Theoretical Computer Science 37 (1985) 51-75
North-Holland

THE THEORY OF ENDS, PUSHDOWN AUTOMATA, AND SECOND-ORDER LOGIC

David E. MULLER and Paul E. SCHUPP

Department of Mathematics, Computer Science and Coordinated Science Laboratory, University of Illinois, Urbana-Champaign, IL 61801, U.S.A.

Communicated by D. Perrin
Received June 1984

Abstract. A class of edge-labeled graphs called *context-free* are defined according to their behavior at infinity. Such graphs are generalizations of Cayley graphs of context-free groups. They are also shown to be definable in a very natural way in terms of push-down automata. Using Rabin's theorem on the monadic second-order theory of the infinite binary tree, these graphs are also shown to have a decidable monadic second-order theory. Questions about tiling systems and cellular automata operating on these graphs are decidable even when the analogous questions are not, when the systems operate on a two-dimensional grid.

Contents

1. Introduction

The theory of ends studies 'ways of going to infinity'. In this paper we use the theory of ends to define a large class of graphs which we call context-free graphs. Roughly, the idea is that a context-free graph is 'finitely behaved at infinity'. One of our main results is that a graph Γ is context-free if and only if Γ is the complete transition graph of some pushdown automaton. Another of our main results consists of using this characterization and Rabin's theorem that the monadic second-order theory of the infinite binary tree is decidable to show that the monadic second-order theory of any context-free graph is decidable. In establishing this result we show that the monadic second-order theory of a graph Γ is equivalent to certain problems about tilings of Γ. For a context-free graph, these problems are reducible to problems on the binary tree and are thus solvable by Rabin's theorem. Cellular automata and vector addition systems are usually considered as involving the grid of integer lattice points in n-dimensional space. We show that such systems are quite natural on a very wide class of graphs and, in contrast to the classical case, all the relevant algorithmic problems about such systems are solvable in context-free graphs. An outline of these results was given by the present authors in a proceedings paper [6].

2. Context-free graphs and graphs of pda's

In this section we do the following. First, with any pushdown automaton M we associate a connected labeled graph $\Gamma(M)$. The graph $\Gamma(M)$ is really the complete transition graph of the machine M in the sense that $\Gamma(M)$ is a picture of all the possible total states of M and all the ways in which these total states can be reached. We also define abstractly a class of connected labeled graphs which we call 'context-free'. The underlying motivation for this definition comes from the theory of ends. We shall prove that a graph Γ is context-free if and only if Γ is the graph of some pda.

We first fix our notation and conventions on labeled graphs. An *alphabet* is simply a finite set of symbols. An *augmented alphabet* Σ is an alphabet together with an involution i: $\Sigma \to \Sigma$. As usual, we write σ^{-1} for $i(\sigma)$ and say that σ^{-1} is the *inverse* of σ. A *labeled graph* Γ consists of a set V of vertices, an augmented *label alphabet* Σ, and a set E of *edges* where E is a subset of $V \times \Sigma \times V$ with the property that if (u, σ, v) is in E, then (v, σ^{-1}, u) is also in E. If $e = (u, \sigma, v)$ is an edge, then u is the *initial vertex* of e, v is the *terminal vertex* of e, and σ is the *label* of e. We say that the edge $e^{-1} = (v, \sigma^{-1}, u)$ is the *inverse* of e.

Our graphs thus allow more than one edge with the same initial and terminal vertices, but distinct such edges must have distinct labels. The initial and terminal vertices of an edge e may be the same (that is, e is a loop at a vertex). An *undirected edge* is a pair of edges of the form $\{e, e^{-1}\}$. In drawing graphs we follow the usual convention of drawing only one segment for each undirected edge and think of the segment as capable of being traversed in either of two directions.

A *path* α in Γ is a sequence $e_1, \ldots, e_k$, $k \geq 0$, of edges such that the initial vertex of e_{i+1} is the terminal vertex of e_i, for $i = 1, \ldots, k-1$. We say that α has *length* k and that the *label* of α is the word $\sigma_1, \ldots, \sigma_k$ where σ_i is the label of e_i. We adopt the convention that there is an *empty path* of length 0 from any vertex to itself. The label of the empty path is the empty word Λ. The graph Γ is *connected* if for every pair u and v of vertices there exists a path with initial vertex u and terminal vertex v. The *distance*, $d(u, v)$, between u and v is the minimum length of a path from u to v. The *degree* of a vertex u is the number of edges having u as an initial vertex.

An *isomorphism* ψ between labeled graphs Γ and Γ' is a one-to-one correspondence ψ: $V \to V'$ between the vertex sets of Γ and Γ' which induces a label-preserving one-to-one correspondence between the edge sets E and E'. (If $e = (u, \sigma, v)$, then $\psi(e) = (\psi(u), \sigma, \psi(v))$ and we require that $\psi: E \to E'$ is a one-to-one correspondence.)

We shall work only with the class of graphs delimited by the next definition.

2.1. Definition. A *finitely generated graph* is a labeled graph Γ having the following properties:

(1) Γ is a connected graph with a distinguished vertex v_0 called the *origin* of Γ.

(2) Γ has uniformly bounded degree b (that is, the degree of any vertex does not exceed b).

(3) The label alphabet, Σ, of Γ is finite.

For example, the Cayley graph of any finitely generated group, say with the identity specified as the origin, is a finitely generated graph.

We assume that the reader is familiar with the concept of a pushdown automaton. For an excellent introduction to the subject, see [5, Chapter 5]. In our conventions and notation concerning pushdown automata we follow Harrison [4]. We abbreviate 'pushdown automaton' by pda. If ζ denotes the contents of the stack at a given time, we consider the rightmost symbol of ζ to be the top symbol of the stack. We use Λ to denote the empty string. We differ from the usual convention in that we allow our machines to continue running when they empty the stack. Indeed, we alloy a machine to start with an empty stack.

As usual, a general pushdown automata is nondeterministic and we consider a deterministic machine to be a special case of a nondeterministic machine. To fix our notation, we consider a pushdown automaton to be a system $M = (Q, \Sigma, Z, \delta, q_0, z_0, \hat{Q})$, where Q is a finite set of *states*, Σ is a finite *input alphabet*, Z is a finite *stack alphabet*, $q_0 \in Q$ is the *initial state*, $z_0 \in Z \cup \{\Lambda\}$ is *start symbol*, $\hat{Q} \subseteq Q$ is the set of *final states*, and δ is the *transition function*. Since we allow a machine to continue running when it empties its stack, δ is a mapping from $Q \times (\Sigma \cup \{\Lambda\}) \times (Z \cup \{\Lambda\})$ to finite subsets of $Q \times Z^*$.

The interpretation of

$$\delta(q, \sigma, z) = \{(q_1, \zeta_1), \ldots, (q_m, \zeta_m)\}$$

where the $q_i \in Q$, $\sigma \in \Sigma$, $z \in Z$, and each $\zeta_i \in Z^*$, $1 \leq i \leq m$, is that when the pda is in state q reading the input symbol σ and z is the top symbol on the stack, then the machine can, for any choice of $i = 1, \ldots, m$, change to state q_i, replace z by ζ_i, and move the head reading the input tape one square to the right. We consider the symbols in ζ_i as being placed on the stack from left to right so that the rightmost symbol of ζ_i (if $\zeta_i \neq \Lambda$) becomes the top of the stack. If $\delta(q, \sigma, z)$ is empty, then the machine halts.

The interpretation of

$$\delta(q, \sigma, \Lambda) = \{(q_1, \zeta_1), \ldots, (q_m, \zeta_m)\}$$

is that when the pda is in state q reading the input symbol σ and the stack is empty, then the machine can change state to q_i and add ζ_i to the stack for a choice of i, $1 \leq i \leq m$.

The interpretation of

$$\delta(q, \Lambda, z) = \{(q_i, \zeta_i), \ldots, (q_m, \zeta_m)\}$$

is that when the pda is in state q with z the top symbol of the stack, then, independently of the input symbol being scanned, the machine can change to state q_i and replace z by ζ_i. In this case the input head is not moved. Such a transition of M is called a *Λ-move*. In the obvious way, we also allow Λ-moves when the stack is empty.

A *total state* of the pda M consists of a pair (q, ζ) where $q \in Q$ is the current state of M and $\zeta \in Z^*$ denotes the contents of the stack. We write $(q, \zeta, z) \vdash_M^{\sigma} (q', \zeta\zeta')$ if $\delta(q, \sigma, z)$ contains (q', ζ'). We write $M \vdash^* (q, \zeta)$ if there exists some word $w \in \Sigma^*$ such that when M is started in its initial total state (q_0, z_0) with w written on the input tape it is possible for M to be in the total state (q, ζ) after reading w.

We now turn to describing the graph $\Gamma(M)$ of M. The vertex set of Γ is

$$V = \{(q, \zeta) \mid q \in Q, \zeta \in Z^*, M \vdash^* (q, \zeta)\}.$$

If (q, ζ) and (q', ζ') are possible total states of M and M can go from (q, ζ) to (q', ζ') in a single transition, that is, if $(q, \zeta) \vdash_M^{\sigma} (q', \zeta')$, then there is an edge e with label σ from (q, ζ) to (q', ζ').

In order to introduce inverse edges and their labels we distinguish two cases. If one is discussing an arbitrary pda M, the input alphabet Σ of M has no structure. In order to label $\Gamma(M)$ it is necessary to extend Σ to an augmented alphabet. On the other hand, there are cases, such as that when M accepts the word problem of a group or when M is associated with a graph, where the input alphabet of M is already augmented. We thus distinguish between a pda and a pda with an augmented input alphabet and proceed as follows. If the input alphabet $\Sigma = \{\sigma_1, \ldots, \sigma_m\}$ is not augmented, we adjoin new symbols $\sigma_1^{-1}, \ldots, \sigma_n^{-1}$ to Σ. If Σ is already augmented we do nothing. In either case, if the pda M can make Λ-moves, we also adjoin new symbols λ and λ^{-1} to the alphabet. We denote the final alphabet by $\Sigma^{\pm 1}$, and we can write $\Sigma^{\pm 1} = \{\sigma_1, \ldots, \sigma_n, \sigma_1^{-1}, \ldots, \sigma_n^{-1}\}$ where possibly σ_n is the symbol λ.

If (q, ζ) and (q', ζ') are in V and $\sigma \in \Sigma^{\pm 1}$, and $(q, \zeta) \vdash_M^{\sigma} (q', \zeta')$, then there is an edge e with label σ from (q, ζ) to (q', ζ'). These edges correspond to the transitions of M. We also introduce an edge e^{-1} with label σ^{-1} from (q', ζ') to (q, ζ). We say that such an edge corresponds to a *reverse transition* of M.

The origin of $\Gamma(M)$ is the initial total state (q_0, z_0) of M. It is clear that all the requirements of $\Gamma(M)$ being a finitely generated graph are met. Note that the graph $\Gamma(M)$ does not depend on the set of final states of M and is thus not closely correlated with the language which M accepts.

As a first example, consider the simple pda M with $Q = \{q_0\}$ and $\Sigma = Z = \{0, 1\}$. M starts with empty stack and on reading an input symbol simply adds that symbol to the stack. It is clear that the graph $\Gamma(M)$ is the infinite binary tree.

In Fig. 1 we have labeled vertices by the corresponding stack contents but note that the vertices are *not* labeled in the actual graph Γ.

As another example, consider the pda M with $\Sigma = \{a, b\}$, $Z = \{z\}$ and $Q = \{q_0, q_1\}$ where the action of M is the following. M starts in state q_0 with empty stack. In either state q_0 or q_1, on reading the input a, M does not change state and adds z

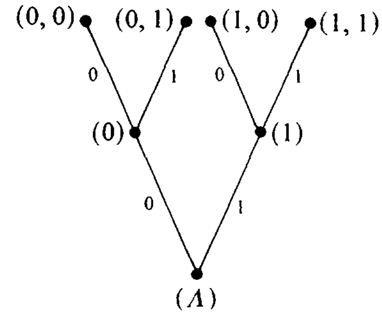


Fig. 1.

to the stack. When in state q_0, on reading input b, M changes states to q_1 and leaves the stack unchanged. When in state q_0, M can also make a Λ-move which consists of changing state to q_1 and removing the top symbol from the stack. Formally,

$$(q_1, \zeta) \vdash^{a} (q_i, \zeta z), \quad i = 1, 2, \qquad (q_0, \zeta) \vdash^{b} (q_1, \zeta) \qquad (q_0, \zeta z) \vdash^{\Lambda} (q_1, \zeta).$$

The augmented alphabet is $\Sigma^{\pm 1} \equiv \{a, b, \lambda, a^{-1}, b^{-1}, \lambda^{-1}\}$. The graph $\Gamma(M)$ is the 'one-way infinite braced ladder' illustrated in Fig. 2.

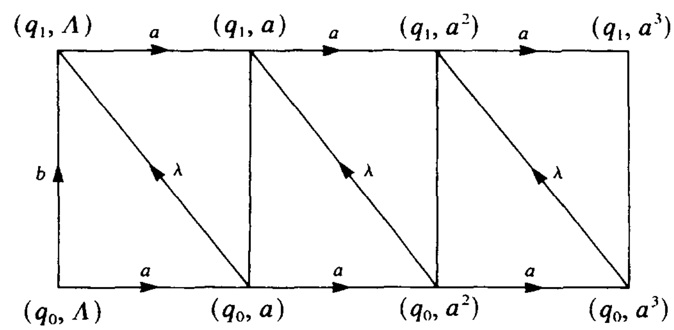


Fig. 2.

Let Γ be a finitely generated graph with origin v_0. If v is any vertex of Γ, we use $|v|$ to denote the length of a shortest path from v_0 to v. By $\Gamma^{(n)}$ we mean the subgraph of Γ consisting of all the vertices and edges which are connected to v_0 by a path of length less than n. (Thus $\Gamma^{(0)}$ is empty and $\Gamma^{(1)}$ consists of v_0 and incident edges.) We now want to consider the connected components of $\Gamma \backslash \Gamma^{(n)}$. (The motivation here comes directly from the theory of ends. Compare [1, 8]. The numbers of ends of a connected locally finite graph Γ is the limit as n goes to infinity of the number of infinite components of $\Gamma \backslash \Gamma^{(n)}$.)

If C is a component of $\Gamma \backslash \Gamma^{(n)}$, a *frontier point* of C is a vertex u of C such that $|u| = n$. If v is a vertex of Γ, say with $|v| = n$, then we shall use $\Gamma(v)$ to denote the component of $\Gamma \backslash \Gamma^{(n)}$ which contains v. In other words, $\Gamma(v)$ is the subgraph of Γ consisting of all vertices v' which can be connected to v by paths all of whose vertices lie at distance at least $|v|$ from the origin and the edges connecting two such vertices. The set of frontier points of $\Gamma(v)$ will be denoted by $\Delta(v)$. Since Γ has uniformly bounded degree each set $\Delta(v)$ is finite.

Let u and v be vertices of Γ. An *end-isomorphism* between the two subgraphs $\Gamma(u)$ and $\Gamma(v)$ is a mapping ψ between $\Gamma(u)$ and $\Gamma(v)$ such that

(i) ψ is a label-preserving graph isomorphism,

(ii) ψ maps $\Delta(u)$ onto $\Delta(v)$.

2.2. Definition. A graph Γ is *context-free* if Γ is a finitely generated graph such that

$$\{\Gamma(v) \mid v \text{ a vertex of } \Gamma\}$$

has only finitely many isomorphism classes under end-isomorphisms.

In order to illustrate the above definition, consider the labeled infinite binary tree T, as depicted in Fig. 3.

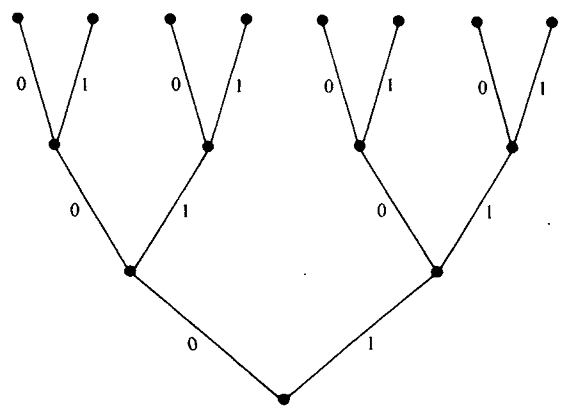


Fig. 3.

There are 2^n components of $T \setminus T^{(n)}$ but there is only one isomorphism class. Regardless of the value of n, a component of $T \setminus T^{(n)}$ is end-isomorphic to the whole tree T.

If one takes Γ to be the graph $\Gamma(M)$ of Fig. 2 it is easy to see that there is only one component of $\Gamma \setminus \Gamma^{(n)}$ for any n. There are two isomorphism classes, that of the whole graph and the isomorphism type of $\Gamma \setminus \Gamma^{(n)}$ for $n \geqslant 1$. A component $\Gamma \setminus \Gamma^{(n)}$, $n \geqslant 1$, has two frontier points.

Given a context-free graph Γ we shall prove that there exists a pda M of a very special type, which we shall call a *canonical pda*, such that Γ is isomorphic to $\Gamma(M)$. In reading the proof of the next lemma, the reader might keep the example of the binary tree in mind.

2.3. Lemma. *Let Γ be a context-free graph with origin v_0. Then there is a canonical pda for Γ.*

Proof. Let $G = \{\Gamma(v) \mid v \in \Gamma\}$, and let $D = \{\Delta(v) \mid v \in \Gamma\}$. We write $\Gamma(u) \sim \Gamma(v)$ if $\Gamma(u)$ and $\Gamma(v)$ are end-isomorphic. The hypothesis that Γ is context-free says that the number of equivalence classes under the relation $\sim$ is finite. Let $[G] =$

$\{\Gamma_0, \dots, \Gamma_k\}$ be a complete set of representatives of the equivalence classes with $\Gamma_0 \sim \Gamma$. If $\Gamma(u)$ is in the equivalence class represented by Γ_i, there is an end-isomorphism which maps $\Delta(u)$ onto the set Δ_i of frontier points of Γ_i. Let $[D] = \{\Delta_0, \dots, \Delta_k\}$.

Since each Δ_i is finite, we can define $m = \max\{|\Delta_i|; 0 \leq i \leq k\}$. We also fix an ordering $\{p_{i,0}, \dots, p_{i,l(i)}\}$ of the vertices of each Δ_i, $i = 0, \dots, k$.

Let $Q = \{q_0, \dots, q_{m-1}\}$ be a set of m states. Our pda M will have a stack alphabet Z consisting of symbols z_{ij} where the first subscript has range $i = 0, \dots, k$. The range of the second subscript depends on the value of i and will be described shortly. The initial total state of M is (q_0, z_{00}).

Let Γ_i be a representative graph with Δ_i its set of frontier points. If the edges of Γ_i which are incident to some vertex of Δ_i are deleted, one is left with a finite set of connected subgraphs of Γ_i. We call these subgraphs the *second level subgraphs* of Γ_i. (If u is a vertex of Γ and $\Gamma(u) \sim \Gamma_i$, with $|u| = n$ say, then these subgraphs will represent those components of $\Gamma \backslash \Gamma^{(n+1)}$ which are contained in $\Gamma(u)$.) It may be that no second-level subgraph is end-isomorphic to the whole graph Γ_0. If this happens, we declare Γ_0 to also be a second-level subgraph.

A second-level subgraph L is end-isomorphic to some graph $\Gamma_i \in [G]$. There may be more than one end-isomorphism between L and Γ_i. We now fix an end-isomorphism between each second-level subgraph and a graph in $[G]$.

A given representative graph Γ_i may have distinct second-level subgraphs which are isomorphic to the same Γ_i in $[G]$. We assign indices to second-level subgraphs as follows. To each second-level subgraph isomorphic to Γ_i we assign first index i. Fixing i, we assign a different second index j to each distinct second-level subgraph which arises from the same Γ_i and is isomorphic to Γ_i. This j is no larger than the maximum over i of the number of second-level subgraphs of Γ_i which are isomorphic to each other. From now on we denote second-level subgraphs as $\Gamma_{i,j}$. We associate a stack symbol $z_{i,j}$ with each second-level subgraph $\Gamma_{i,j}$ and let Z denote the set of all such stack symbols.

Each vertex v of Γ with $|v| = n$ is a frontier point of some component of $\Gamma \backslash \Gamma^{(n)}$. Our pda will work outwards from the origin v_0. The string of pushdown symbols will specify which components are entered in going from v_0 to a vertex v and the state of the pda will specify which frontier point in the current component corresponds to v.

We assume inductively that for some $d \geq 0$ we have defined a pda M_d with state set Q, input alphabet $\Sigma^{\pm 1}$, stack alphabet Z, and initial total state (q_0, z_{00}). The initial pda M_0 has no transitions.

In order to motivate the statement of our general inductive assumptions, we first discuss the construction of M_1. Suppose that u' is a vertex at distance one from v_0. Then there is an edge $e = (v_0, \sigma, u')$ in Γ. Suppose that u' is in the second-level subgraph $\Gamma_{i',j'}$ of $\Gamma_0 \sim \Gamma(v_0)$. Also, suppose that u' corresponds to the vertex $p_{i',l}$ of $\Delta_{i'}$ under the fixed end-isomorphism between $\Gamma_{i',j}$ and $\Gamma_{i'}$. Then we make $(q_0, z_{00}) \vdash^{\sigma} (q_l, z_{00}z_{i',j'})$ a transition of M_1. Thus each vertex u' with $|u'| = 1$ will correspond to a unique total state of M_1 in which the length of the stack is two. We also possibly need to add transitions corresponding to edges joining two vertices at distance one from v_0.

Precisely, we assume that for some $d \geqslant 0$ we have defined a pda M_d having the following properties.

(1) Each vertex u of Γ with $|u| \leqslant d$ has been placed in correspondence with a unique possible total state (q, ζ) of M_d such that $|\zeta| = |u| + 1$. If $|u| = |u'| = n \leqslant d$ and (q, ζ) and (q', ζ') are the total states corresponding to u and u' respectively, then $\zeta = \zeta'$ if and only if u and u' are in the same component of $\Gamma \backslash \Gamma^{(n)}$. If u corresponds to the total state (q, ζ) of M and α is a path of shortest length from v_0 to u and α has label w, then $(q_0, z_{00}) \vdash^{w}_{M_d} (q, \zeta)$.

(2) M_d makes no Λ-moves. A transition of M_d either adds exactly one letter to the stack or leaves the stack unchanged. M_d has a transition of the form $(q, z_{ij}) \vdash^{\sigma} (q', z_{ij} z_{i'j'})$ if and only if there are vertices u and u' of Γ such that $|u| + 1 = |u'| \leqslant d$, u corresponds to a total state of the form (q, ζ, z_{ij}), there is an edge $e = (u, \sigma, u')$, and u' is in the second-level subgraph of $\Gamma(u)$ corresponding to $\Gamma_{i'j'}$. M_d has a transition of the form $(q, z_{ij}) \vdash^{\sigma} (q', z_{ij})$ if and only if there are vertices u and u' such that $|u'| = |u| = n \leqslant d$ and there is an edge $e = (u, \sigma, u')$. (Since this condition is symmetric in u and u', M_d must also contain the transition $(q', z_{ij}) \vdash^{\sigma^{-1}} (q, z_{ij})$.)

We construct the pda M_{d+1} as follows. Suppose that there is an edge $e = (u, \sigma, u')$ of Γ with $|u| = d$ and $|u'| = d + 1$. Let (q, σ, z_{ij}) be the total state corresponding to u. Suppose that the second-level subgraph of $\Gamma(u) \sim \Gamma_i$ which contains u' is Γ_{ij}, and that u' corresponds to the vertex $p_{i,l}$ of Δ_i, under the fixed end-isomorphism taking $\Gamma_{i'j'}$ into $\Gamma_{i'}$. If it is not already present, add the transition $(q, z_{ij}) \vdash^{\sigma} (q_l, z_{ij} z_{i'j'})$ to the transition of M_{d+1}.

It may be that there are vertices u' and u'' with $|u'| = |u''| = d + 1$ which are connected by an edge $e = (u', \sigma, u'')$. In this case, u and u' are in the same component of $\Gamma \backslash \Gamma^{(d+1)}$. By the previous part of the construction of M_{d+1}, u' and u'' correspond respectively to total states of the form $(q', \zeta' z_{i'j'})$ and $(q'', \zeta' z_{i'j'})$ which differ only in q' and q''. If they are not already present, add the transitions $(q', z_{i'j'}) \vdash^{\sigma} (q'', z_{i'j'})$ and $(q'', z_{i'j'}) \vdash^{\sigma^{-1}} (q', z_{i'j'})$ to M_{d+1}.

This completes the description of M_{d+1}. It is immediate from the construction that M_{d+1} satisfies the inductive assumptions with $d + 1$ replacing d. Observe from the form of the transitions that there is a fixed a priori upper bound (certainly $|Q||Z|^2|\Sigma^{\pm 1}| + |Q||Z||\Sigma^{\pm 1}|$ will suffice) on the number of transitions that can ever occur. Thus from some point no new transitions are ever added. Indeed, it is easy to prove that if no new transitions are added in going from M_d to M_{d+1}, then no new transitions will be added at any later stage and we may take $M = M_d$. □

It is easy to describe the canonical pda's for the examples which we have considered. For the infinite binary tree illustrated in Fig. 3 there is only one isomorphism class but two distinct second level subgraphs Γ_{00} and Γ_{01}. Since there is only one frontier point, M has a single state q_0. The initial total state of M is (q_0, z_{00}), and the action of M is simply to add z_{00} or z_{01} to the stack according as the input is 0 or 1.

The graph of Fig. 2 was defined as the graph of a pda, but the canonical pda behaves quite differently from the defining pda. There are two isomorphism types, that of the entire graph and that of $\Gamma\backslash\Gamma^{(n)}$, for $n \geq 1$. Since in either case there is only one actual second level subgraph, $Z = \{z_{00}, z_{10}\}$. Since the maximum number of frontier points is two, $Q = \{q_0, q_1\}$. The transitions of M are

$$\begin{array}{ll} (q_0, z_{00}) \vdash^a (q_0, z_{00}z_{10}), & (q_1, z_{10}) \vdash^a (q_1, z_1, z_{10}z_{10}), \\ (q_0, z_{00}) \vdash^b (q_1, z_{00}z_{10}), & (q_0, z_{10}) \vdash^\lambda (q_1, z_{10}), \\ (q_0, z_{10}) \vdash^a (q_0, z_{10}z_{10}), & (q_1, z_{10}) \vdash^{\lambda^{-1}} (q_1, z_{10}), \\ (q_0, z_{10}) \vdash^b (q_1, z_{10}z_{10}). & \end{array}$$

The remainder of this section is devoted to proving that the graph $\Gamma(M)$ of an arbitrary pda M is actually context-free and that one can effectively construct from M another pda M' which is the canonical pda for $\Gamma(M)$. The proofs are more difficult than the description of a canonical pda for a context-free graph. The reader interested in the monadic theory of context-free graphs may proceed directly to the next section. Before proving that the graph of a pda is context-free, we need a preliminary 'normal form' type lemma.

2.4. Lemma. *Let M be a pda and let (q, ζ) be a possible total state of M. Then there is a pda M' such that:*

(i) *No transition of M' changes the length of the stack by more than one.*

(ii) *M' never empties its stack.*

(iii) *There is a labeled graph isomorphism $\varphi: \Gamma(M) \to \Gamma(M')$ with the property that the total state (q', ζ') of M' which corresponds to (q, ζ) under φ has $|\zeta'| = 1$.*

Proof. Let M have stack alphabet Z and let k be the maximum number of symbols which any transition of M can add to the stack. Let $r = \max\{k, |\zeta|, 1\}$. The pda M' will have as its stack alphabet symbols which code words of Z^* whose length does not exceed r. To ensure that M' never empties its stack, we use two codings. Let Z'_1 be a set of symbols in one-to-one correspondence with all words of Z^* of length not exceeding r, and let Z' be a disjoint set of symbols in one-to-one correspondence with the nonempty words of Z^* of length not exceeding r. The pda M' will always have a symbol from Z'_1 on the bottom of its stack, and all other stack symbols will be from Z'. Now M' 'does what M does' but has on its stack the code for the segments of length r of the corresponding stack of M. $\square$

2.5. Lemma. *Let $M = \langle Q, \Sigma, Z, \delta, q_0, z_0, \hat{Q}\rangle$ be a pda. Then the graph $\Gamma = \Gamma(M)$, with any vertex chosen as origin, is context-free.*

Proof. Let (q_1, ζ_1) be the total state chosen as origin. Applying Lemma 2.4, we may assume without loss of generality that no single transition changes the stack length by more than one, that M never empties its stack, and that $|\zeta_1| = 1$. We begin with a crucial definition. Let (q, ζ) be a possible total state of M. Let $\Gamma'(q, \zeta)$ be the subgraph of $\Gamma(M)$ whose vertices are the possible total states that can be reached from (q, ζ) by a sequence of transitions and reverse transitions in which the stack length never becomes shorter than ζ, and whose edges are the edges of $\Gamma(M)$ connecting two such vertices. Define $\Delta'(q, \zeta)$ to be the subset of $\Gamma'(q, \zeta)$ consisting of those total states whose stack length is equal to $|\zeta|$. It is clear that if (q, ζ) and $(q, \hat{\zeta})$ are possible total states with the same component from Q and with ζ and $\hat{\zeta}$ having the same top letter, then $\Gamma'(q, \zeta)$ and $\Gamma'(q, \hat{\zeta})$ are isomorphic as labelled graphs and that $\Delta'(q, \zeta)$ and $\Delta'(q, \hat{\zeta})$ correspond under this isomorphism. The number of possible isomorphism classes is thus clearly bounded by $|Q||Z|$. The subgraphs $\Gamma'(q, \zeta)$ are defined in terms of the machine M. We shall see that the actual end-isomorphism classes of $\Gamma(M)$ are constructable from the $\Gamma'(q, \zeta)$ in a bounded number of ways.

If (q', ζ') is in $\Delta'(q, \zeta)$, then ζ' and ζ can differ only in their last letter (the top of the stack). Thus $|\Delta'(q, \zeta)| < |Q||Z|$. Any two states in $\Delta'(q, \zeta)$ are connected to each other by a path lying entirely in $\Gamma'(q, \zeta)$ and which is shortest among all such paths. We call the maximum length of such shortest paths the *diameter* of $\Delta'(q, \zeta)$. Since the number of isomorphism classes of the graphs $\Gamma'(q, \zeta)$ is finite, there is a maximum diameter c for all $\Delta'(q, \zeta)$. The constant c will play a large role in the consideration following.

Let (q, ζ) be any vertex of Γ. Let π be a shortest path from the chosen origin (q_1, ζ_1) of Γ to (q, ζ). We write $\lambda(q, \zeta)$ for the length of π. Recall that $\Gamma(q, \zeta)$ denotes the connected component of $\Gamma \backslash \Gamma^{(n)}$ containing (q, ζ) where $n = \lambda(q, \zeta)$. Thus $\Gamma(q, \zeta)$ is defined in terms of the graph Γ and not in terms of the way the pda M works. The task of the proof is to relate $\Gamma(q, \zeta)$ to $\Gamma'(q, \zeta)$. We shall now prove that we can find a vertex (q', ζ') on π such that:

(i) the distance along π from (q', ζ') to (q, ζ) does not exceed $2(c+1)$,

(ii) $\Gamma(q, \zeta) \subseteq \Gamma'(q', \zeta')$.

First of all, if $\lambda(q, \zeta) \leq 2(c+1)$ choose (q', ζ') to be (q_1, ζ_1). Since $|\zeta_1| = 1$, the minimum stack length, $\Gamma'(q_1, \zeta_1)$ is all of Γ and we are done. Now assume that $\lambda(q, \zeta) > 2(c+1)$. We now describe a *segmentation* of π as follows. The first segment begins at (q_1, ζ_1) and continues up to the last point on π having stack length one. Assuming inductively that the first segments have been defined, the $(i+1)$st segment consists of the first remaining point (whose stack length must be $i+1$) and continues until the last point of π with stack length $i+1$. It is possible that a segment consists only of a single vertex. By construction and the fact that no transition can increase the stack length by more than one, the minimum stack length of points in successive segments increases monotonically. Indeed, the minimum length is i in the ith segment. Thus, if (q_i, ζ_i) is any point of minimum stack length in the ith segment, then all later points of π lie in $\Gamma'(q_i, \zeta_i)$. From the definition of the constant c, each segment has length at most c.

We now choose (q', ζ') to be the last point of the last segment which has the property that its points all lie at distance at least $c+1$ from (q, ζ). This choice has property (i) and we want to show that (ii) holds. Let (q_2, ζ_2) be any point of $\Gamma(q, \zeta)$. For the sake of contradiction, assume that $(q_2, \zeta_2) \notin \Gamma'(q', \zeta')$. There is a path α from (q, ζ) to (q_2, ζ_2) which lies entirely in $\Gamma(q, \zeta)$. By definition, no point of $\Gamma(q, \zeta)$ can be closer to the origin than (q, ζ). Now one can go from (q', ζ') to (q_2, ζ_2) by first following π to (q, ζ) and then following α to (q_2, ζ_2). Since, by assumption, $(q_2, \zeta_2) \notin \Gamma'(q', \zeta')$ but all points of π between (q', ζ') and (q, ζ) do lie in $\Gamma'(q', \zeta')$, the path α must go through a point with stack length less than $|\zeta'|$. Thus α contains a point with stack length equal to $|\zeta'|$ since no transition or reverse transition alters stack length by more than one and the first such point, say (q_3, ζ_3) must lie in $\Delta'(q', \zeta')$. Hence the distance $d((q', \zeta'), (q_3, \zeta_3))$ does not exceed c. But since $d((q', \zeta'), (q, \zeta)) > c$, we have $\lambda(q_3, \zeta_3) < \lambda(q, \zeta)$ which contradicts $(q_3, \zeta_3) \in \Gamma(q, \zeta)$. This contradiction establishes that $\Gamma(q, \zeta) \subseteq \Gamma'(q', \zeta')$.

We now prove that $\Delta'(q, \zeta)$ lies within distances $3(c+1)$ of $\Delta'(q', \zeta')$. First consider the case where $\lambda(q, \zeta) < 2(c+1)$ and (q', ζ') was chosen to be (q_1, ζ_1). By definition, all points of $\Delta(q, \zeta)$ are at distance $\lambda(q, \zeta)$ from (q_1, ζ_1). Since all points of $\Delta'(q_1, \zeta_1)$ are within distance c of (q_1, ζ_1), all of $\Delta(q, \zeta)$ lies within distance $3c+1$ of $\Delta'(q', \zeta')$. Now consider the case where $\lambda(q, \zeta) \geqslant 2(c+1)$ and (q', ζ') is chosen as described earlier. Let (q'', ζ'') be in $\Delta(q, \zeta)$ and chose a shortest path β from (q_1, ζ_1) to (q'', ζ''). (Necessarily, β has length $\lambda(q, \zeta)$.) We proved above that no point of $\Gamma(q, \zeta)$ can have stack length as short as $|\zeta'|$. Thus $|\zeta''| > |\zeta'|$. Now one can go along π from (q', ζ') to (q, ζ), then along a path lying in $\Gamma(q, \zeta)$ to (q'', ζ''), and then along β^{-1} to the last point on β with stack length $|\zeta'|$. Call this point (q_4, ζ_4). Since no transition or reverse transition can alter the stack length by more than one, $(q_4, \zeta_4) \in \Delta(q', \zeta')$. One can thus go from (q_1, ζ_1) along β to (q_4, ζ_4), then a distance no greater than c to (q', ζ'), and then along π to (q, ζ). Since β and π are shortest paths (of the same length) from (q_1, ζ_1) to (q'', ζ'') and (q, ζ) respectively, we must have $d((q_4, \zeta_4), (q'', \zeta'')) \leqslant c + d((q', \zeta'), (q, \zeta)) < 3(c+1)$. We have thus established that all points of $\Delta(q, \zeta)$ lie within distance $3(c+1)$ of some point of $\Delta'(q', \zeta')$.

As we have seen, there are only finitely many isomorphism types of graphs $\Gamma'(q', \zeta')$. Also, $\Delta(q, \zeta)$ can be embedded in $\Gamma'(q', \zeta')$ in only finitely many ways since all its points must be within distance $3(c+1)$ of $\Delta'(q', \zeta')$. Consider the connected components into which $\Gamma'(q', \zeta')$ is partitioned when $\Delta(q, \zeta)$ is removed. Now a component must lie entirely in $\Gamma(q, \zeta)$ or be entirely outside $\Gamma(q, \zeta)$. For, a path joining vertices v_1 and v_2 which lie respectively in $\Gamma(q, \zeta)$ and outside of $\Gamma(q, \zeta)$ must, by definition, contain a point of $\Delta(q, \zeta)$. Therefore, $\Gamma(q, \zeta)$ is the union of $\Delta(q, \zeta)$ and some of the finitely many components of $\Gamma'(q', \zeta')$ which are obtained by removing $\Delta(q, \zeta)$. In summary, $\Gamma(q, \zeta)$ is completely determined by the isomorphism class of $\Gamma'(q', \zeta')$, some selection of points lying within fixed distance of $3(c+1)$ of $\Delta'(q', \zeta')$, and a choice of components after removing the selected points. The total number of choices for $\Gamma(q, \zeta)$ is thus bounded and $\Gamma(M)$ is context-free. □

Combining Lemmas 2.3 and 2.5 we have the following.

2.6. Theorem. *A finitely generated graph Γ is context-free if and only if Γ is the graph of some pushdown automaton.*

2.7. Corollary. *If Γ is a context-free graph, then Γ remains context-free with any vertex chosen as origin.*

Proof. Let Γ be $\Gamma(M)$ for a pda M and apply Lemma 2.5. □

We described the canonical pda M of a context-free graph Γ in Lemma 2.3. The operation of M is based on the second-level subgraphs of representatives of the end-isomorphism classes. We now observe that if one begins with any finite collection $G' = \{\Gamma_0, \ldots, \Gamma_n\}$ of subgraphs of Γ, all of the form $\Gamma(u)$ such that each end-isomorphism type does have a representative in G' (but Γ_i and Γ_j may be end-isomorphic for $i \neq j$) and a definite indexing of the second-level subgraphs of the Γ_i such that each Γ_{ij} is indeed end-isomorphic to Γ_i, then the rest of the construction yields a pda M' such that Γ is isomorphic to $\Gamma(M')$ and such that M' satisfies properties (1) and (2) of the construction of a canonical pda. We shall call such a pda a *pseudo-canonical pda* for Γ.

2.8. Theorem. *There is an algorithm which, when given a pda $M = \langle Q_1 \Sigma, Z, S, q_0, z_0, \hat{Q}\rangle$ and a pair (q_1, ζ_1), with $q_1 \in Q$ and $\zeta_1 \in Z^*$, decides if (q_1, ζ_1) is a possible total state of M, and, if so, constructs a canonical pda M' for the graph of M with (q_1, ζ_1) chosen as origin.*

Proof. Let M and (q_1, ζ_1) be given. We may assume that M has the properties of Lemma 2.4 and that $|\zeta_1| = 1$. (Precisely, of course, we switch from the given pda to the one constructed on Lemma 2.4.)

We begin by considering a sequence of graphs $\Omega_0 \subseteq \Omega_1 \subseteq \cdots$, each of whose vertex sets is the same as that of Γ, that is, all the possible total states of M. (Note that we are not yet claiming that the Ω_i are effectively constructable.) In each Ω_i there is a directed edge from (q, ζ) to (q', ζ') if $|\zeta'| = |\zeta|$ and there is a sequence of transitions of M from (q, ζ) to (q', ζ') in which the stack never becomes shorter than $|\zeta|$ nor longer than $|\zeta| + i$. (Note that we do *not* allow reverse transitions.)

As a consequence of the definition above, if there is an edge between (q, ζ) and (q', ζ') in any Ω_i, then (q', ζ') belongs to $\Delta'(q, \zeta)$. The number of total states in a fixed set $\Delta'(q, \zeta)$ cannot exceed $|Q||Z|$. Hence, there are no more than $(|Q||Z|)^2$ edges connecting points of the same $\Delta'(q, \zeta)$ on any Ω_i. Also the number of possible isomorphism classes of the $\Delta'(q, \zeta)$ is bounded by $|Q||Z|$ since only the state and the top symbol determine the isomorphism class. Thus, after a certain number of steps we must have $\Omega_{s+1} = \Omega_s$. Indeed, the smallest s for which this happens does not exceed $(|Q||Z|)^3$.

We next observe that if s is any nonnegative integer for which $\Omega_{s+1}=\Omega_s$, then $\Omega_{s+k}=\Omega_s$ for all $k\geqslant 1$. Assume the equality holds for all $j\leqslant k$ and consider Ω_{s+k+1}. Suppose that a new edge were introduced in Ω_{s+k+1}, say connecting two vertices in $\Delta'(q,\zeta)$. This edge corresponds to a sequence of transitions starting from (q,ζ), going through a point with stack length $|\zeta|+s+k+1$ and then to a point (q',ζ') with $|\zeta'|=|\zeta|$. Now no transition of M alters the stack length by more than one. Let (q^*,ζ^*) be the first point encountered in the sequence such that $|\zeta^*|=|\zeta|+1$ and let (q^{**},ζ^{**}) be the last point with stack length $|\zeta|+1$. Now, (q^*,ζ^*) and (q^{**},ζ^{**}) are connected in Ω_{s+k} and hence in Ω_s by the induction assumption. Hence (q,ζ) and (q',ζ') are connected in Ω_{s+1} and thus they were already connected in Ω_s.

The method of determining if (q_1,ζ_1) is a possible total state is now clear. Construct all points connected to the origin (q_0,z_0) in Ω_1. Next construct all points which are connected in Ω_2 to points already obtained. Iterate the procedure until no new points are obtained. Now see if (q_1,ζ_1) is connected to (q_0,ζ_0).

The calculation of the n-ball $\Gamma^{(n)}$ around (q_0,ζ) for any given n is now easy. Starting with the origin (q_0,ζ_0) list all total states which can be reached from (q_0,ζ_0) by a single transition or reverse transition and which are actually reachable by a sequence of transitions. (The latter condition is verifiable by the argument given above.) Next, list those possible total states which are reachable from the possible total states already obtained by a single transition or reverse transition. Iterate the procedure for n steps.

We now show, from the description of the pda M, how to calculate a bound c' on the constant c (which in the proof that $\Gamma(M)$ is context-free is defined to be the maximum diameter of any $\Delta'(q,\zeta)$). As usual, we may assume that M never empties its stack and that no transition of M alters the length of the stack by more than one. The proof involves a variant of the 'Ω-graph' construction above. We now consider a sequence of graphs $\Omega''_0\subseteq\Omega''_1\subseteq\Omega''_2\subseteq\cdots$ each of which has as vertex set $Q\times Z^*$. In other words, all pairs (q,ζ) occur as vertices regardless of whether or not (q,ζ) is a possible total state of M when started in the initial state (q_0,z_0). There is a directed edge from (q,ζ) to (q',ζ') in Ω''_i if $|\zeta|=|\zeta'|$ and there is a sequence of transitions and reverse transitions of M from (q,ζ) to (q',ζ') in which the stack never becomes shorter than $|\zeta|$ or longer than $|\zeta|+i$. As before, there is an integer t such that $\Omega''_{t+k}=\Omega''_t$ for all $k\geqslant 1$.

Start with all points of the form (q,z) with $q\in Q$, $z\in Z$. Construct all points connected to these points in Ω''_1. Next construct all points connected in Ω''_α to points already obtained. Iterate the procedure until no new points are obtained, and let Ω'' be the finite graph finally obtained. If (q,ζ) is actually a possible total state of M, then $\Delta'(q,\zeta)$ depends only on q and z where z is the right most symbol of ζ and $\Delta'(q,\zeta)$ is isomorphic to a connected subgraph of Ω'' which contains (q,z). Thus, if c' is the maximum length of any path in Ω'' which begins at a point of the form (q,z) and which does not traverse the same edge twice, we have $c'\geqslant c$.

We next show that we can solve the following problem. Given two disjoint finite subsets D_1 and D_2 of $\Gamma=\Gamma(M)$ we want to be able to decide which points of D_1

are in the same connected component of $\Gamma \setminus D_2$. Yet, another version of the Ω-graph construction solves this problem. Applying Lemma 2.4, we may assume that all points of D_2 have stack length one. We now consider a sequence of graphs $\Omega'_0 \subseteq \Omega'_1 \subseteq \Omega'_2 \subseteq \cdots$ where the vertex set of each Ω'_i is the set of possible total states of M except those in D_2, that is, the vertices of $\Gamma \setminus D_2$. In each Ω'_i there is an edge joining (q, ζ) and (q', ζ') if $|\zeta'| = |\zeta|$ and there is a sequence of transitions and reverse transitions of M from (q, ζ) to (q', ζ') which does not go through any total state in D_2 and in which the stack length never becomes shorter than $|\zeta|$ or longer than $|\zeta| + i$. As before, there is an integer r such that $\Omega'_{r+k} = \Omega'_r$ for all $k \geq 1$. Start with the points of D_1 and construct all points connected in Ω'_2 to points already obtained. Iterate the process until no new points are obtained and let Ω' denote the finite graph finally obtained. Now two points of D_1 are in the same component of $\Gamma \setminus D_2$ if and only if they are connected in Ω'.

We are now able to calculate a canonical pda M' for Γ. First calculate the k-ball $\Gamma^{(k)}$ where $k = k' + 3(c'+1)$ is such that no new state and top of stack pairs occur between $\Gamma^{(k')}$ and $\Gamma^{(k)}$. From the proof that Γ is context-free it follows that $\Gamma^{(k')}$ contains every state and top of stack pair occurring in Γ. For each point $(q, \zeta) \in \Gamma^{(k)}$ we may effectively calculate a point (q', ζ') (with (q', ζ') closer to the origin than (q, ζ) unless (q', ζ') is the origin) with $d((q', \zeta'), (q, \zeta)) \leq 2(c'+1)$ and $\Gamma(q, \zeta) \subseteq \Gamma'(q', \zeta')$. Let $m = \lambda(q', \zeta')$ and let $n = \lambda(q, \zeta)$. We may effectively calculate the finite graph $H(q, \zeta) = (\Gamma'(q', \zeta') \cap \Gamma^{(n+1)}) \setminus \Gamma^{(m)}$ together with a designation of the point (q', ζ') and the pair (q', z') with z' the top symbol of ζ'. The designation of which points are in $\Delta(q, \zeta)$, and the designation of the points of $\Gamma(q, \zeta) \cap \Gamma^{(n+1)}$ together with the information as to which are in the same component of $\Gamma \setminus \Gamma^{(n+1)}$. (The latter calculation is effective by the procedure for determining components which we discussed above.) For isomorphism types Γ_i we take the collection of distinct such graphs $H(q, \zeta)$. ('Distinct' means up to labeled graph isomorphism matching the distinguished sets of points and the corresponding information.) We may effectively calculate the 'isomorphism types' of the second level subgraphs Γ_{ij} of the Γ_i by calculating the corresponding $H(q, \zeta)$ graphs. Finally, we may use this information to calculate a canonical pda M' for Γ. $\square$

The next theorem refers to our paper [7] on groups with context-free word problem, and the proof assumes familiarity with that paper.

2.9. Theorem. *Let $G = \langle X; R \rangle$ be a finitely generated group. Then G is context-free if and only if the Cayley graph $\Gamma(G)$ is context-free.*

Proof. First assume that G is context-free and let $\Gamma = \Gamma(G)$. Let K be the triangulation constant for the given presentation, so that every closed path in Γ can be K-triangulated. We first claim that if u is any vertex of Γ, then the diameter of the set $\Delta(u)$ does need exceed $3K$. For, if u_1 and u_2 are two points of $\Delta(u)$, then their distances from the origin v_0 of Γ are both equal to n, say, and there is some path

β connecting u_1 and u_2 so that β lies entirely in $\Gamma \backslash \Gamma^{(n)}$. Let α and γ be paths of length n from v_0 to u_1 and u_2 respectively. There is a K-triangulation of the closed path $\alpha\beta\gamma^{-1}$, and some triangle t has vertices a, b, c on α, β and γ respectively. Now the distance $d(v_0, a)$ is at least $n-K$, for, otherwise, the total distance going along α from v_0 to a and then to b along the edge of the triangle t connecting a and b would be less than n; a contradiction. Similarly, $d(v_0, c) \geqslant n-K$. It immediately follows that $d(u_1, a) \leqslant K$ and $d(u_2, c) \leqslant K$. Thus the path from u_1 to u_2 composed of going along α^{-1} to a, then along the edge of the triangle connecting a and c, and then along γ to u_2 has length not exceeding $3K$.

If $\Delta(u)$ is removed from the graph Γ, the resulting graph is partitioned into a finite number of components determined solely by the set $\Delta(u)$. The set $\Gamma(u)$ consists of $\Delta(u)$ and a union of some of the components. Since Γ is the graph of the group G, there is a labeled graph isomorphism taking any vertex to any other vertex. Since $\Delta(u)$ has bounded diameter, there are only finitely many ways that $\Gamma(u)$ can be chosen as u ranges over Γ and thus Γ is context-free. Note that the only property used to establish the context-freeness of Γ is the uniform bound on the diameters of the sets $\Delta(u)$ and, thus, this property also characterizes context-free graphs.

Now assume that $\Gamma(G)$ is a context-free graph. Let M be the canonical pda for the graph Γ. Recall that M has the property that it never adds more than one symbol to the stack. By a standard trick, we may enlarge the stack alphabet Z to the stack alphabet $Z \times (Z \cup \{\Lambda\})$ and alter M so that the second component of each stack symbol records the first component of the previous symbol. We now convert our pda to a new pda M' having a reverse transition α^{-1} for each transition of M. The reverse transition corresponding to $(q, z) \vdash^{\alpha} (q', zz_1)$ is $(q', zz_1) \vdash^{\alpha^{-1}} (q, z)$ which actually depends only on the top symbol z_1 since z_1 'remembers' that the previous symbol is z. The new automaton is reversible by construction since for each transition there is now a reverse transition.

Since G is a group, there is unique edge with label $y \in X^{\pm 1}$ leaving each vertex of Γ. The canonical pda M is thus deterministic and therefore M' is also deterministic. Also, the possible total states of M' are in one-to-one correspondence with the group elements of G (i.e., the vertices of Γ). The word problem of G is accepted when the initial total state (q_0, z_{00}) is also used as the only terminal state. □

2.10. Corollary. *If G is a context-free group, then there is a deterministic, reversible pda which accepts the word problem of G and whose possible total states are in one-to-one correspondence with the elements of G.*

3. Tilings and monadic theories

In this section we establish a connection between the monadic theory MTh(Γ) of a finitely generated graph Γ and certain tilings on Γ. Using tilings we define the class of 'regular languages on Γ' and show that the decidability of $\mathrm{MTh}(\Gamma)$ is equivalent to deciding the emptiness of regular languages on Γ. The results of this section require no hypothesis on Γ except that Γ is finitely generated.

We first describe the monadic theory of a fixed finitely generated graph with origin v_0 and label alphabet Σ. The *monadic language* $\mathrm{ML}(\Gamma)$ has countably infinitely many (set) *variables* $X, Y, \ldots$ which denote sets of vertices of Γ. There is single constant v_0 which denotes the singleton set containing the origin of Γ. For each $\sigma \in \Sigma$ there is a unary function symbol which we also denote by σ. If X is a set, then $X\sigma$ denotes the set of vertices obtainable by starting at a vertex in X and traversing an edge with label σ. We call $X\sigma$ the set of *σ-successors* of X. If w is a word on Σ and X is a variable, then Xw and v_0w are *terms*. (By iteration, the set Xw of *w-successors* of X is the set of vertices obtainable as endpoints of paths which start at a vertex of X and which have label w. Also, v_0w is similarly interpreted in the obvious way.) If τ_1 and τ_2 are terms, the expression $\tau_1 \subseteq \tau_2$ is an *atomic formula* where the binary relation symbol $\subseteq$ denotes set inclusion. Starting with the atomic formulas one constructs the class of *formulas* by using propositional connectives and quantifiers in the usual way. The great expressive power of $\mathrm{ML}(\Gamma)$ lies in the fact that one can quantify over sets of vertices. Our formulation of $\mathrm{ML}(\Gamma)$ shows that one is really discussing the first order theory of sets of vertices of Γ.

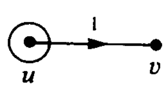
A *sentence* is a formula with no free variables. The monadic theory of Γ, $\mathrm{MTh}(\Gamma)$, is the set of all sentences in $\mathrm{ML}(\Gamma)$ which are true of Γ. We use the notation $F(X_1, \ldots, X_k)$ to denote a formula whose free variables are among $X_1, \ldots, X_k$. If $(S_1, \ldots, S_k)$ is a k-tuple of sets of vertices of Γ, we write $\Gamma F(S_1, \ldots, S_k)$ if the formula $F(X_1, \ldots, X_k)$ is true of Γ when each X_i is interpreted by S_i.

Many set-theoretic notions are easily definable in $\mathrm{ML}(\Gamma)$. Equality is immediately definable since $X = Y$ is equivalent to $[X \subseteq Y \wedge Y \subseteq X]$. There is not a constant for the empty set but $X = \emptyset$ is equivalent to $\forall Y[X \subseteq Y]$. A set X is a singleton set if X has a unique nonempty subset:

$$[X \neq \emptyset \wedge \forall Y[(Y \subseteq X \wedge Y \neq \emptyset) \rightarrow Y = X]].$$

We may thus talk about individual vertices. We write $u \in X$ as an abbreviation for $|U| = 1 \wedge U \subseteq X$.

We now consider tilings of graphs. Let Γ be a finitely generated graph with label alphabet Σ and origin v_0. A *general array* is a finite connected graph A labeled by Σ together with a distinguished subset of vertices called the *core* of A. The core of A, written $\mathrm{core}(A)$, may be empty. A *matching* $h: A \rightarrow \Gamma$ is a labeled graph monomorphism from A onto a subgraph A' of Γ such that if $u \in \mathrm{core}(A)$ and $(h(u), \sigma, v)$ is an edge of Γ, then $v \in h(A)$. In other words, we only allow A to be placed on Γ so that for a core vertex u all edges incident to $h(u)$ are images of edges of A. For example, the array



,

where the circled vertex u is in the core, does not have a matching in the infinite binary tree. (The reader will see later that the use of core vertices is necessary because we allow non-homogeneous graphs.)

We shall also need two other types of arrays. An *origin-specified* array is a general array A in which a vertex has been designated as the origin of the array. A matching $h: A \to \Gamma$ of an origin-specified array must, in addition to the requirement for core vertices, have $h(a_0) = v_0$. An *origin-excluded array* is a general array with the designation that it is origin-excluded. A matching $h: A \to \Gamma$ of an origin excluded array is a labeled graph monomorphism satisfying the core condition and such that the origin v_0 of Γ is *not* in the image $h(A)$. Finally, an *array* is an array of one of the three types above. The *diameter* of an array A is the maximum distance between two vertices of A.

Let C be a finite alphabet of colors. A *pattern* on an array A using the colors C is a mapping p from the vertex set of A into C. We simply write $p: A \to C$. A *coloring* of Γ is a map $J: \Gamma \to C$. We say that the pattern p *agrees* with the coloring J if there is a matching $h: A \to \Gamma$ such that the colors agree, that is, $p(a) = J(h(a))$ for all vertices a of A.

A *tiling constraint* is a pair (C, F) which consists of a finite color alphabet C and a finite collection F of patterns using C which are called *forbidden patterns.* A *tiling* of Γ in accordance with (C, F) is a mapping $T: \Gamma \to C$ such that no pattern in F agrees with T.

A collection of colorings using the same set C of colors is a *Γ-language* on C. If (C, F) is a tiling constraint, the set of all tilings T of Γ which satisfy the given tiling constraint is called a *basic tiling language.* Various operations can be performed on languages. First, there are the Boolean operations of intersection union and complementation applied to languages on the same alphabet C. If D is alphabet and $f: C \to D$ is a function, we can *project* the language L to obtain the language $f(L) = \{fJ: J \in L\}$ over D.

We assume that all the finite color alphabets under consideration are subsets of some fixed infinite recursive set. The class of all languages on the graph Γ which are obtainable by starting with the basic tiling languages and closing under the Boolean operations and projection is called the class of *Γ-regular languages.* When we say that we are given a Γ-regular language L we mean that we are given a complete description of how L is built up from basic tiling languages and the specifications of the basic tiling languages by tiling constraints. (The situation here is rather analogous to the specification of ordinary regular languages by regular expressions.) The *emptiness problem* for Γ-regular languages is solvable if there is an algorithm which, when given a Γ-regular language, decides if the language is empty. We next investigate the close connection between $\mathrm{MTh}(\Gamma)$ and Γ-regular languages.

3.1. Theorem. *Let Γ be a finitely generated graph. The monadic theory of Γ is decidable if and only if the emptiness problem for Γ-regular languages is solvable.*

Proof. First suppose that the emptiness problem for Γ-regular languages is solvable. Let $C_n = \{0, 1\}^n$, $n \geq 1$. There is an obvious one-to-one correspondence between n-tuples of sets of vertices of Γ and colorings of Γ using the alphabet C_n. If J is a coloring using C_n, then J defines an n-tuple $(S_1, \ldots, S_n)$ by viewing, for each vertex x of Γ, $J(x)$ as being $(\chi_1(x), \ldots, \chi_n(x))$ where χ_i is the characteristic function of S_i. Using this correspondence, we now view a collection of n-tuples of sets of vertices as being a language on C_n. Let C_0 be the singleton set $\{c_0\}$.

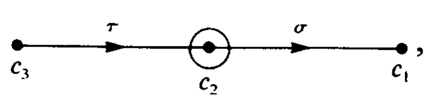
We prove that there is an effective construction which, when given a formula $F(X_1, \ldots, X_k)$ whose free variables are exactly $X_1, \ldots, X_k$, yields a Γ-regular language L on C_k such that, for every n-tuple $(S_1, \ldots, S_k)$ of sets of vertices of Γ, $\Gamma \models F(S_1, \ldots, S_k)$ if and only if the coloring J corresponding to $(S_1, \ldots, S_k)$ is in the language L. We say that L *codes* $F(X_1, \ldots, X_k)$. Any formula F can be effectively written in prenex form $QX_{i_s} \ldots QX_{i_1} G(X_1, \ldots, X_n)$ where G is quantifier-free. (For ease of notation, we assume the variables of G are $X_1, \ldots, X_n$.) We use induction on the complexity of formulas. First, consider an atomic formula $X_i u \subseteq X_j w$. Let $r = |u| + |w|$. If b is a vertex of Γ, the question of whether or not the set bu of u-successors of b is included in a set Sw is completely determined by the r-ball centered at b. Since there is a uniform upper bound m on the degrees of vertices of Γ and the finite label alphabet Σ is fixed, all graphs which could be the r-ball around a vertex can be effectively written down. Make each such graph a general array and let a be the 'center' of the array. (The vertex a has no formal status in defining matchings.) Put all vertices at distance less than r from a in the core of the array. For each such array forbid all patterns using C_n which put a in S_i but do not have au contained in $S_j w$. Let F be the set of all such forbidden patterns. Then the basic tiling language (C_n, F) is exactly the set of n-tuples $(S_1, \ldots, S_n)$ which satisfy $X_i u \subseteq X_j w$. For, $S_i u \not\subseteq S_j w$ if there is some vertex b of Γ with $b \in S_i$ but $bu \not\subseteq S_j w$. Picking the array with center a which has the same structure as the actual r-ball around b and the pattern which correctly represents membership in the sets S_l, there is a matching h with $h(a) = b$ such that the coloring agrees with a forbidden pattern. Thus, if J is a tiling in accordance with (C_n, F), the n-tuple which J represents satisfies $X_i u \subseteq X_j w$ and conversely.

We now consider the other possibilities for atomic formulas. One handles a formula of the type $v_0 u \subseteq X_j w$ as above except that the 'center' of an array is now specified as the origin so that all arrays used are origin-specified. We next deal with a formula of the form $X_i u \subseteq v_0 w$. For a tuple $(S_1, \ldots, S_n)$ to satisfy this formula all members of S_i which have u-successors must certainly be at a distance no more than $r = |u| + |w|$ from the origin of Γ. Write down all possible arrays which could be an r-ball (again with all vertices at distance less than r from the center in the core) and designate them as origin-excluded. Forbid all patterns which put the center a in S_i but which have au nonempty. Let $r' = 2|u| + |w|$ and write down all possibilities for the r'-ball with the center as the origin. Forbid all patterns which have a vertex in S_i whose u-successors are not in $v_0 w$. Letting F be the collection of forbidden patterns of both types, the basic tiling language (C_n, F) defines the collection of n-tuples satisfying $X_i u \subseteq v_0 w$. Finally, a formula of the form $v_0 u \subseteq v_0 w$ is easily handled.

A quantifier-free formula G is a Boolean combination of atomic formulas. To code G we need only perform the corresponding Boolean operations starting with the basic tiling languages coding the atomic formulas of G. We next use projection and complementation to handle quantifiers. Consider the formula $\exists X_i G(X_1, \ldots, \ldots, X_n)$ and let G be coded by the language L on C_n. If $f_i\colon C_n \to C_{n-1}$ is the function which deletes the ith coordinate, then the language $f_i(L)$ codes $\exists X_i G$. Write $\forall X_i G$ as $\neg\exists X_i \neg G$ and use projection and complementation to obtain the language coding $\forall X_i G$. For the case of quantifying the only free variable of G, say $\exists X_1 G(X_1)$, define $f\colon C_1 \to C_0$ by sending both 0 and 1 to c_0. Thus, in particular, given a sentence F of $\mathrm{ML}(\Gamma)$ we can effectively find the description of a Γ-regular language L on C_0 such that F is true of Γ if and only if L is nonempty. Hence, solvability of the emptiness problem for Γ-regular languages yields the decidability of $\mathrm{MTh}(\Gamma)$.

For the second half of the theorem suppose that $\mathrm{MTh}(\Gamma)$ is decidable. A tuple $(S_1, \ldots, S_n)$ of sets of vertices of Γ is a *disjoint cover* of Γ if every vertex of Γ belongs to exactly one of the S_i. (A disjoint cover differs from a partition only in that some of the S_i may be empty.) Being a disjoint cover is clearly definable in $\mathrm{ML}(\Gamma)$. Given a Γ-regular language L on an alphabet $C = \{c_1, \ldots, c_n\}$, we can effectively find a formula $H(X_1, \ldots, X_n)$ such that $\Gamma \models H(S_1, \ldots, S_n)$ if and only if $(S_i, \ldots, S_n)$ is a disjoint cover and coloring the elements of S_i by c_i yields a coloring $J \in L$. We say that *H codes L*. If this has been done, then L is nonempty if and only if $\Gamma \models \exists X_1 \ldots \exists X_n H(X_1, \ldots, X_n)$ so the decidability of $\mathrm{MTh}(\Gamma)$ implies that the emptiness problem for Γ-regular languages is solvable. □

We shall use lower case letters $x, y, \ldots$ to denote sets which have been defined to be singletons. Thus $QxG(x, \ldots)$ is an abbreviation for QX["X is a singleton" $\wedge G(X, \ldots)$]. First consider a basic tiling language defined by the constraint (C, F) where $C = \{c_1, \ldots, c_n\}$ and $F = \{P_1, \ldots, P_t\}$ say. Our formula $H(Y_i, \ldots, Y_n)$ will have the form "$(Y_1, \ldots, Y_n)$ is a disjoint cover $\wedge \bigwedge_{l=1}^{t} G_l$" where the formula G_l says that the scheme of coloring each Y_i by c_i does not produce the forbidden pattern P_l. It is easy to forbid a specific pattern by starting at some vertex and, working outwards, writing down the necessary core, origin, and color conditions. For example, suppose that $\Sigma = \{\sigma, \tau, \sigma^{-1}, \tau^{-1}\}$ and $C = \{c_1, c_2, c_3\}$ and P_1 is the pattern



where the circled vertex is in the core. Then, representing the core vertex by x_1, we can take G_1 as

$$\neg\exists x_1\{(x_1 \in Y_2 \wedge x_1\sigma^{-1} = \emptyset \wedge x_1\tau = \emptyset)$$
$$\wedge\, \exists x_2(x_2 = x_1\sigma \wedge x_2 \in Y_1) \wedge \exists x_3(x_3 = x_1\tau^{-1} \wedge x_3 \in Y_3)\}.$$

If $H_1(Y_1, \ldots, Y_n)$ and $H_2(Y_1, \ldots, Y_n)$ code languages L_1 and L_2 on the same alphabet C, then $H_1 \wedge H_2$ codes $L_1 \cap L_2$ while $\bar{L}_2$ is coded by "$(Y_1, \ldots, Y_n)$ is a disjoint cover $\wedge \neg H_2(Y_1, \ldots, Y_n)$".

Suppose that $f: C \to D = \{d_1, \ldots, d_t\}$ is a function. Resubscript the elements of C as $c_{1,1}, \ldots, c_{1,k_1}, \ldots, c_{t,1}, \ldots, c_{t,k_t}$ so that $f(c_{i,j}) = d_i$. (If some d_l is not in $f(C)$, there are no $c_{l,j}$.) Then, let $K(Z_1, \ldots, Z_t)$ be

$$\exists Y_1, \ldots, \exists Y_n \Bigg[H_1(Y_1, \ldots, Y_n) \wedge \forall x \Bigg[\bigwedge_{i=1}^{t} [x \in Z_i \leftrightarrow x \in Y_{i,1} \vee \cdots \vee x \in Y_{i,k_i}] \Bigg] \Bigg],$$

which codes $f(L_1)$.

It is often of considerable interest to study the monadic theory of graph Γ with certain distinguished subsets of vertices, say $K_1, \ldots, K_t$. We add new constants denoting these sets of $\mathrm{ML}(\Gamma)$ to obtain the monadic language $\mathrm{ML}(\Gamma, K_1, \ldots, K_t)$. The tiling approach is easily adapted to this situation. A *generalized array* for $K_1, \ldots, K_t$ is an array as before except that each vertex of an array A has an attached tuple of 0's and 1's intended to denote membership in the sets $K_1, \ldots, K_t$. A *matching* $h: A \to \Gamma$ satisfies the previous conditions and must also be such that for each $a \in H$, $h(c) \in K_i$ if and only if the tuple attached to a has 1 in its ith coordinate. Using patterns in generalized arrays for $K_1, \ldots, K_t$ we obtain generalized basic tiling languages, and closing under Boolean operations and projection we obtain the family of *$(\Gamma, K_1, \ldots, K_t)$-regular languages.* By extending the proof of the previous theorem to deal with atomic formulas involving the new constants one easily establishes the following theorem.

3.2. Theorem. *Let Γ be a finitely generated graph and let $K_1, \ldots, K_t$ be sets of vertices of Γ. Then* $\mathrm{MTh}(\Gamma, K_1, \ldots, K_t)$ *is decidable if and only if the emptiness problem for $(\Gamma, K_1, \ldots, K_t)$-regular languages is solvable.*

4. The monadic theory of context-free graphs

In this section we shall show that the monadic theory of any context-free graph Γ is decidable. The overall plan of the proof is the following. Since Γ is context-free, there is a canonical pda M for Γ. Using M we define a tree T_Γ (which is a definable subtree of a full k-ary tree). We shall show that the emptiness problem for Γ-regular languages is reducible to the emptiness problem for regular languages on T_Γ and the latter problem is solvable by Rabin's theorem.

Let Γ be a context-free graph and let M be a canonical pda for Γ. Let $Z = \{z_{ij}\}$ be the stack alphabet of Γ with $|Z| = k$ say. Let S_Γ be the full k-ary tree with edges labeled from Z. Thus each vertex v of S_Γ corresponds to the unique word ζ_v of Z^* obtained by starting at the origin v_0 and reading the edge labels on the minimal path form v_0 to v.

4.1. Definition. Let T_Γ be the subtree of S_Γ which consists of all those vertices corresponding to words ζ such that, for some q in the state set Q of M, (q, ζ) is a possible total state of M. Also, T_Γ has the edges of S_Γ which join two such vertices. (In short, T_Γ is the tree of possible stacks of M.)

4.2. Lemma. *The monadic theory* $\mathrm{MTh}(T_\Gamma)$ *is decidable.*

Proof. Recall that M never removes a symbol from the stack, that M has a transition of the form $(q, z_{ij}) \vdash^{\sigma} (q', z_{ij}z'_{ij})$ if and only if it is possible to reach a second-level subgraph Γ'_{ij} from the subgraph Γ_{ij} along an edge labeled σ, and that M has initial total state (q_0, z_{00}). As usual, we use the same letters to denote letters of Z and the corresponding unary function of $\mathrm{ML}(S_\Gamma)$. The vertex set of the tree T_Γ is the smallest subset Y of vertices of S_Γ such that $(q_0, z_{00}) \in Y$ and Y satisfies each sentence of the form $\forall x[x \in Y \wedge \exists y[[y \in Y \wedge x = yz_{ij}] \rightarrow xz'_{ij} \in Y]]$ whenever $(q, z_{ij}) \vdash^{\sigma} (q', z_{ij}z'_{ij})$ is a transition of M for some choice of q, q', and σ. We may thus relativize statements from S_Γ to T_Γ and decidability follows from Rabin's theorem. $\square$

4.3. Lemma. *There is an effective construction which yields, for every Γ-regular language L, a regular language L' on T_Γ whose tilings are in one-to-one correspondence with those of L.*

Proof. The origin of T_Γ corresponds to the origin of Γ. A point of T_Γ at the end of an edge labeled z_{ij} corresponds to the set of frontier points of a component of some $\Gamma\backslash\Gamma^{(n)}$ which has isomorphism type Γ_i. Let m be the maximum number of frontier points. As in the construction of the canonical pda for Γ, fix a numbering $\{u_0, \ldots, u_{k(i)}\}$ of the frontier points of each isomorphism class Γ_i. If C is a set of colors, let C' be the set of all tuples of the form $(c_0, \ldots, c_k)$ where $k < m$ and each $c_t \in C$. We next observe that there is a one-to-one correspondence between colorings of Γ using C and colorings of T_Γ using C' having the property that if u is a point of T_Γ at the end of an edge labeled by z_{ij}, then the tuple assigned to u has the same number of entries as the number of frontier points of Γ_i, and the tuple assigned to the origin has one entry. For, given an n and a component k of $\Gamma\backslash\Gamma^{(n)}$, the set Δ of the frontier points of K corresponds to a unique vertex u of T_Γ. Assign u to the tuple of colors assigned to the points of Δ in the fixed ordering. The converse is also clear.

Now suppose that L is a basic tiling language on Γ defined by the tiling constraint (C, F). Define a class F' of forbidden patterns using the color set C' as follows. First, require the origin to be colored by a tuple with one entry and forbid all patterns

$$\bullet \xrightarrow{\;z_{ij}\;} \bullet\, c'$$

where the color c' assigned to a vertex at the end of an edge labeled z_{ij} does not have the same number of entries as the number of frontier points of Γ_i.

By definition, any pattern $P \in F$ is connected. Therefore, in any matching $h: P \rightarrow \Gamma$, if u is a point of $h(P)$ at minimum distance from the origin v_0, then all of $h(P)$

must lie in the same component of $\Gamma \backslash \Gamma(u)$ as u does. If k is the diameter of P, then $h(P)$ lies within the subgraph of Γ corresponding to the set of points coded by a subtree of T_Γ of height at most k beginning at the point corresponding to the set $\Delta(u)$. Let k^* be the maximum diameter of patterns in F. Now place in F' all subtrees of height at most k^* (including designation of origin) which could occur in T_Γ and whose coloring transferred to Γ would contain a forbidden pattern of F. By the remarks above, there is a one-to-one correspondence between tilings in L and tilings in the language (C', F') on T_Γ.

Under the correspondence given so far, if L_i' corresponds to L_i, $i=1,2$, then $L_1' \cap L_2'$ corresponds to $L_1 \cap L_2$. If L' corresponds to L, then we may obtain a language corresponding to $\bar{L}$ by intersecting $\bar{L}'$ with the basic tiling language requiring that the tuples assigned to points of T_Γ have the correct number of entries. Finally, a projection $f: C \to D$ induces a projection $f'(c_0, \ldots, c_t) = (f(c_0), \ldots, f(c_t))$ from tuples of elements of C to tuples of elements of D. If L' corresponds to L, then $f'(L')$ corresponds to $f(L)$ and we are done. □

4.4. Theorem. *If Γ is a context-free graph, then* $\mathrm{MTh}(\Gamma)$ *is decidable.*

Proof. By Theorem 3.1 and Lemma 4.3 we may do the following. Given a sentence F of $\mathrm{ML}(\Gamma)$ we may effectively pass to a Γ-regular language L on Γ which codes F, then to a corresponding regular language L' on T_Γ, and then to a sentence F' of $\mathrm{ML}(T_\Gamma)$ which codes L'. Since $\mathrm{MTh}(T_\Gamma)$ is decidable by Lemma 4.2, we may decide if F' and thus F is true. □

It should be pointed out that there are known examples of graphs which are not context-free but which do have decidable monodic theory. Elgot and Rabin [2] have shown that the graph Γ illustrated in Fig. 4, where there are vertical segments at each distance from v_0 of the form $\frac{1}{2}n(n+1)$, has decidable monadic theory. ($\mathrm{MTh}(\Gamma)$ is equivalent to the monodic theory of the natural numbers with successor and with the subset $S = \{\frac{1}{2}n(n+1)\colon n \in N\}$ distinguished.) It is clear that Γ is not context-free since $\Gamma \backslash \Gamma^{(n)}$ and $\Gamma \backslash \Gamma^{(m)}$ are not end-isomorphic if $n \neq n$.

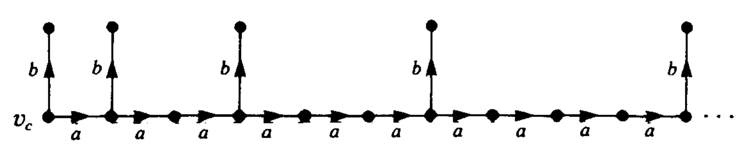


Fig. 4.

4.5. Conjecture. *If Γ has decidable monodic theory, then there are finitely many isomorphism types $J_1, \ldots, J_m$ of context-free graphs (with J_1 consisting of a single vertex) and a context-free graph $\hat{\Gamma}$ such that $\hat{\Gamma}$ is obtainable from Γ by following an effective prescription for attaching one of the graphs J_i to each vertex v of Γ (by identifying the origin of J_i with the vertex v).*

In other words, we conjecture that a graph with decidable monodic theory must be obtained from a context-free graph by some type of very restricted 'recursive pruning'.

We turn to some applications of Theorem 4.4. In Von Neumann's conception of a cellular automaton the underlying *universe* consists of the integer lattice points of n-dimensional Euclidean space. 'Identical' finite state automata lie at each point of the universe. Next, a neighborhood set B is specified. The next state of any particular automaton depends on the current state of the automaton and upon the current state of its neighbors. Each automaton thus changes states according to the same *local transition function* δ. A *state of the universe* is an assignment of a state from the state set Q to each automaton. Let U denote the set of all possible states of the universe. The local transition function L induces a global transition function $D: U \to U$. A question of interest in cellular automata theory is the decision problem "given δ, is D surjective (or injective)?". Whether or not these questions are algorithmically decidable in the two-dimensional case has not yet been settled, but it seems most likely that the questions are undecidable. Injectivity and surjectivity are known to be decidable in the one-dimensional case.

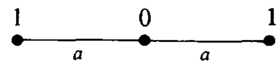
One can imagine a cellular automaton whose underlying universe is a fixed finitely generated graph Γ. All that is necessary is that neighbors are determined in a unique way. Thus we define a *neighborhood* to be a finite connected labeled graph N with a vertex r distinguished as the *root* and with the property that for each vertex v of Γ there is a unique labeled graph embedding $\varphi_v: N \to \Gamma$ with $\varphi_v(r) = v$. A *cellular automaton* on Γ is a triple $\langle Q, N, \delta\rangle$ where Q is a finite state set, N is a neighborhood and the local transition function δ assigns a state of Q to each pattern of states in N.

4.6. Theorem. *Let Γ be a context-free graph. There is an algorithm which decides, when given a triple $\langle Q, N, \delta\rangle$, whether N is actually a neighborhood, and if so, whether the global transition function is surjective (or injective).*

Proof. The proof is given by simply noting that given $\langle Q, N, \delta\rangle$ one may effectively write down a sentence of $\mathrm{ML}(\Gamma)$ which expresses the desired condition. If $Q = \{q_0, \ldots, q_n\}$, the fact that the global transition function is surjective is expressed by saying that for every disjoint cover $(Y_0, \ldots, Y_n)$ of Γ (where Y_i represents the points currently in state q_i) there is a disjoint cover $(X_0, \ldots, X_n)$ (the assignment of predecessor states) such that for every vertex v, $v \in Y_i$ if and only if the points in the neighborhood of v are in the correct X-sets for δ to assign state q_i to v. □

For example, suppose that Γ is the standard Cayley graph of the infinite cyclic group (the classical one-dimensional case), $Q = \{0, 1\}$, N is the usual neighborhood

and δ assigns the next state 1 to the pattern



and the next state 0 to all other patterns. That N is a neighborhood is expressed by

$$\forall v \exists ! u \; \exists ! w[u = va^{-1} \wedge w = va].$$

We abbreviate "(Y_0, Y_1) is a disjoint cover" by $\mathrm{dc}(Y_0, Y_1)$. The surjectivity of the global transition function is expressed by

$$\forall Y_0 Y_1[\mathrm{dc}(Y_0, Y_1)$$
$$\rightarrow \exists X_0, X_1[\mathrm{dc}(X_0, X_1) \wedge \forall u[u \in Y_1 \leftrightarrow ua^{-1} \in X_1 \wedge u \in X_0 \wedge ua \in X_1]]].$$

The study of cellular automata on general graphs seems interesting in its own right. In particular, the above theorem applies to cellular automata on the Cayley graph of a finitely generated free group of rank greater than one which seems to be an interesting case.

We have found that reachability problems of the type connected with vector addition systems are very natural problems on general graphs. In the standard formulation, an n-dimensional vector addition system consists of a finite set U of n-dimensional integer vectors. The first quadrant of Z^n consists of those vectors all of whose entries are nonnegative. If z is a point of the first quadrant, then the reachability set of z with respect to U, denoted by Uz, consists of all points v which can be obtained from Z by successively adding vectors in U while staying within the first quadrant. Precisely, $v \in Uz$ if there exists a sequence $z = v_1, \ldots, v_n = v$, where each v_i is in the first quadrant and for $i = 1, \ldots, n-1$ there exists a $u_i \in U$ such that $v_{i+1} = v_i + u_i$. The *membership problem* for the vector addition system U asks for an algorithm which, when given points z and v of the first quadrant, decides if $v \in Uz$. Whether or not the membership problem is always decidable still seems to be an open question.

The *inclusion problem* for reachability sets asks for an algorithm which, when given two n-dimensional vector addition systems U_1 and U_2 and two points z_1 and z_2 in the first quadrant, decides if $Uz_1 \subseteq U_2 z_2$. Rabin has shown that the inclusion problem is undecidable (see [3]).

Now the first quadrant is imply the set of points obtainable from 0 by adding the standard basis vectors. Thus, the first quadrant itself is a suitable reachability set.

We now formulate the idea of a vector addition system on an arbitrary finitely generated graph Γ, say with label alphabet Σ. Let t be any vertex of Γ and let W be a finite subset of Σ^*. The *quadrant* tW of t with respect to W consists of all vertices t_n such that there is a finite sequence $t = t_1, \ldots, t_n$ such that for each $i = 1, \ldots, n-1$ there exists a $w_i \in W$ such that $t_{i+1} \in t_i w_i$ (where, as usual, $t_i w_i$ is the set of vertices obtainable by starting at t_i and tracing out a path with label w_i). A *vector addition system* on Γ is a triple $\langle t, W, U \rangle$ where t is a vertex of Γ and W and

U are finite subsets of Σ^*. The *reachability set* Uz of a point z with respect to the system consists of all vertices v such that there is a sequence $z = v_1, \ldots, v_n = v$, where each v_i is in the quadrant tW and for each $i = 1, \ldots, n-1$ there is a $u_i \in U$ with $v_{i+1} \in v_i u_i$.

Viewed in this generality, reachability problems about vector addition systems include problems from several areas. For example, consider the following problem from group theory. Let $G = \langle X ; R \rangle$ be a finitely generated group and let H be a finitely generated subgroup of G. Let $W = X \cup X^{-1}$, so the quadrant $1W$ is all of G. Let U consist of a finite set of generators of H together with their inverses. An arbitrary element g of G is in the subgroup H if and only if g is in the reachability set $1U$. This problem of deciding if $g \in H$ is called the *generalized word problem* for H in G.

It is clear that the membership and inclusion problems for reachability sets can be formulated in ML(Γ). Thus we have the following theorem.

4.7. Theorem. *Let Γ be a context-free graph. Then the membership and inclusion problems for reachability sets of vector addition systems on Γ are uniformly solvable.*

References

[1] D.E. Cohen, *Groups of Cohomological Dimension One*, Lecture Notes in Mathematics **245** (Springer, Berlin/Heidelberg, 1972).

[2] C.C. Elgot and M.O. Rabin, Decidability and undecidability of extensions of the second-order theory of successor, *J. Symbolic Logic* **31** (1966) 169–181.

[3] M. Hack, The equality problem for vector addition systems is undecidable, *Theoret. Comput. Sci.* **2** (1978) 77–96.

[4] M.A. Harrison, *Introduction to Formal Language Theory* (Addison-Wesley, Reading, MA, 1978).

[5] J.E. Hopcroft and J.D. Ullman, *Introduction to Automation Theory, Languages and Computation* (Addison-Wesley, Reading, MA, 1979).

[6] D. E. Muller and P.E. Schupp, Pushdown automata, ends, second-order logic and reachability problems, in: *Proc. 13th Annual ACM Symp. on the Theory of Computing*, Milwaukee (1981) 46–54.

[7] D.E. Muller and P.E. Schupp, Groups, the theory of ends and context-free languages, *J. Comput. System Sci.* **26** (3) (1983) 295–310.

[8] M. Rabin, Decidability of second-order theories and automata on infinite trees, *Trans. Amer. Math. Soc.* **141** (1969) 1–35.

[9] J. Stallings, *Group Theory and Three-Dimensional Manifolds* (Yale University Press, New Haven, CT, 1971).