

The Bounded Weak Monadic Quantifier Alternation Hierarchy of Equational Graphs is Infinite

Olivier Ly

ly@labri.u-bordeaux.fr
LaBRI, Université Bordeaux I

Abstract. Here we deal with the question of definability of infinite graphs up to isomorphism by weak monadic second-order formulæ. In this respect, we prove that the quantifier alternation bounded hierarchy of equational graphs is infinite. Two proofs are given: the first one is based on the Ehrenfeucht-Fraïssé games; the second one uses the arithmetical hierarchy. Next, we give a new proof of the Thomas's result according to which the bounded hierarchy of the weak monadic second-order logic of the complete binary tree is infinite.

Introduction

Logic is by now a classical mean in theoretical computer science to describe complexity issues; it has been used in many areas, for instance effective computability (cf. [20]), descriptive complexity (cf. [7, 15]), or else formal language theory (cf. [16]).

This paper deals with the question of definability of infinite graphs up to isomorphism by logical formulæ. The graphs which are studied here are *equational graphs* which have been introduced in [4] as models of program schemes. Such a graph is the inductive limit of a sequence of finite graphs generated by a deterministic hyperedge replacement grammar (cf. [5, 21]). They extend strictly *context-free graphs* which have been introduced in [14] as configuration graphs of pushdown automata. Note that these kinds of graphs generalize the concept of *regular trees*; such a tree is defined as the tree of all the runs of a finite state automaton (see [3]).

We deal with *monadic second-order formulæ* on graphs (MS-formulæ for short) which are the logical formulæ which deal with graphs as relational logical structures using individual and set variables ranging over vertices and edges. We consider the *weak monadic second-order logic* (WMS logic for short) which consists in interpreting the MS-formulæ by considering set variables as ranging over *finite* sets of vertices and edges. WMS logic is a classical extension of the first-order logic. It is a variation of the well-known *monadic second-order logic* (MS-logic for short) (cf. [21, 8]). The monadic second-order logic is related to the concept of equational graphs because of the two following results: first, one can decide in an effective way whether a given equational graph satisfies a given

MS-formula according to MS-logic, which is also true for WMS-logic (cf. [4], for generalizations cf. [21, 2]). Second, the equational graphs are exactly the graphs of bounded tree-width (cf. [17]) which are definable up to isomorphism by MS-formulae (cf. [5]) according to MS-logic. Note that these results generalise the fundamental results of [1] and [18, 19] where MS and WMS were studied in the contexts of infinite words and infinite trees.

The present work is motivated by the conjecture of [5] according to which any equational graph is definable up to isomorphism by a formula of WMS-logic. This is true if one considers MS-logic (cf. [5]); let us note that this implies that the isomorphism problem for equational graphs is decidable. Concerning WMS-logic, some steps have been ever raised: the conjecture has been firstly proved to be true for context-free graphs (cf. [22]), and then for the equational graphs which have covering trees of finite degree (cf. [23]).

Here, we investigate equational WMS-definable graphs through the study of the *quantifier alternation bounded hierarchy*. A graph is said to be in the n -th level of the bounded WMS-hierarchy if there exists a WMS-formula which defines it up to isomorphism and which has $n - 1$ alternations of existential and universal unbounded quantifiers. One says that the hierarchy is infinite if for each integer n , the n -th level is *strictly* included in the $n + 1$ -th one. The main result of this paper is the following:

Theorem 1. *The bounded WMS-hierarchy of equational graphs is infinite.*

In order to see how this theorem fits in existing works, let us now mention some results about hierarchies relating to monadic second-order logical systems. Firstly, the bounded MS and WMS-hierarchies of languages of infinite words, i.e. the one successor theory, are finite, which follows from [1] and [12]. Next, the bounded MS-hierarchy of languages of infinite binary trees, i.e. the two successors theory (MS2S for short) is also finite, which follows from Rabin's Theorem (cf. [18], see also e.g. [9] for further results), while the weak one (the bounded WMS2S-hierarchy) is infinite (cf. [24], see also [13]). Concerning definability of graphs up to isomorphism, it follows from the results of [5] that the bounded MS-hierarchy of equational graphs is finite. Theorem 1 shows that the situation is different when considering WMS-logic, even though it follows from [22] that the bounded WMS-hierarchy of context-free graphs is finite and stops at most at the fifth level.

As we mentioned above the best result concerning the conjecture of WMS-definability of equational graphs has been raised in [23]. The WMS-formulae which are constructed in this work have unbounded numbers of quantifier alternations; Theorem 1 shows that one can not get away from this fact: equational graphs are hard to define up to isomorphism by using weak monadic second-order formulae.

We give two proofs of Theorem 1. The first one is based on an extension to WMS-logic of the classical technique of Ehrenfeucht-Fraïssé games [8, 10]). The method is similar to the well-known construction used to show that $FO(LFP)$ is strictly more expressive than $FO(TC)$ (see e.g. [6]). The second one is based on infinity of arithmetical hierarchy (see [20]). It allows us to recover the fact that WMS2S-hierarchy is infinite, which is deserved in [24] where it was proved

using arithmetical hierarchy as well together with Rabin's Theorem (cf. [18]). The first proof is in some sense stronger than the second one because the graphs which are constructed in it have covering trees of finite degree, which is not true for the second proof. But in other respects, this last one allows us to recover the result of [24] without using Rabin's Theorem, which does not seem to be possible with the game-based proof.

Acknowledgements. The author is greatly indebted to his supervisor G. Sénizergues for having spent much time in very helpful discussions. He also wants to thank the reviewers for their work which led to important improvements of this paper.

1 Preliminary

For an introduction to monadic second-order logical systems, the reader is referred to [6, 8], cf. also [21, chap. 5].

We deal with *labelled directed multi-graphs* (graphs for short), i.e. tuple $\langle V, E, \text{vert}, A, \text{lab} \rangle$ where V is an at most countable set whose elements are the vertices; E is the set of edges; $\text{vert} : E \rightarrow V \times V$ is the map defining the target and the origin of each edge. Vertices and edges are labelled by elements of A according to the map $\text{lab} : V \cup E \rightarrow A$. Such a graph is represented by a *logical relational structure* $\langle D, (D_a)_{a \in A}, \text{inc} \rangle$ where $D = V \cup E$; for all $a \in A$, $D_a \subset D$ is a unary relation on D defining elements labelled by a and inc is a ternary relation defining incidence in the sense that $(x, y, z) \in \text{inc}$ if and only if $x \in E$ is an edge of origin $y \in V$ and target $z \in V$. In order to simplify notations, the relational structure associated to a graph G is still denoted by G . We consider *monadic second-order formulæ* on graphs. These formulæ are constructed using individual variables (usually denoted by latin letters x, y, z, \dots) and set variables (usually denoted by greek letters α, β, \dots). Atomic formulæ are of the forms $x \in \alpha$ or $\alpha \subset \beta$ or $\text{inc}(x, y, z)$ or else $x \in D_a$ with $a \in A$. Syntax is not restricted: we allows existential and universal quantifiers over individual and set variables, conjunctions and negations. The *semantics* of such formulæ differs in the *monadic second-order logic* and in the *weak monadic second-order logic*: in the former one set variables range over all the subsets of the domain, while in the last one they range over finite subsets only.

Remark 1. What we call here MS-logic of graphs is usually denoted by MS_2 -logic in order to distinguish it from MS_1 -logic in which quantifications are done over vertex sets only (cf. e.g. [21]). MS_1 -logic deals with simple graphs; such a graph is encoded by a relational structure whose domain is its set of vertices only, instead of the union of its set of vertices and its set of edges as we consider here. MS_2 is more expressive than MS_1 . This is why we have chosen to prove Theorem 1 for MS_2 . As we shall see, it is true for MS_1 as well (see Remark 3).

A formula is said to be in *prenex form* if it is of the form $Q_1 X_1 \dots Q_n X_n \psi$ where for each i , $Q_i \in \{\exists, \forall\}$, the X_i 's are variables and ψ does not contain any quantifier. Any formula can be put in prenex form in an effective way (cf. e.g. [8]).

Let α and β be some set variables; let $\varphi(\alpha, \beta)$ be a formula with α and β as free variables. The quantifier over α in the formula $\exists\alpha \varphi(\alpha, \beta)$ ($\forall\alpha \varphi(\alpha, \beta)$ respectively) is said to be *bounded* if this last formula is equivalent to $\exists\alpha (\alpha \subset \beta \wedge \varphi(\alpha, \beta))$ ($\forall\alpha (\alpha \subset \beta \Rightarrow \varphi(\alpha, \beta))$ respectively). In this case it is denoted by $\exists\alpha \subset \beta \varphi(\alpha)$ ($\forall\alpha \subset \beta \varphi(\alpha)$ respectively). A formula in which all the quantifiers are bounded is said to be bounded. As stated by next Lemma, bounded quantifications can always be put after the unbounded ones (cf. [13] for a proof).

Lemma 1. *Let $\psi(\alpha, \beta, \gamma)$ be a formula, then we have:*

- $\exists\alpha \subset \beta \forall\gamma \psi(\alpha, \beta, \gamma) \equiv \forall\gamma' \exists\alpha \subset \beta \forall\gamma \subset \gamma' \psi(\alpha, \beta, \gamma)$
- $\exists\alpha \subset \beta \exists\gamma \psi(\alpha, \beta, \gamma) \equiv \exists\gamma \exists\alpha \subset \beta \psi(\alpha, \beta, \gamma)$
- $\forall\alpha \subset \beta \forall\gamma \psi(\alpha, \beta, \gamma) \equiv \forall\gamma' \forall\alpha \subset \beta \psi(\alpha, \beta, \gamma')$
- $\forall\alpha \subset \beta \exists\gamma \psi(\alpha, \beta, \gamma) \equiv \exists\gamma' \forall\alpha \subset \beta \exists\gamma \subset \gamma' \psi(\alpha, \beta, \gamma)$

A formula is called a Σ_n -formula (a Π_n -formula respectively) if it has the form $\exists X_1 \forall X_2 \dots \exists X_n \psi(X_1, X_2, \dots, X_n)$ ($\forall X_1 \exists X_2 \dots \exists X_n \psi(X_1, X_2, \dots, X_n)$ respectively) where ψ is bounded. The above lemma implies that any formula is equivalent to a Σ_n -formula (a Π_n -formula respectively) for a suitable n .

This classification of formulæ provides a classification of definable graph properties, i.e. the properties which can be expressed by logical formulæ. What we call here a graph property is formally a class of graphs, for instance the class of all the connected graphs. Let \mathcal{G}^{Σ_n} (\mathcal{G}^{Π_n} respectively) denote the set of families of graphs which are WMS-definable by some Σ_n -formulæ (by Π_n -formulæ respectively). Note that for all $n \geq 0$: $\mathcal{G}^{\Sigma_n} \cup \mathcal{G}^{\Pi_n} \subset \mathcal{G}^{\Sigma_{n+1}} \cap \mathcal{G}^{\Pi_{n+1}}$. This classification of WMS-definable graph families, i.e. the sequence \mathcal{G}^{Σ_n} , is called the *bounded weak monadic quantifier alternation hierarchy*.

2 Ehrenfeucht-Fraïssé Games

2.1 Ehrenfeucht-Fraïssé Games for Weak Monadic Second-Order Logic

Let $G = \langle V_G, E_G, \text{vert}_G, A, \text{lab}_G \rangle$ and $G' = \langle V_{G'}, E_{G'}, \text{vert}_{G'}, A, \text{lab}_{G'} \rangle$ be two A -labelled graphs. Let $\bar{\alpha}$ and $\bar{\alpha}'$ be some finite parts of G and G' respectively, i.e. finite subsets of vertices and edges. We call *partial isomorphism* between $\bar{\alpha}$ and $\bar{\alpha}'$ a one-to-one mapping $\sigma : \bar{\alpha} \rightarrow \bar{\alpha}'$ which preserves the adjacency, i.e. $\forall a, x, y \in \bar{\alpha} : \text{vert}_G(a) = (x, y)$ if and only if $\text{vert}_{G'}(\sigma(a)) = (\sigma(x), \sigma(y))$, and the labels, i.e. $\forall a \in \bar{\alpha} : \text{lab}(a) = \text{lab}(\sigma(a))$. We shall identify such a mapping with its graph, i.e. the subset of $(V_G \cup E_G) \times (V_{G'} \cup E_{G'})$ which encodes it. If σ and σ' are two partial isomorphisms, we say that σ *extends* σ' if $\sigma' \subset \sigma$ as subsets of $(V_G \cup E_G) \times (V_{G'} \cup E_{G'})$. The set of partial isomorphisms between two finite parts $\bar{\alpha}$ and $\bar{\alpha}'$ is denoted by $\text{PartIsom}(\bar{\alpha}, \bar{\alpha}')$.

Let **A** and **B** be two players, a session in the *Ehrenfeucht-Fraïssé game* associated to the pair of graphs (G, G') goes on as follows : at the first round, **A** chooses a finite part $\bar{\alpha}_1$ of G , and then **B** replies with a finite part $\bar{\alpha}'_1$ of G' together with a partial isomorphism $\sigma_1 \in \text{PartIsom}(\bar{\alpha}_1, \bar{\alpha}'_1)$. At the second round,

A chooses a finite part $\bar{\alpha}'_2$ of G' and B replies with a finite part $\bar{\alpha}_2$ of G together with a partial isomorphism $\sigma_2 \in \text{PartIsom}(\bar{\alpha}_1 \cup \bar{\alpha}_2, \bar{\alpha}'_1 \cup \bar{\alpha}'_2)$ which extends σ_1 ; and so on, A chooses finite parts alternatively in G and G' and B extends the isomorphism as A goes along. The game stops when B can not find a good answer. We say that B has a strategy of order n if he can play the first n rounds whatever choices A makes.

The following results show the classical link between this game view point and logic.

Lemma 2. *Let G and G' be two graphs and let $\psi(\alpha_1, \dots, \alpha_n)$ be a bounded formula. Let $\bar{\alpha}_1, \dots, \bar{\alpha}_n$ ($\bar{\alpha}'_1, \dots, \bar{\alpha}'_n$ respectively) be some finite parts of G (G' respectively) and let $\sigma \in \text{PartIsom}(\bigcup \bar{\alpha}_i, \bigcup \bar{\alpha}'_i)$ be a partial isomorphism which exchanges $\bar{\alpha}_i$ and $\bar{\alpha}'_i$ for all i , then we have $(G, \bar{\alpha}_1, \dots, \bar{\alpha}_n) \models \psi(\alpha_1, \dots, \alpha_n)$ if and only if $(G', \bar{\alpha}'_1, \dots, \bar{\alpha}'_n) \models \psi(\alpha_1, \dots, \alpha_n)$*

Lemma 3. *Let G and G' be two graphs such that B has a strategy of order n in the Ehrenfeucht-Fraïssé game associated to (G, G') ; then for all Σ_n -formula φ , $G \models \varphi$ implies that $G' \models \varphi$.*

2.2 First Proof of Theorem 1

This section is devoted to the proof of Theorem 1.

We begin by defining two sequences of graphs $(G_n)_{n \geq 1}$ and $(G'_n)_{n \geq 1}$ such that for all $n \geq 1$, G_n and G'_n are not isomorphic and B has a strategy of order n in the Ehrenfeucht-Fraïssé game associated to (G_n, G'_n) .

Let us set $V = (\mathbb{Z})^*$, i.e. the set of finite sequences of integers, and $E = (\mathbb{Z})^+ \times \{r, t\}$, where $(\mathbb{Z})^+$ denotes the set of finite non empty sequences of integers, r and t are two symbols (r like radial and t like transversal); let us consider the mapping $\text{vert} : E \rightarrow V \times V$ defined as following:

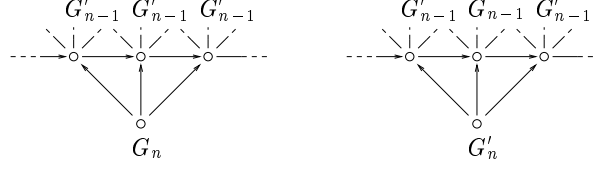
- $\forall e = ((x_1, \dots, x_l), r) \in E$, $\text{vert}(z) = ((x_1, \dots, x_{l-1}), (x_1, \dots, x_{l-1}, x_l))$,
- $\forall e = ((x_1, \dots, x_l), t) \in E$, $\text{vert}(z) = ((x_1, \dots, x_l), (x_1, \dots, x_l + 1))$.

Let A be a set; let $\mathcal{Z}(A)$ denote the set of A -labelled graphs whose vertex set, edge set and edge mapping respectively are V , E and vert . Let $G \in \mathcal{Z}(A)$ and $z_0 \in V$, we denote by $G(z_0)$ the graph of $\mathcal{Z}(A)$ defined by : $\forall z \in V \cup E$: $\text{lab}_{G(z_0)}(z) = \text{lab}_G(z_0.z)$, where $z_0.z$ denotes the concatenation of z_0 and z if $z \in V$, and by abuse of notation $(z_0.u, r)$ or $(z_0.u, t)$ if $z = (u, r) \in E$ or $z = (u, t) \in E$ respectively.

G_n and G'_n are now defined as elements of $\mathcal{Z}(\{0, 1\})$. First, all edges are labelled by 0: for all $n \geq 1$ and for all $e \in E$, let $\text{lab}_{G_n}(e) = \text{lab}_{G'_n}(e) = 0$. The labels of vertices are defined inductively as following:

- For all $z \in V$, let $\text{lab}_{G_1}(z) = 0$
- For all $z \in V$, $\text{lab}_{G'_1}(z) = \begin{cases} 1 & \text{if } z = (0), \\ 0 & \text{otherwise.} \end{cases}$
- For $n \geq 2$, G_n is defined from G'_{n-1} as following: $\text{lab}_{G_n}(()) = 0$ and $\forall z \in V$ such that $|z| = 1$: $G_n(z) = G'_{n-1}$.

- For $n \geq 2$, G'_n is defined from G_{n-1} and G'_{n-1} as following: $lab_{G_n}((0)) = 0$, $\forall z \in V \setminus \{(0)\}$ such that $|z| = 1 : G_n(z) = G'_{n-1}$ and $G_n((0)) = G_{n-1}$.



Lemma 4. For all $n \geq 1$, **B** has a strategy of order n in the Ehrenfeucht-Fraïssé game associated to (G_n, G'_n) .

Sketch of proof. (induction on n)

- *Case $n = 1$:* Let $\bar{\alpha}_1 \subset V_{G_1} \cup E_{G_1}$ be the choice of **A** in the first round of the game. All the elements of $\bar{\alpha}_1$ are labelled by 0 and the unique element of $V_{G'_1} \cup E_{G'_1}$ labelled by 1 is $(0) \in V_{G'_1}$; so **B** performs a shifting on the left of $m = \min\{x_1 \in \mathbb{Z} \mid \exists x_2, \dots, x_l \text{ such that } (x_1, \dots, x_l) \in \bar{\alpha}\}$ first level vertices; more precisely he uses the one-to-one mapping $\lambda_m : V \rightarrow V$ defined by $\lambda_m((x_1, \dots, x_l)) = (x_1 - m + 1, \dots, x_l)$ which extends in a natural way to $V \cup E$ to define an automorphism of $(V, E, vert)$ which shall be still denoted by λ_m ; **B** then chooses $\bar{\alpha}'_1 = \lambda(\bar{\alpha}_1)$ and $\sigma_1 = \lambda_m^{-1}|_{\bar{\alpha}_1}$.

- *Case $n > 1$:* Suppose now that **B** has a strategy ζ of order $n - 1$ relatively to (G_{n-1}, G'_{n-1}) ; we will then define a strategy of order n relatively to (G_n, G'_n) . The first round proceeds exactly like in the case $n = 1$; if $\bar{\alpha}_1 \subset V_{G_n} \cup E_{G_n}$ is the first choice of **A**, we define $m, \lambda_m, \bar{\alpha}'_1$ and σ_1 as above.

Let us consider $\bar{\alpha}'_2 \subset V_{G'_n} \cup E_{G'_n}$ the second choice of **A** which we divide into two parts $\bar{\alpha}'_2{}^1 = \{(x_1, \dots, x_l) \in \bar{\alpha}'_2 \mid x_1 \neq 0\}$ and $\bar{\alpha}'_2{}^2 = \{(x_1, \dots, x_l) \in \bar{\alpha}'_2 \mid x_1 = 0\}$. For $\bar{\alpha}'_2{}^1$ **B** uses the above shifting λ_m : let $\bar{\alpha}_2{}^1 = \lambda_m^{-1}(\bar{\alpha}'_2{}^1)$ and $\sigma_2{}^1 = \lambda_m^{-1}|_{\bar{\alpha}_2{}^1} \cup \sigma_1$. For $\bar{\alpha}'_2{}^2$, let us remark that $G'_n((0)) = G_{n-1}$ and $G_n((m-1)) = G'_{n-1}$, so $\bar{\alpha}'_2{}^2$ induces a finite part of G_{n-1} which can be considered as the first choice of **A** in the game associated to (G_{n-1}, G'_{n-1}) then, ζ gives a answer, i.e. a finite part $\tilde{\alpha}_2{}^2 \subset G'_{n-1}$ and a partial isomorphism which induce a finite part $\bar{\alpha}_2{}^2$ of the subgraph of G_n of root $(m-1)$ (the subset of G_n of words of which $(m-1)$ is a prefix) and a partial isomorphism $\sigma_2{}^2 \in \text{PartIsom}(\bar{\alpha}_2{}^2, \tilde{\alpha}_2{}^2)$.

Finally $\bar{\alpha}_2{}^1 \cup \bar{\alpha}_2{}^2$ and $\sigma_2{}^1 \cup \sigma_2{}^2$ is a correct answer of **B**.

For the succeeding rounds, **B** uses in the same way the strategy ζ in the game relative to the pair $(G'_n((0)), G_n((m-1)))$ and λ_m in the rest. Since ζ the strategy of **B** of order $n - 1$, intervenes in the second and later rounds, the above method gives a strategy of order n .

Since G_n and G'_n are not isomorphic, the preceding lemma together with Lemma 3 proves the next result:

Lemma 5. *For all $n \geq 1$, G_n is not definable up to isomorphism by a Σ_n -formula.*

We have now to see that G_n is indeed WMS-definable up to isomorphism, which is stated by next Lemma.

Lemma 6. *There exists a MS-formula Φ_n such that for any graph G , $G \models \Phi_n$ according to WMS-logic if and only if G is isomorphic to G_n ; moreover, there exists $N \geq 1$ such that for all $n \geq N$, Φ_n can be constructed to be Σ_{n+1} .*

Because of the lack of space, Proof is omitted (cf. [11]).

Hence, for all $n \geq N$, the isomorphism class of G_n belongs to $\mathcal{G}^{\Sigma_{n+1}} \setminus \mathcal{G}^{\Sigma_n}$ which proves that the bounded weak monadic quantifier alternation hierarchy is infinite. To conclude the proof of Theorem 1, it remains to show that the bounded hierarchy is still infinite when restricted to isomorphism classes of equational graphs. This is true seeing that the G_n 's actually are equational. Equational graphs can be seen as canonical solutions of systems of graph equations (cf. [5]). The lack of space makes impossible to construct in details such systems for the G_n 's. Nevertheless, we give the main ideas: first, we have to note that the graph which is made of one vertex of infinite degree which is connected to all the vertices of an infinite linear graph is equational; let us denote it by G . Then, the equations defining G_1 say that it is obtained by gluing one copy of itself on each vertex of G except the root, i.e. the vertex of G of infinite degree. On the other hand, G'_1 is obviously equational as it is obtained from G_1 by modifying the label of only one vertex. Then, the equations defining G_n and G'_n are constructed by induction from those defining G and those defining G_{n-1} and G'_{n-1} by following the definition scheme given above. For instance, G_n is obtained by gluing a copy of G'_{n-1} on each vertex of G except the root (cf. [11] for more details).

Let us note that instead of constructing directly such systems of graph equations, one can notice that the G_n 's are of bounded tree width (cf. [17]) and WMS-definable, as we saw in Lemma 6. In view of the results of [5], this implies that they are equational.

Remark 2. Transversal edges of G_n are useless in the proof of Theorem 1. However, they give the existence of a covering tree of finite degree, which, to some extent, is meaningful because of the following: the equational graphs with covering trees of finite degree have been proved to be WMS-definable up to isomorphism (cf. [23]); but the numbers of quantifier alternations of the formulæ which are constructed in this proof are not bounded. We have thus shown that one can not get away from this fact.

Remark 3. Let us note that MS_1 -logic (see Remark 1) gives rise to another bounded quantifier alternation hierarchy. In this respect, it turns out that our construction also shows that this hierarchy is also infinite. First, one can verify that a MS_1 -formula can be translated into an MS_2 -formula, adding at most one quantifier alternation. Therefore, in view of Lemma 5, G_n can not be defined up to isomorphism by a MS_1 -formula with less than $n - 1$ quantifier alternation. Second, one shows that G_n is MS_1 -definable.

Remark 4. The bounded weak monadic quantifier alternation hierarchy differs from the *weak monadic quantifier alternation hierarchy* (weak monadic hierarchy for short) which is defined in an analogous way by considering *first-order formulæ* instead of bounded formulæ. In this respect, one shows that there exists a fixed integer k such that for all n , the isomorphism class of G_n belongs to the k -th level of the weak monadic hierarchy. Indeed, the formula Φ_n given in Lemma 6 actually consists in the conjunction of a fixed weak monadic formula Φ which defines the family $\mathcal{Z}(\{0, 1\})$, and a first-order formula φ_n which checks the labels of G_n . So, the level of Φ_n in the weak monadic hierarchy is the one of Φ , which is fixed. We hence obtain that the k -th level of the weak monadic hierarchy contains instances beyond any given level of the bounded weak monadic hierarchy.

3 Arithmetical Hierarchy and Graph Hierarchy

3.1 Arithmetical hierarchy

For basics about effective computability, the reader is referred to [20].

Let $\tau^* : (\mathbb{N})^* \rightarrow \mathbb{N}$ be a Gödel numbering of finite integer sequences, i.e. a bi-recursive one-one mapping (cf. [20, p. 70] for such a construction); as usually, $\tau^*((x_1, \dots, x_k))$ shall be sometimes denoted by $\langle x_1, \dots, x_k \rangle$.

Let $(M_i^X)_{i \geq 0}$ be a Gödel numbering of the *set oracle Turing machines*. For any $A \subset \mathbb{N}$ and $i \geq 0$, let $f_i^A : W_i^A \subset \mathbb{N} \rightarrow \{0, 1\}$ be the partial function computed by M_i^X with A as oracle, where $W_i^A = \{x \in \mathbb{N} \mid M_i^X \text{ stops on the instance } x \text{ using } A \text{ as oracle}\}$ and $\forall x \in W_i^A, f_i^A(x) = 0$ if and only if M_i^X give 0 on the instance x using A as oracle. Classical *Turing machines* are identified with oracle machines with \emptyset as oracle. For all k , f_k^0 and W_k^0 shall be denoted by f_k and W_k respectively; f_k is called the k -th *recursive partial function*. Let Rec denote the set of all the recursive functions.

Let Σ_n^{arith} be the set of subsets of \mathbb{N} of the form $\{k \mid \exists k_1 \forall k_2 \dots \exists k_n f_{i_0}(\langle k_1, k_2, \dots, k_n, k \rangle) = 1\}$ for any fixed i_0 . We also consider Π_n^{arith} which denotes the set of subsets of \mathbb{N} of the form $\{k \mid \forall k_1 \exists k_2 \dots \forall k_n f_{i_0}(\langle k_1, k_2, \dots, k_n, k \rangle) = 1\}$ for any fixed i_0 . $\bigcup_n \Sigma_n^{arith}$ is called the *arithmetical hierarchy*.

Let us consider the *jump operation* which associates to any subset $A \subset \mathbb{N}$ the set $A' = \{x \in \mathbb{N} \mid x \in W_x^A\}$. For all $n \geq 1$, let $A^n = (A^{n-1})'$ where $A^0 = A$. We then consider the sequence $(\emptyset^n)_{n \geq 0}$, which is called the sequence of *jumps*.

Let A and B be two subsets of \mathbb{N} , let us recall that B is said to be *recursive* in A if and only if its characteristic function is equal to f_k^A for some k . B is said to be *recursively enumerable* in A if and only if there exists $k \in \mathbb{N}$ such that $B = W_k^A$.

The proofs of the two following results can be found in [20].

Lemma 7. *For all $n \geq 1$, $B \in \Sigma_n^{arith}$ if and only if B is recursively enumerable in \emptyset^n .*

The next result implies that the arithmetical hierarchy is infinite, i.e. for each n , $\Sigma_n^{arith} \subsetneq \Sigma_{n+1}^{arith}$.

Lemma 8. *For all $n \geq 0$, \emptyset^{n+1} is not recursive in \emptyset^n .*

3.2 Second Proof of Theorem 1

The first part of this second proof consists in constructing a sequence $(t_n)_{n \geq 1}$ of recursive trees such that \emptyset^n is reducible to the problem of being isomorphic to t_n . The second argument is that the problem of determining whether a recursive tree satisfies a Σ_n -formula according to weak monadic second-order logic is recursive in \emptyset^n (see Lemma 10 below). Seeing that t_n is constructed in order to be equational and WMS-definable up to isomorphism, Theorem 1 follows.

Let us make precise what we shall call a *recursive tree*. Let \mathcal{T}^∞ be the set of trees of the form $T = \langle V_T, E_T, \text{Edg}_T, \{0, 1\}, \text{lab}_T \rangle$ where

- $V_T \subset (\mathbb{N})^*$ is prefix closed, i.e. $x \in V_T$ and $y <_{\text{pref}} x$ implies $y \in V_T$;
- E_T is a copy of $V_T \setminus \{\emptyset\}$;
- For $x = (x_1, \dots, x_n) \in E_T$: $\text{Edg}_T(x) = ((x_1, \dots, x_{n-1}), (x_1, \dots, x_n))$;
- $\text{lab}_T : V_T \cup E_T \rightarrow \{0, 1\}$ and $\text{lab}_T|_{E_T} \equiv 0$.

For $T \in \mathcal{T}^\infty$ and $x_0 \in V_T$, let $T(x_0)$ be the *subtree* of T of root x_0 : $T(x_0) \in \mathcal{T}^\infty$, $V_{T(x_0)} = \{x \in (\mathbb{N})^* \mid x_0.x \in V_T\}$ and $\forall x \in V_{T(x_0)}$, $\text{lab}_{T(x_0)}(x) = \text{lab}_T(x_0.x)$. Let us consider for any partial function $f : \text{Dom}(f) \subset \mathbb{N} \rightarrow \{0, 1\}$ the element $\text{Tr}(f)$ of \mathcal{T}^∞ defined as following: $V_{\text{Tr}(f)}$ is the greater subset of $\text{Dom}(f \circ \tau^*) = \tau^{*-1}(\text{Dom}(f))$ which is prefix-closed; and for all $x \in V_{\text{Tr}(f)}$: $\text{lab}_{\text{Tr}(f)}(x) = f \circ \tau^*(x)$.

Definition 1 (Recursive Trees). A *recursive tree* is an element of \mathcal{T}^∞ of the form $\text{Tr}(f_i)$ for some i where f_i denotes the i -th recursive partial function. Let $T_i = \text{Tr}(f_i)$ denote the i -th recursive tree.

For each tree $t \in \mathcal{T}^\infty$ let us consider the set of integer $\text{RecIsom}(t) = \{i \in \mathbb{N} \mid T_i \text{ is isomorphic to } t\}$.

Let us turn to the construction of $(t_n)_{n \geq 1}$. We need for that an auxiliary sequence $(t'_n)_{n \geq 1}$ of trees. First, for all $n \in \mathbb{N}$, $V_{t_n} = V_{t'_n} = (\mathbb{N})^*$; then

- $\forall x \in V_{t_1}$, $\text{lab}_{t_1}(x) = \begin{cases} 1 & \text{if } x = (i) \text{ for any even integer } i \\ 0 & \text{otherwise} \end{cases}$
- $\text{lab}_{t'_1} \equiv 0$
- For $n \geq 1$, t_{n+1} is defined by: $\text{lab}_{t_{n+1}}(\emptyset) = 0$ and $\forall x \in V_{t_{n+1}}$ such that $|x| = 1$: $t_{n+1}(x) = t_n$ if $x = (i)$ with i even and $t_{n+1}(x) = t'_n$ otherwise.
- For $n \geq 1$, t'_{n+1} is defined by: $\text{lab}_{t'_{n+1}}(\emptyset) = 0$ and $\forall x \in V_{t'_{n+1}}$ such that $|x| = 1$: $t'_{n+1}(x) = t_n$.

Lemma 9. For all $n \geq 1$, \emptyset^n is recursive in $\text{RecIsom}(t_n)$.

Proof. Let $B(n, i) \subset \mathbb{N}$ be the i -th set of the n -th level of the arithmetical hierarchy, i.e. $\{k \mid \exists k_1 \forall k_2 \dots \forall k_n : f_i(\langle k_1, k_2, \dots, k_n, k \rangle) = 1\}$. Let us consider the following induction hypothesis :

HR_n: *there exists a computable recursive function $\tilde{\rho}_n$ such that $\forall i, k \in \mathbb{N}$: $\tilde{\rho}_n(\langle i, k \rangle) \in \text{RecIsom}(t_n) \cup \text{RecIsom}(t'_n)$ and $\tilde{\rho}_n(\langle i, k \rangle) \in \text{RecIsom}(t_n)$ iff $k \in B(n, i)$. In other words, for all n there is an algorithm which associates to any pair $i, k \in \mathbb{N}$ an algorithm which computes a tree which is isomorphic to t_n or t'_n and is isomorphic to t_n if and only if $k \in B(n, i)$.*

It follows from Lemma 7 that there exists an index i_n such that $\emptyset^n = B(n, i_n)$. Therefore, HR_n implies that \emptyset^n is Turing reducible to $\text{RecIsom}(t_n)$ by the function $\rho_n : k \mapsto \tilde{\rho}(\langle i_n, k \rangle)$. This implies the lemma.

• Proof of HR_1 : Here we describe the algorithm which computes $T_{\tilde{\rho}_1(\langle i, k \rangle)}$:

Instance : $x \in (\mathbb{N})^*$

if $|x| \neq 1$ then $\text{lab}_T(x) = 0$

else let $x_1, x_2 \in \mathbb{N}$ be such that $x = \langle x_1, x_2 \rangle$

if $x_1 = 0$ then $\text{lab}_T(x) = 0$

else $\text{lab}_T(x) = \begin{cases} 1 & \text{if } M_i \text{ stops and not give 0} \\ & \text{before the } x_2 \text{th calculus steps} \\ & \text{on } \langle x_1 - 1, k \rangle \\ 0 & \text{otherwise} \end{cases}$

$\tilde{\rho}_1(\langle i, k \rangle)$ is then defined to be the index of the tree which is described by this algorithm. Because of the lack of space, we shall omit the proof that $\tilde{\rho}_1$ indeed satisfies HR_1 .

• Let us suppose that HR_n is true.

We begin the proof of the induction step with some preliminaries: Let $i, k \in \mathbb{N}$; we have $B(n+1, i) = \{k \mid \exists k_1 \forall k_2 \dots \forall k_{n+1} : f_i(\langle k_1, k_2, \dots, k_{n+1}, k \rangle) = 1\}$.

Let us consider the integer set $\tilde{B}(n+1, i) = \{\langle k, k_1 \rangle \mid \forall k_2 \dots \forall k_{n+1} : f_i(\langle k_1, k_2, \dots, k_{n+1}, k \rangle) = 1\}$. Note that $B(n+1, i) = \{k \mid \exists k_1 : \langle k, k_1 \rangle \in \tilde{B}(n+1, i)\}$. This is the complement of an integer set $B(n, \delta(i, n))$ which belongs to the n -th level of the hierarchy; note that $\delta(i, n)$ is computable. Therefore, it follows from HR_n that $\langle k, k_1 \rangle \in \tilde{B}(n+1, i)$ if and only if $\tilde{\rho}_n(\langle \delta(i, n), \langle k, k_1 \rangle \rangle) \in \text{RecIsom}(t'_n)$.

Let us now describe the algorithm computing $T_{\tilde{\rho}_{n+1}(\langle i, k \rangle)}$:

Instance : $x = (x_1, \dots, x_l) \in (\mathbb{N})^*$

if $x = ()$ then $\text{lab}_{T_{\tilde{\rho}_{n+1}(\langle i, k \rangle)}}(x) = 0$

Let $x_{11}, x_{12} \in \mathbb{N}$ be such that $x_1 = \langle x_{11}, x_{12} \rangle$

if $x_{11} = 0$ then $\text{lab}_{T_{\tilde{\rho}_{n+1}(\langle i, k \rangle)}}(x) = \text{lab}_{t_n}((x_2, \dots, x_l))$ (i)

else $\text{lab}_{T_{\tilde{\rho}_{n+1}(\langle i, k \rangle)}}(x) = \text{lab}_{T_{\tilde{\rho}_n(\langle \delta(i, n), \langle k, x_{11}-1 \rangle \rangle)}}((x_2, \dots, x_l))$ (ii)

First, (i) guarantees that there are infinitely many sub-trees of level 1 $T_{\tilde{\rho}_{n+1}(\langle i, k \rangle)}$ which are isomorphic to t_n .

Second, (ii) guarantees by induction that each sub-tree of level 1 is isomorphic to t_n or to t'_n . One verifies that if there is at least one sub-tree of level 1 which is isomorphic to t'_n then there are infinitely many one.

Now $T_{\tilde{\rho}_{n+1}(\langle i, k \rangle)}$ is isomorphic to t_{n+1} if and only if at least one of its sub-trees of level 1 is isomorphic to t'_n . This is true if and only if there exists $x_{11} \neq 0$ such that $\tilde{\rho}_n(\langle \delta(i, n), \langle k, x_{11}-1 \rangle \rangle) \in \text{Isom}(t'_n)$, which is equivalent, by using preliminaries, to $k \in B(n+1, i)$.

Because of the lack of space, we state the next lemma without proof.

Lemma 10. *The problem of determining whether a recursive tree satisfies a Σ_n -formula according to weak monadic second-order logic is recursive in \emptyset^n .*

It follows from the previous lemma that whatever $n \geq 1$ is, there is no Σ_n -formula which defines t_{n+1} up to isomorphism. Indeed, the existence of such a formula would be a contradiction with Lemma 8. On the other hand, like in section 2.2, t_n is equational and it is WMS-definable. We shall omit to prove that it is WMS-definable. The construction of some systems of graph equations which define the t_n 's can be performed by using the ideas of section 2.2. First, transversal edges are no longer considered here. And even if the definition schemes of the t_n 's and the t'_n 's are slightly different from the ones of the G_n 's and the G'_n 's, they mainly follow the same idea. And one can verify that the idea for the construction of the systems which define the G_n 's applies here as well to construct some systems defining the t_n 's. Theorem 1 then follows.

3.3 Thomas Theorem

Here we deal with *labelled complete binary trees*, i.e. mapping $t : \{l, r\}^* \rightarrow \{0, 1\}$; their set is denoted by $T_{inf}(\{0, 1\})$. In the context of the weak monadic second-order logic of the binary tree (WMS2S for short), the usual concept of bounded quantifier is different: following [24] and [13], bounded quantifiers are indeed those of the form $\exists \alpha \leq_{pref} \beta \dots$ or $\forall \alpha \leq_{pref} \beta \dots$ where \leq_{pref} denote the prefix ordering of $\{l, r\}^*$. This defines an other concept of bounded hierarchy. However, one verifies that levels are the same.

In [24], it is proved that the WMS2S bounded hierarchy is infinite, i.e. Theorem 2 below; the proof involves infiniteness of arithmetical hierarchy together with Rabin's theorem (cf. [18]). By using the tools introduced in the preceding section, we give an alternative proof of this result which does not use Rabin's theorem.

Theorem 2 (Thomas 82). *The bounded WMS2S hierarchy is infinite.*

Proof. Let $\Lambda : \{l, r\}^* \rightarrow \mathbb{N}^*$ be the mapping defined as follows: for any x_1, \dots, x_{l+1} , $\Lambda(l^{x_1} r^{x_2} \dots l^{x_l} r^{x_{l+1}}) = (x_1, \dots, x_l)$. Λ allows us to consider a mapping from $T_{inf}(\{0, 1\})$ to \mathcal{T}^∞ , which shall be still denoted by Λ , defined as following: for $t \in T_{inf}(\{0, 1\})$, $V_{\Lambda(t)} = (\mathbb{N})^*$ and for any x_1, \dots, x_l , $lab_{\Lambda(t)}(x_1, \dots, x_l) = t(l^{x_1} r^{x_2} \dots l^{x_l} r)$. Λ contracts all the left edges of t . We also consider the partial converse Λ^{-1} defined on the set of trees of \mathcal{T}^∞ whose domains are equal to $(\mathbb{N})^*$.

Let us set $\mathcal{T}_n = \{t \in T_{inf}(\{0, 1\}) \mid \Lambda(t) \text{ is isomorphic to } t_n\}$. We will see that the tree family \mathcal{T}_n is definable, but not at a lower level than the n -th of the hierarchy. Let ρ_n be the Turing reduction of \emptyset^n to $\text{RecIsom}(t_n)$ which has been constructed in the proof of Lemma 9; let us note that for each integer k , $V_{T_{\rho_n(k)}} = (\mathbb{N})^*$ and thus $\Lambda^{-1}(T_{\rho_n(k)})$ is defined. Now, for each integer k , we have:

$$k \in \emptyset^n \text{ if and only if } \Lambda^{-1}(T_{\rho_n(k)}) \in \mathcal{T}_n.$$

On the other hand, one can verify that the family \mathcal{T}_n can be WMS2S-defined by a Σ_ℓ -formula φ_n for a suitable ℓ . Then $k \in \emptyset^n$ if and only if $\Lambda^{-1}(T_{\rho_n(k)}) \models \varphi_n$. By a result similar to Lemma 10, one verifies that this last predicate is recursive in \emptyset^ℓ , which implies that $\ell \geq n$. Theorem 2 is proved.

References

1. J.R. Büchi, On a decision method in restricted second order arithmetic. Proc. Internat. Congr. on Logic, Methodology and Philosophy of Science, E. Nagel and al. eds. p1-11 (1960).
2. D. Caucal, On Infinite Transition Graphs having Decidable Monadic Theory, ICALP'96 - LNCS 1099:194-205 (1996).
3. B. Courcelle, Fundamental Properties of Infinite Trees, TCS 25:95-169 (1983).
4. B. Courcelle, The monadic second-order logic of graphs II : Infinite Graphs of Bounded Width, Math. System Theory 21:187-221 (1989).
5. B. Courcelle, The monadic second-order logic of graphs IV : Definability properties of Equational Graphs, Annals of Pure and Applied Logic 49:193-255 (1990).
6. Ebbinghaus H.-D and Flum J., Finite Model Theory, second edition, Springer Verlag (1999).
7. R. Fagin, Generalized first-order spectra and polynomial-time recognizable sets. In "Complexity of Computation", SIAM-AMS Proceedings 43-73 (1974).
8. Y. Gurevich, Monadic Second-Order Theories, in Model Theoretic Logic, Barwise and Ferferman eds. Springer (1985).
9. D. Janin and G. Lenzi, On the structure of the monadic logic of the binary tree. MFCS'99, LNCS 1672:310-320 (1999).
10. R. Lassaigne and M. de Rougemond, Logique et Complexité, Hermes - collection informatique (1996).
11. O. Ly, On Hierarchy of Graphs, internal report no 1178-97 LaBRI - 1997.
12. R. McNaughton, Testing and Generating Infinite Sequences by a Finite Automaton, Inf. Contr. 9:521-530 (1966)
13. A.W. Mostowski Hierarchies of Weak Monadic Formulas for Two Successors Arithmetic, J. Inf. Process. Cybern. EIK 23 10/11, 509-515 (1987).
14. D.E. Muller and P.E. Schupp, The Theory of Ends, Pushdown Automata, and Second-Order Logic, TCS 37: 51-75 (1985).
15. C.H. Papadimitriou, Computational Complexity, Addison-Wexley Pub. Comp. (1994).
16. D. Perrin and J. E. Pin, First-Order Logic and Star-Free Sets, J. Comp. Syst. Sci. 32:393-406 (1986).
17. N. Robertson and P. Seymour, Some new results on Well-Quasi Ordering of Graphs. Annals of Discrete Math., 23:343-354 (1984).
18. M.O. Rabin, Decidability of second order theories and automata on infinite trees, Trans. Amer. Math. Soc. 141 (1969).
19. M.O. Rabin, Weakly Definable Relations and Special Automata
20. H. Rogers, Theory of Recursive Functions and Effective Computability, McGraw-Hill, Series in Higher Mathematics (1967).
21. G. Rozenberg ed., Handbook of Graph Grammars and Computing by Graph Transformation, vol. 1, World Scientific (1997).
22. G. Sénizergues, Définissabilité des graphes context-free, unpublished work.
23. G. Sénizergues, Definability in weak monadic second order logic of some infinite Graphs, Dagstuhl seminar on Automata theory : Infinite Computations 28:16-16 (1992).
24. W. Thomas, A Hierarchie of Sets of Infinite Trees, LNCS 145:335-342 (1982).