $a_i \in \mathbf{2}_s$. Since A is a contraction, the closure is always finite. We will refer to the resulting automaton as a canonical automaton (for A) and write $\mathfrak{A}(\mathsf{A}, e, u)$. We may safely assume that u is odd, otherwise $\mathfrak{A}(\mathsf{A}, e, u)$ has the form $\mathfrak{A}(\mathsf{A}, e, u_0)$ plus a copy chain ending at u_0, the first toggle state obtained by repeated residuation from u. The description of points in the closure becomes somewhat simpler if the anchor point u is equal to e; correspondingly we write $\mathfrak{A}(\mathsf{A}, e)$ and even $\mathfrak{A}(\mathsf{A})$ if in addition $e = \mathbf{e}_1$. The latter automaton will be called the principal automaton (for A) and is entirely determined by the residuation matrix A. In either case, we are dealing with subautomata of the finite nucleus. Note that the principal automaton always contains the sink 0 since there is a transition $\mathbf{e}_1 \xrightarrow{-1} 0$. Computational evidence suggests that the principal automaton almost always contains $-\mathbf{e}_1$ and is skew-symmetric. The only exception to this rule appears to be the successor automata based on the characteristic polynomial $\chi(z) = z^m - 1/2$ as in figure 1.2: here the principal automaton has three strongly connected components, the sink plus two parts that are skew-symmetric to each other. In this case, either one of these components alone produces just a monoid, not a group. To avoid special cases, in this situation we will here refer to the automaton comprised of all three strongly

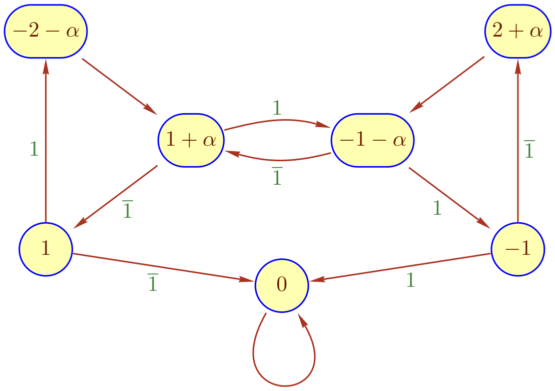

Fig. 1.7 The principal automaton associated with $\chi(z) = z^2 + z + 1/2$. States are labeled by algebraic integers. Note the skew-symmetry.

connected components as the principal automaton. Note that the nucleus automaton is strictly larger than the principal one for all $m \geq 2$.

Now consider the condensation graph of $\mathfrak{C}(\mathsf{A}, e)$, i.e., the graph whose nodes are the strongly connected components of $\mathfrak{C}(\mathsf{A}, e)$ and whose edges are inherited. Then remove all nodes that fail to be both reachable and co-reachable from non-trivial strongly connected components; call this the strict condensation graph. It is easy to see from equation (1.4) that there are only finitely many non-trivial strongly connected components, and they are all finite. A component induces a subautomaton if it is terminal in the strict condensation graph: it has no out-edges. Again, we will think of the terminal component 0 as being part of the principal automaton.

An algorithm to compute the nucleus is implicit in Okano [24]. Let V by the Vandermonde matrix given by the m roots of the characteristic polynomial $\chi(z)$, define the vector norm $\|x\| = \|V \cdot x\|_\infty$ in terms of the Chebyshev norm and let $\lambda < 1$ be the spectral radius of A. Then $\|c_1\| = \lambda^m$ and, for the induced matrix norm, we have $\|\mathsf{A}\| = \lambda$. For any point u on a cycle one can then show that $\|u\| \leq \lambda^m/(1-\lambda)$. Thus the search for strongly connected components can be limited to a finite region of the complete automaton. Note, though, that the region becomes quite large when λ is close to 1.

As an example, consider the residuation matrix

$$\mathsf{A} = \begin{pmatrix} -1 & 1 \\ -1/2 & 0 \end{pmatrix} \qquad \chi(z) = z^2 + z + 1/2$$

This is the matrix associated with the cycle-cum-chord transducer CC_2^3 in figure 1.4 and has spectral radius $\lambda = 1/\sqrt{2}$. The roots of $\chi(z)$ have absolute

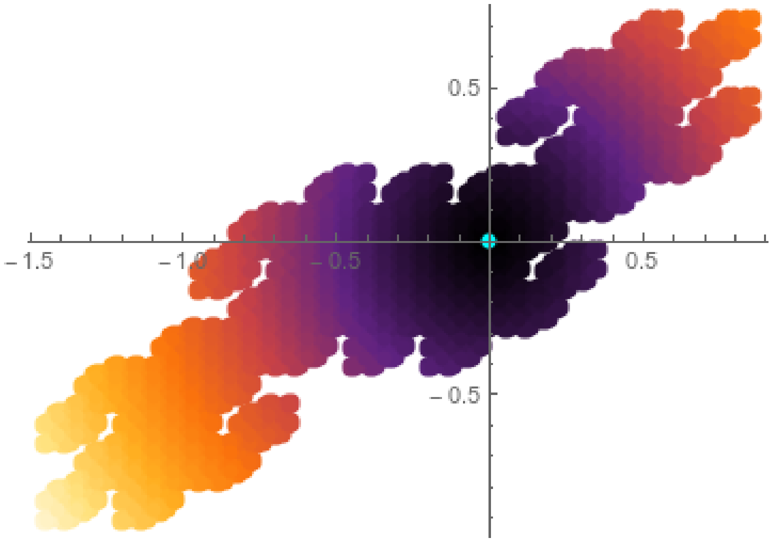

Fig. 1.8 The integral self-affine tile associated with $\chi(z) = z^2+z+1/2$, using the standard digit $(1,0)$. The tile has Lebesgue measure 1 and lattice tiles $\mathbb{R}^2$ with tiling set $\mathbb{Z}^2$. It is known as the twin-dragon.

norm 2. The corresponding principal automaton, a 7 state machine, is shown in figure 1.7; the state labels will be explained in section 1.4. In this case, the strongly connected component of the anchor point $\mathbf{e}_1$ admits a skew-symmetry. The principal automaton here coincides with the nucleus. On the other hand, if we select the residuation vector to be $e = (3,2)$, the canonical automaton $\mathfrak{A}(\mathsf{A}, e)$ has but 3 states and is isomorphic to CC_2^3. Clearly $\mathfrak{A}(\mathsf{A}, e)$ fails to admit any skew-symmetry, yet it still generates the free Abelian group of rank 2 [33]. In this case, the nucleus has 5 subautomata: the principal automaton, plus two pairs of skew-symmetric ones. These automata are generated by the powers of the transduction $\mathsf{CC}_2^3(0)$.

Consider some subautomaton $\mathcal{A}$ of $\mathfrak{C}(\mathsf{A}, e)$ and a transition $p \xrightarrow{d} q$ whose target q lies in $\mathcal{A}$, but whose source p does not. Let $\mathcal{A}_+$ denote the smallest subautomaton of $\mathfrak{C}(\mathsf{A}, e)$ that contains $\mathcal{A}$ and p.

Lemma 3. *$\mathcal{A}_+$ generates the same group as $\mathcal{A}$.*

To see why, first assume that p is a copy state and consider the copy chain at $q = q_0$ of length m^-. As a consequence of equation (1.3) and using additive notation we have

$$2\underline{q_0} + g_{m-1}\underline{q_1} + \ldots \pm \underline{q_{m^-}} = I. \tag{1.5}$$

By repeated residuation it follows that $p = q_1 \in \mathbb{Z}[Q]$ where Q is the state set of $\mathcal{A}$. If p is a toggle state, we may safely assume that the new transition has the form $p \xrightarrow{\overline{1}} q$. Let $p \xrightarrow{1} q'$, so that $q' = q + \gamma \in \mathbb{Z}[Q]$ where γ is the gap value from above. Hence we can apply the same residuation argument as in the first case to show that $p \in \mathbb{Z}[Q]$.

Reading the lemma in the opposite direction, we see that there are two types of interesting subautomata of the nucleus of $\mathfrak{C}(\mathsf{A}, e)$

- the principal automaton $\mathfrak{A}(\mathsf{A})$, and
- terminal strongly connected automata.

To see why, consider a terminal strongly connected component S in a subautomaton $\mathcal{A}$ of the complete automaton. If S consists only of 0 we are dealing with the principal automaton; otherwise S itself defines a subautomaton that generates the same group as $\mathcal{A}$. The principal automata all seem to have a non-trivial skew automorphism, all the others do not (but recall our convention regarding successor automata as in figure 1.2).

1.3.2 Self-Affine Tiles

Let us digress briefly to comment on the connection between Abelian invertible automata and questions related to tilings and iterated function systems. Since A is a contraction, the representation of the elements of a subautomaton of $\mathfrak{C}(\mathsf{A}, e)$ in equation (1.4) naturally gives rise to a so-called tile, a compact subset of $\mathbb{R}^m$ of positive Lebesgue measure. More precisely, fix a set $\mathcal{D} \subseteq \mathbb{R}^m$ of generalized digits. We are only interested in the case $|\mathcal{D}| = 2$; moreover, we may assume without loss of generality that one of the digits is 0 so that $\mathcal{D} = \{0, d\}$. We can think of the tile as being determined by an iterated function system given by A and $\mathcal{D}$: we are interested in the compact set $T \subseteq \mathbb{R}^m$ such that

$$T = \mathsf{A}(T + \mathcal{D}) \tag{1.6}$$

It is easy to see that T has the explicit representation

$$T = \left\{ \sum_{i=1}^{\infty} \mathsf{A}^{-i} d_i \;\middle|\; d_i \in \mathcal{D} \right\} \subseteq \mathbb{R}^m \tag{1.7}$$

The tile T is called self-affine if it has positive Lebesgue measure, a property that is somewhat rare. To develop a test for positive measure, consider all real vectors that admit a description in terms of a k-digit expansion of the form

$$\mathcal{D}_k = \left\{ \sum_{i=0}^{k-1} \mathsf{A}^{-i} d_i \;\middle|\; d_i \in \mathcal{D} \right\} \tag{1.8}$$

It was shown by Lagarias and Wang [14, 15] that T has positive measure if, and only if, the cardinality of $\mathcal{D}_k$ is 2^k for all k. It follows from the results in section 1.5 that this condition is satisfied for our residuation matrices and digit sets described below.

Now define the containment lattice of A^{-1} and $\mathcal{D}$ to be the $\mathbb{Z}$-module generated by the pre-images of $\mathcal{D}$ under A^{-1}:

$$\mathbb{Z}[\mathsf{A}^{-1}, \mathcal{D}] = \mathbb{Z}[\mathcal{D}, \mathsf{A}^{-1}\,\mathcal{D}, \ldots, \mathsf{A}^{-n+1}\mathcal{D}] \tag{1.9}$$

Clearly, the containment lattice contains the symmetric digit set $\mathit{\Delta}\mathcal{D} = \{-d, 0, +d\}$ and is closed under A^{-1}. A digit set is said to be primitive if $\mathbb{Z}[\mathsf{A}^{-1}, \mathcal{D}] = \mathbb{Z}^m$. Now consider the digit $d = \mathsf{A}^{-1}(c_1 + v)$. The containment lattice is none other than the $\mathbb{Z}$-linear closure of the set of points co-reachable from the origin in $\mathfrak{C}(\mathsf{A}, d)$. Indeed, in conjugation with the conjecture in section 1.6, it consists precisely of these points; to wit, the collection of all tame automorphisms, see section 1.6 for definitions. At any rate, in particular for $d = \mathbf{e}_1$ we obtain a primitive digit set. A primitive digit set is standard if the digits form a complete residue system of $\mathbb{Z}^n/\mathsf{A}^{-1}\mathbb{Z}^n$. In other words, for a non-zero digit d we must have $\mathsf{A}\,d$ non-integral. This is, of course, precisely the choice of the residuation vector. Hence we obtain self-affine tiles, and for $d = \mathbf{e}_1$ the tiling set can be chosen to be the full lattice $\mathbb{Z}^m$. Figure 1.8 shows an example of such a tile. Note that the origin is the only integral point in the tile.

1.4 Path Polynomials

We will now develop yet another representation of the complete automaton that is helpful in studying path existence problems. As a starting point, consider the question of how the nuclei of $\mathfrak{C}(\mathsf{A}, e)$ and their subautomata compare for different values of the residuation vector e. Fix some residuation matrix A and vector $e = \mathsf{A}^{-1}(c_1 + v)$ where c_1 is the first column of A and v is integral. Our first result shows that as far as the gap value is concerned, only the residuation matrix matters.

Theorem 1. *The principal automaton is isomorphic to a subautomaton of $\mathfrak{C}(\mathsf{A}, e)$ for all e. As a consequence, the gap value of $\mathfrak{C}(\mathsf{A}, e)$ depends only on the residuation matrix A.*

There are two ways to establish this result. The first is essentially taken from [24] and directly constructs a linear map that defines the embedding. To this end, define an $m \times m$ integral matrix with column vectors

$$E = \left(e, \mathsf{A}^{-1}e, \ldots, \mathsf{A}^{-m^{-}}e\right) \tag{1.10}$$

One can check that E is non-singular and it commutes with A. As a consequence, E induces a monomorphism from $\mathfrak{C}(\mathsf{A})$ to $\mathfrak{C}(\mathsf{A}, e)$, so that $\mathfrak{C}(\mathsf{A})$ is always (isomorphic to) a subautomaton of $\mathfrak{C}(\mathsf{A}, e)$. In particular, E shows that $\mathfrak{A}(\mathsf{A}, e, e)$ is an isomorphic copy of the principal automaton $\mathfrak{A}(\mathsf{A})$.

A rather different approach to theorem 1 focuses on a description of paths in the complete automaton $\mathfrak{C}(\mathsf{A}, e)$, and in particular in the principal automaton. Consider a path

$$\pi : e \xrightarrow{d_0} q_1 \xrightarrow{d_1} \dots \xrightarrow{d_{k-2}} q_{k-} \xrightarrow{d_{k-}} q_k \tag{1.11}$$

Only paths such that $q_i \neq 0$ for $i < k$ are of interest. We will interpret the label $\mathsf{lab}(\pi) = d = d_0 d_1 \dots d_{k-}$ as a word over the three-letter alphabet $\mathbf{2}_s = \{-1, 0, 1\}$. For reasons of legibility, we will often write $\overline{1}$ instead of -1 in this context. Define the path polynomial $P^e_d(z) \in \mathbb{Z}[z]$, for any word d over $\mathbf{2}_s$, as follows: $P^e_\varepsilon(z) = 1$ and

$$P^e_{d\delta}(z) = z \cdot (P^e_d(z) + \delta) \tag{1.12}$$

Then $P^e_d(A)$ is a linear map from $\mathbb{Q}^m$ to $\mathbb{Q}^m$ and $P^e_d(A)(e)$ is the state of $\mathfrak{A}(A, e, e)$ after scanning $d \in \mathbf{2}_s{}^\star$, starting at initial state e. We write $P_d(A)$ for the path polynomial for the principal automaton $\mathfrak{A}(A)$, in which case the initial state is $\mathbf{e}_1 = A^{-1} c_1$. Note that all coefficients of a path polynomial encode the corresponding path in an entirely straightforward manner.

The extension $d0$ is valid if, and only if, $P^e_d(A)(e)$ is even; otherwise the valid extensions are $d1$ and $d\overline{1}$. An induction on the length of a path then shows that P_d is defined whenever P^e_d is and that $P_d(A) = P^e_d(A)$. This provides another proof of the claim that the principal automaton $\mathfrak{A}(\mathsf{A})$ is embedded in all complete automata $\mathfrak{C}(\mathsf{A}, e)$.

As will see shortly, path polynomials define algebraic integers in a natural way. To see why, consider the algebraic number field $\mathbb{F} = \mathbb{Q}[z]/\chi(z)$ of degree m. We write α for the representative of $1/z$ of the inverse of a root of $\chi(z)$, and $\chi^\star(z)$ for the minimal polynomial of α, the reciprocal of χ. We have $\mathbb{F} = \mathbb{Q}(\alpha)$, but α is an algebraic integer and more useful for our purposes. Now suppose we have two directed paths $\mathbf{e}_1 \longrightarrow p$, with corresponding path polynomials P_1 and P_2. Then $P_1 = P_2 \pmod{\chi(z)}$. The see this, observe that $P_1(\mathsf{A})(\mathbf{e}_1) = P_2(\mathsf{A})(\mathbf{e}_1)$ implies that $\mathbf{e}_1$ is in the null space of $P = P_1 - P_2$. By Cayley-Hamilton, the remainder operation with respect to $\chi(z)$ does not affect the corresponding linear operators. Hence the null space of $P \pmod{\chi(z)}$ is non-zero. If P is not zero, it has degree less than m and is thus coprime with $\chi(z)$. Hence there are cofactors Q_1 and Q_2 such that $Q_1 P + Q_2 \chi(z) = 1$. But then Q_1 is the inverse of P, contradiction. By slight abuse of notation, we will refer to $P \bmod \chi(z) \in \mathbb{F}$ also as the path polynomial for p. Hence we can label states p in the principal automaton by algebraic integers. Figure 1.7 shows an example of such a labeling.

Path polynomials suggest a generalization where we label a node p in $\mathfrak{C}(\mathsf{A})$ by an algebraic integer $\Phi(p) \in \mathbb{Z}_{\mathbb{F}}$ subject to the constraint

$$p \xrightarrow{d} q \iff \Phi(p) = \alpha\,\Phi(q) - d\beta \tag{1.13}$$

where $\beta = e \circ (\alpha^i)_{i<m}$. An easy induction shows the analogue of equation (1.4):

Lemma 4.

$$\alpha^k \Phi(q_k) = \Phi(q_0) + \beta \sum_{i<k} d_i \alpha^i.$$

Considering the self-loop at 0 it follows that $\Phi(0) = 0$. In particular for $e = \mathbf{e}_1$ we have $\beta = 1$. In this case for source $q_0 = \mathbf{e}_1$ and target $q_k = 0$ we have

$$1 + \sum_{i<k} d_i \alpha^i = 0.$$

Similarly, if there is a cycle of length k at a point q we have

$$(\alpha^k - 1)\,\Phi(q) = \sum_{i<k} d_i \alpha^i.$$

The string d is nothing but the reverse base α expansion of the algebraic integer $\sum_{i<k} d_i \alpha^i$ on the right hand side. It is easy to check that

$$P_d(z) = z^k \left(1 + \sum_{i<k} d_i z^{-1}\right).$$

Thus, path polynomials and labels are closely connected:

Lemma 5. *For any path from $\mathbf{e}_1$ to q labeled d: $P_d(1/\alpha) = \Phi(q)$.*

The polynomial representation is useful because it allows us to use polynomial arithmetic to search for paths. As an example, consider again the characteristic polynomial $\chi(z) = z^2 + z + 1/2$ with principal automaton shown in figure 1.7. There is a trivial path from 0 to $\mathbf{e}_1$ labeled $\overline{1}$, corresponding to $\rho = -1$. This is obviously the shortest path, but we can try to find others by writing

$$\rho = \beta\chi^\star + \sigma \qquad \sigma = \sum s_i \alpha^i \tag{1.14}$$

where the digits s_i are again trits. We have $2 + 2\alpha + \alpha^2 = 0$ in $\mathbb{Z}_{\mathbb{F}}$. Hence we can rewrite the digits string -1 as follows:

1	α	α^2	α^3	α^4
-1				
2	2	1		
	-2	-2	-1	
		2	2	1
1	0	1	1	1

Thus $\beta = 1 - \alpha + \alpha^2$ and $\sigma = 1 + \alpha^2 + \alpha^3 + \alpha^4$. One can easily check in figure 1.7 that this corresponds indeed to the shortest path from 0 to $\mathbf{e}_1$ that passes through $-\mathbf{e}_1$. This approach is similar in spirit to Gilbert's clearing algorithm [9].

Now consider a copy chain of the form

$$q_k \xrightarrow{0} q_{k-} \xrightarrow{0} \dots \xrightarrow{0} q_1 \xrightarrow{0} q_0$$

where q_0 belongs to some subautomaton $\mathcal{A}$ with state set Q, while some or all of the other states in the chain lie outside of $\mathcal{A}$. As we have seen in lemma 3, adjoining these states does not change the group generated by $\mathcal{A}$. In the special case where $q_0 = \mathbf{e}_1$ is the generator of the principal automaton, and there are at least $m - 1$ copy states in the chain, we even have equality: $\mathbb{Z}[q_{m-}, \dots, q_0] = \mathbb{Z}[Q]$. This follows easily by induction on the length of a path from the generator to a state in the automaton. In this case, the label of q_i is simply α^i. To generalize lemma 5 one needs to admit Laurent polynomials as path polynomials. Since A is invertible this causes no difficulties. At any rate, the transduction group generated by $\mathcal{A}$ can thus be represented by a lattice of algebraic integers.

Lemma 6. *Let $\mathcal{A}$ be a canonical automaton generating an m-lattice G. The G is isomorphic to a lattice of algebraic integers of the form $\sum_{i<m} a_i \alpha^i$, $a_i \in \mathbb{Z}$.*

For example, in a cycle-cum-chord automaton the length of the backbone copy chain is always larger than the rank of the lattice. Thus the group is generated by the first m states on the chain.

1.5 Knuth Normal Form

The last lemma suggests that one consider numeration systems for the algebraic integers in an algebraic number field, a subject first breached by Knuth in 1960 [13] in the case of the Gaussian integers $\mathbb{Z}[\mathbf{i}]$. As it turns out, every Gaussian integer ρ can be written uniquely in the form $\rho = \sum_{i<k} b_i \alpha^i$ where $\alpha = -1 + \mathbf{i}$ using only binary digits 0 and 1. Since α is a root of $\chi(z) = z^2 + z + 1/2$, we can translate this result into the world of subau-

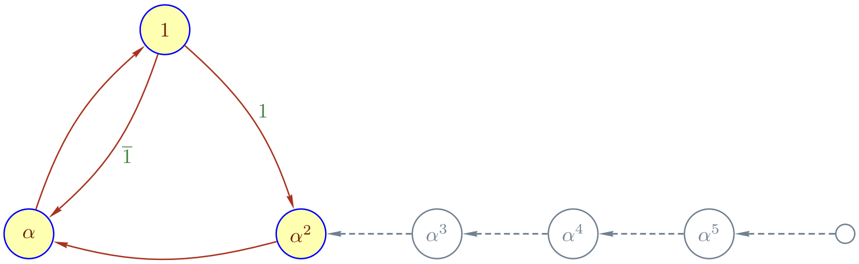

Fig. 1.9 Adding an infinite copy chain to a strongly connected canonical automaton.

tomata as follows [12]. Attach an infinite copy chain to the generator $\mathbf{e}_1$ of CC_2^3 and refer to the resulting automaton as $\mathcal{A}_+$, see figure 1.9. By slight abuse of notation, let us refer to these states by their labels as α^k, with $\alpha^0 = 1$ representing the generator $\mathbf{e}_1$. As we have seen, $\mathcal{A}_+$ still generates the same group. More importantly, every element of the associated 2-lattice can be written uniquely in Knuth normal form (KNF) $\sum_{i<k} b_i \alpha^i$, where $b_i \in \mathbf{2}$, according to [34, 13].

As an example, consider $f = \underline{1}^5$ in the automaton from figure 1.7. Written as a bit-vector, the Knuth normal form of f is 100010111. The next table shows the corresponding rewrite process.

1	α	α^2	α^4	α^4	α^5	α^6	α^7	α^8
5								
-4	-4	-2						
	4	4	2					
		-2	-2	-1				
				2	2	1		
					-2	-2	-1	
						2	2	1
1	0	0	0	1	0	1	1	1

Knuth normal form is extremely helpful in exploring properties of the automorphisms generated by $\mathcal{A}$, rather than just their group structure. For example, one can show that $\underline{1}^{2^{4k}}$ has normal form α^{8k}, a map that copies the first $8k$ input bits and then behaves like the odd transduction $\underline{1}$ on the remainder of the input. This can be used to show that Knuth normal form for this particular automaton can be computed by a suitable finite state transducer [31]. Similarly one can show that the group of automorphisms acts transitively on each level set $\mathbf{2}^k$ and a little more work makes it possible to identify the levels where the group acts simply transitively. To construct Schreier graphs like the one in figure for $\mathbf{2}^k$ one can use the fact that in Knuth normal form the generators look like $0^i 1 0^j \in \mathbf{2}^k$.

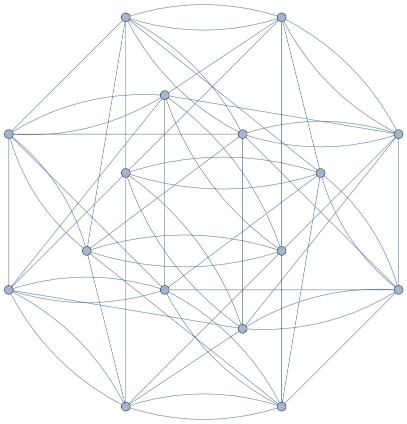

Fig. 1.10 A Schreier graph associated with CC_2^3 and the subtree $\mathbf{2}^4$.

Uniqueness of Knuth normal form is not hard to see: assume $\sum_{i<k} b_i\alpha^i = \sum_{i<k} b'_i\alpha^i$ for digits $b_i, b'_i \in \mathbf{2}$. Then $b_i = b'_i$ for all $i < \ell$, but, say, $0 = b_\ell \neq b'_\ell = 1$ and we have $\sum_{i>\ell} b_i\alpha^i = \alpha^\ell + \sum_{i>\ell} b'_i\alpha^i$. But the first automorphism copies at least ℓ bits, whereas the second changes the bit in position ℓ.

Existence is much harder to deal with. For example, it is known that for $m = 2$ there are only six characteristic polynomials that give rise to residuation matrices; all of these are unique up to $\mathsf{GL}_m(\mathbb{Z})$ similarity. The matrices together with the corresponding α values and their minimal polynomials are shown below. The first matrix, which gives rise to the successor automaton, is the only one that produces a real field. The forth matrix is associated with the automaton in figure 1.3. As the table shows, only 4 out of the 6 admit a Knuth normal form.

A	α	min. pol.	KNF
$\left(\begin{smallmatrix}0&1\\1/2&0\end{smallmatrix}\right)$	$-\sqrt{2}$	$-2+z^2$	no
$\left(\begin{smallmatrix}1&1\\-1/2&0\end{smallmatrix}\right)$	$1+i$	$2-2z+z^2$	no
$\left(\begin{smallmatrix}1/2&1\\-1/2&0\end{smallmatrix}\right)$	$\left(1+\mathbf{i}\sqrt{7}\right)/2$	$2-z+z^2$	yes
$\left(\begin{smallmatrix}0&1\\-1/2&0\end{smallmatrix}\right)$	$i\sqrt{2}$	$2+z^2$	yes
$\left(\begin{smallmatrix}-1/2&1\\-1/2&0\end{smallmatrix}\right)$	$\left(-1+\mathbf{i}\sqrt{7}\right)/2$	$2+z+z^2$	yes
$\left(\begin{smallmatrix}-1&1\\-1/2&0\end{smallmatrix}\right)$	$-1+i$	$2+2z+z^2$	yes

In the case where all coefficients of the minimal polynomial of α are nonnegative, one can show that the Gilbert-style rewrite process always terminates.

More precisely, we can interpret equation (1.5) as a cancellation rule that simplifies some expressions. Applying equation (1.5) twice we obtain the shift rule

$$2\alpha \mapsto (2 - g_{m-1})\alpha^2 + (g_{m-1} - g_{m-2})\alpha^3 + \ldots + g_0\alpha^{m+2} \tag{1.15}$$

that can be used to eliminate coefficients other than 0 and 1. Given an algebraic integer $\rho = \sum_{i<k} a_i\alpha^i$, we may assume that all coefficients are nonnegative. Let us refer to $\sum_{i<k} a_i$ as the weight of the integer. Clearly, application of the cancellation rule reduces weight. But note that the sum of exponents in the shift rule is telescoping, so that its application does not affect weight. Now consider a rewrite system that tries to remove all coefficients other than 0 and 1 by first applying the cancellation rule, and then the shift rule. Assume that the rewrite process fails to terminate on some input ρ. By deleting an initial segment, we may safely assume that the weight of the expression remains constant throughout the process, i.e., we only apply the shift rule. Using the notation from equation (1.14) we see that after a transient phase, only the last $m+1$ coefficients of σ will be non-zero. Since the weight does not change, this block of coefficients must ultimately repeat in some later term σ'. But that means that $\sigma' = \alpha^k\sigma$, a contradiction.

The copy chain extension to the principal automaton is perhaps the most natural way to define Knuth normal form, but there are other options. For example, we can use the generator of one of the strongly connected automata in $\mathfrak{C}(A, e)$ that generate the lattice as an anchor for the chain. For CC_2^3 this is indicated in figure 1.9. The Knuth normal forms of the automorphisms $\underline{1}^{\,k}$ for $0 \le k < 2^{10}$ are shown in figure 1.11. The height of each column indicates the number of terms, and the actual terms are color coded.

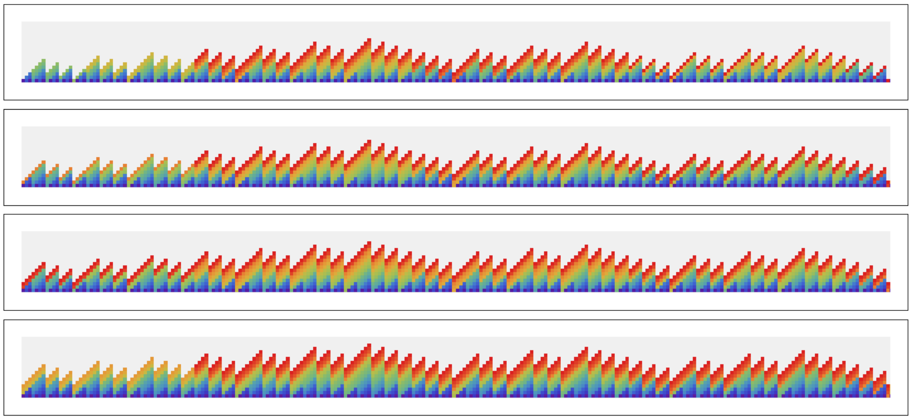

Fig. 1.11 The Knuth normal forms for some automorphisms defined by a cycle-cum-chord transducer.

In cases where Knuth normal form fails to exist one can try allow for larger digits set, and in particular for trits $\{-1, 0, 1\}$. Define the weak Knuth normal form to be an expansion of the form $\sum_{i<k} d_i \alpha^i$ where $d_i \in \mathbf{2}_s$. This generalization corresponds to the step from canonical numeration systems for algebraic number fields, where the digit set is of the form $\{0, 1, \ldots, N^-\}$ to symmetric canonical number systems where the digits are chosen from $\{0, \pm 1, \ldots, \pm N^-\}$. Here N is typically the absolute value of the absolute norm of the generator α and thus $N^- = 1$ in our case, There is large body of literature on these numeration systems, see [25] and the references therein. Call an algebraic integer expanding if all its conjugates have modulus larger than 1. The following result follows from the work in [14, 15].

Theorem 2 (Lagarias, Wang 1997). *Let α be an expanding algebraic integer of norm 2. Then all algebraic integers ρ in $\mathbb{F} = \mathbb{Q}(\alpha)$ have an expansion $\rho = \sum_{i<k} d_i \alpha^i$ where $d \in \{-1, 0, 1\}$.*

Corollary 1. *All m-lattices admit a weak Knuth normal form.*

Of course, we no longer have uniqueness. For example, for the 2-lattice associated with $\alpha = 1 + \mathbf{i}$ the non-trivial weak normal forms of $\overline{1} = -1$ are $1011 \ldots 11\overline{1}1$. Another possibility is to move to the completion of the group and consider infinite normal forms $\sum_i d_i \alpha^i$: the automorphism corresponding to digit d_k leaves the first k input bits unchanged, so this formal infinite sum indeed defines an automorphism of $\mathbf{2}^\star$ [22]. For example, for the successor automaton from figure 1.2 with characteristic polynomial $\chi(z) = z^2 + 1/2$, the automorphism $\underline{1}^{-1}$ produces the infinite digit sequence $(1, 0, 1, 0, 1, \ldots)$.

1.6 Open Problems

It is shown in [22] that the nucleus automaton is a natural finite subautomaton of the complete automaton $\mathfrak{C}(\mathsf{A}, e)$. Alas, from the perspective of automata theory, the nucleus hides a lot of interesting fine structure. As we have seen, there may be smaller subautomata that also generate the same m-lattice; the cycle-cum-chord transducer CC_2^3 being a case in point.

Question: Is there a reasonable description of the smallest subautomaton of $\mathfrak{C}(\mathsf{A}, e)$ that generates the full lattice? Is its state complexity computable in polynomial time?

Note that the algorithm to compute the nucleus outlined above is clearly not polynomial in A. On the other hand, the principal automaton might be computable in polynomial time.

The question arises at to what the essential differences between the principal automaton and strongly connected subautomata of the nucleus might be. Let us call an automorphism f tame if there is some word w such that $\partial_w f = I$, and strongly tame if all its residuals are tame. Thus f is tame if, and only if, the corresponding state p in $\mathfrak{C}(\mathsf{A}, e)$ is co-reachable from 0. Similarly, f is strongly tame if, and only if, every state reachable from the corresponding state p in $\mathfrak{C}(\mathsf{A}, e)$ is co-reachable from 0. If a principal automaton $\mathcal{A}$ has the two component structure just described, then all its basic transductions are obviously strongly tame. But the same holds true for the whole group generated by $\mathcal{A}$: any composition of basic transductions can be residuated to I by systematic removal of components in the representation of f. By contrast, all the basic transductions defined by canonical automata of the single component type such as CC^n_m fail to be tame. Note, though, that this classification depends crucially on the fact that there are final strongly connected components in the principal automaton other than the trivial one. Otherwise, an automorphism defined by a state in such a putative final component obviously fails to be tame. Computational evidence supports the following conjecture.

Conjecture: Every principal automaton $\mathfrak{A}(\mathsf{A})$ other than a successor automaton is skew-symmetric and consists of exactly two strongly connected components, one of them being the sink 0. For successor automata, the principal automaton has 3 strongly connected components.

Note that it suffices to show that in all these principal automata there is a path from $\mathbf{e}_1$ to $-\mathbf{e}_1$: this would suffice to rewrite any improper path polynomial as a proper one. Equivalently, one needs to rewrite the trit representation of $\overline{1}$ to a form $(1, d_2, d_3, \dots, d_\ell, 1)$. In case the conjecture fails in this strong form, one might wish to consider the situation where the underlying residuation matrix has only one similarity type.

Acknowledgments It is a pleasure to acknowledge many helpful conversations with Tsutomo Okano and Tim Becker.

References

1. S. Abramsky. A structural approach to reversible computation. *Theo. Comp. Science*, 347:441–464, 2005.
2. L. Bartholdi, I. I. Reznykov, and V. I. Sushchansky. The smallest mealy automaton of intermediate growth. *J. Algebra*, 295:387–414, 2005.
3. L. Bartholdi and P. V. Silva. Groups defined by automata. *CoRR*, abs/1012.1531, 2010.
4. C. H. Bennett. Logical reversibility of computation. *IBM journal of Research and Development*, 17:525–532, 1973.
5. C. H. Bennett. The thermodynamics of computation—a review. *IJTP*, 21(12):905–940, 1982.

6. J. Berstel. Transductions and context-free languages. `http://www-igm.univ-mlv.fr/~berstel/LivreTransductions/LivreTransductions.html`, 2009.
7. S. Eilenberg. *Automata, Languages and Machines*, volume A. Academic Press, 1974.
8. C. C. Elgot and J. E. Mezei. On relations defined by generalized finite automata. *IBM J. Res. Dev.*, 9:47–68, January 1965.
9. W. J. Gilbert. Radix representations of quadratic fields. *J. Math. Anal. Appl.*, pages 264–274, 1981.
10. R. R. Grigorchuk, V. V. Nekrashevich, and V. I. Sushchanski. Automata, dynamical systems and groups. *Proc. Steklov Institute of Math.*, 231:128–203, 2000.
11. J. E. Hopcroft and J. D. Ullman. *Introduction to Automata Theory, Languages and Computation*. Addison-Wesley, 1979.
12. D. Knuth. Private communication, 2010.
13. D. E. Knuth. An imaginary number system. *Comm. ACM*, 3:245–247, 1960.
14. J. C. Lagarias and Y. Wang. Self-affine tiles in $\mathbb{R}^n$. *Adv. Math.*, 121:21–49, 1996.
15. J. C. Lagarias and Y. Wang. Integral self-affine tiles in $\mathbb{R}^n$ ii. lattice tilings. *J. Fourier Anal. Appl.*, 3(1):83–102, 1997.
16. R. Landauer. The physical nature of information. *Phy. Lett. A*, 217:188–193, 1996.
17. Y. Lecerf. Machine de Turing réversible. Insolubilité récursive en $n \in N$ de l'équation $u = \theta^n u$, où θ est un "isomorphisme de codes". *C. R. Acad. Sci. Paris*, 257:2597–2600, 1963.
18. K. Morita. Computation universality of one-dimensional reversible cellular automata. *Information Processing Letters*, 42:325–329, 1992.
19. K. Morita. Reversible cellular automata. *J. Information Processing Society of Japan*, 35:315–321, 1994.
20. K. Morita. *Encyclopedia of Complexity and System Science*, chapter Reversible Cellular Automata. Springer, Berlin, 2009.
21. K. Morita and M. Harao. Computation universality of 1 dimensional reversible (injective) cellular automata. *Transactions Institute of Electronics, Information and Communication Engineers, E*, 72:758–762, 1989.
22. V. Nekrashevych. *Self-Similar Groups*, volume 117 of *Math. Surveys and Monographs.* AMS, 2005.
23. V. Nekrashevych and S. Sidki. *Automorphisms of the binary tree: state-closed subgroups and dynamics of 1/2-endomorphisms.* Cambridge University Press, 2004.
24. T. Okano. Invertible binary transducers and automorphisms of the binary tree. MS Thesis, CMU, May 2015.
25. A. Pethö. Connections between power integral bases and radix representations in alebraic number fields. `https://arato.inf.unideb.hu/petho.attila/cikkek/cnsnagoya_paper_110.pdf`, 2009.
26. G. N. Raney. Sequential functions. *J. Assoc. Comp. Mach.*, 5(2):177–180, 1958.
27. J. Sakarovitch. *Elements of Automata Theory.* Cambridge University Press, 2009.
28. J.-P. Serre. *Arbres, Amalgames, SL_2*. Number 46 in Astérisque. Société Mathématique de France, Paris, 1977.
29. S. Sidki. Automorphisms of one-rooted trees: Growth, circuit structure, and acyclicity. *J. Math. Sciences*, 100(1):1925–1943, 2000.
30. K. Sutner. Invertible transducers and iteration. In H. Juergensen and R. Reis, editors, *Descriptional Complexity of Formal Systems*, volume 8031 of *Lecture Notes in Computer Science*, pages 18–29. Springer Berlin, 2013.
31. K. Sutner. Invertible transducers, iteration and coordinates. In S. Konstantinidis, editor, *CIAA*, volume 7982 of *LNCS*, pages 306–318. Springer Berlin, 2013.
32. K. Sutner. Iteration of invertible transductions. Submitted, 2013.
33. K. Sutner and K. Lewi. Iterating invertible binary transducers. *JALC*, 17(2-4):293–213, 2012.
34. K. Sutner and K. Lewi. Iterating invertible binary transducers. In M. Kutrib, N. Moreira, and R. Reis, editors, *Descriptional Complexity of Formal Systems*, volume 7386 of *Lecture Notes in Computer Science*, pages 294–306. Springer Berlin, 2012.