Shelah-Stupp's and Muchnik's Iterations Revisited*

Paweł Parys[0000−0001−7247−1408]

Institute of Informatics, University of Warsaw, Poland parys@mimuw.edu.pl

Abstract. Iteration is a model-theoretic construction that replicates a given structure in an infinite, tree-like way. There are two variants of iteration: basic iteration (a.k.a. Shelah-Stupp's iteration), and Muchnik's iteration. The latter has an additional unary predicate (not present in basic iteration), which makes the structure richer. These two variants lead to two hierarchies of relational structures, generated from finite structures using MSO-interpretations and either basic iteration or Muchnik's iteration. Caucal and Knapik (2018) have shown that the two hierarchies coincide at level 1, and that every level of the latter hierarchy is closed under basic iteration (which in particular implies that the former hierarchy collapses at level 1). We prove the same results using a different, significantly simpler method.

Keywords: Shelah-Stupp's iteration, Muchnik's iteration, hierarchy, MSO decidability, infinite relational structures

1 Introduction

The story about iterations starts with the monadic second-order (MSO) logic. While defining sets of words or trees, this logic is equiexpressive with finite-state automata, thus defines regular languages. The MSO logic is of course decidable over finite structures. Moreover, it have been shown decidable over natural numbers with successor [2, 12, 24], and over the infinite complete binary tree [18].

After these fundamental results, a long series of other examples of infinite structures with decidable MSO theory has emerged. They include natural numbers with successor and an additional unary predicate [13, 21, 23], transition graphs of pushdown automata [16] or higher-order pushdown automata [3, 4], HR-equational hypergraphs [7] and VR-equational hyperhraphs [9], prefix-recognizable graphs [5], and trees generated by higher-order recursion schemes [15, 17].

Besides the particular classes of structures with decidable MSO theory, some operations that preserve MSO-decidability, creating a more complex structure from a simpler one, were proposed; such operations are called MSO-*compatible*. Among those, we have generalised unions of Shelah [20], MSO-interpretations

* Work supported by the National Science Centre, Poland (grant no. 2016/22/E/ST6/00041).

(or, more generally, MSO-transductions) [1, 8, 14], unfolding of directed graphs into trees [11], and iteration. We concentrate here on the last operation on this list, namely on iteration.

While iterating a structure, we create infinitely many copies of it, and we organize them in a shape of a tree. Children of every node of the tree are indexed by elements of the structure itself. Thus, elements of the iterated structure can be seen as nonempty (finite) words whose letters are elements of the original structure: the last letter is an element in one of the copies, and the prefix without the last letter is an index of this copy. Relations are preserved within each copy, and a new binary "son" relation is added, connecting an element in one copy with all elements belonging to a child of this copy indexed by the former element (i.e., every word w with words of the form wa). This construction is first mentioned by Shelah [20], who refers to an unpublished paper of Stupp [22], which contains the proof of the fact that this operation is indeed MSO-compatible; thus the above operation is called *Shelah-Stupp's iteration*, or *basic iteration*. The resulting structure may be extended by a unary "clone" predicate, which holds in the unique element of every copy that is an index of this copy among its siblings (i.e., in words of the form waa); this way we obtain Muchnik's iteration, which is also MSO-compatible. This result is attributed to Muchnik, but was presented by Semenov [19] and Walukiewicz [25].

The MSO-compatible operations allow to create hierarchies of classes of structures with decidable MSO theories, containing most of the examples mentioned so far. In the most known Caucal's hierarchy of directed graphs [4], one starts from finite graphs, and repeatedly applies unfolding and MSO-interpretation. In an equivalent definition [3], unfolding is replaced by Muchnik's iteration. We consider here a generalization of the latter hierarchy from directed graphs to arbitrary relational structures. Thus, starting from finite structures, we construct structures on the next level of the hierarchy by applying Muchnik's iteration to structures on the previous level, followed by arbitrary MSO-interpretations. Another hierarchy can be constructed using basic iteration instead of Muchnik's iteration.

The latter two hierarchies were considered by Caucal and Knapik [6], who prove that

- the two hierarchies coincide at level 1, and
- every level of the hierarchy involving Muchnik's iteration is closed under basic iteration (which in particular implies that the hierarchy involving basic iteration collapses at level one).

We prove the same results using a different, significantly simpler method.

The proof of Caucal and Knapik is quite indirect: it utilizes prefix-recognizable structures, as well as higher-order pushdown automata. Moreover, their constructions are rather involved. We, instead, work directly with the definition of iterations. Recalling that elements of the iterated structure can be seen as words, our approach is based on a very simple idea saying that a word of words can be encoded in a word (we only need some separator to mark the glue points). Based on this idea, we prove that (modulo existence of the aforementioned separator) a composition of a Muchnik's iteration (applied first) with a basic iteration (applied later) can be encoded in a single Muchnik's iteration. From this statement, the aforementioned second main result of Caucal and Knapik (i.e., closure under basic iteration) easily follows.

2 Preliminaries

The MSO logic and MSO-interpretations. A *signature* Σ (of a relational structure) is a finite set of relation names, $R_1, \dots, R_r$, together with a natural number called an *arity* assigned to each of the names. A *(relational) structure* $\mathcal{S} = (U^{\mathcal{S}}, R_1^{\mathcal{S}}, \dots, R_r^{\mathcal{S}})$ over such a signature Σ is a set $U^{\mathcal{S}}$, called the *universe*, together with *relations* $R_i^{\mathcal{S}}$ over $U^{\mathcal{S}}$, for all relation names in the signature; the arity of the relations is as specified in the signature.

We assume two countable sets of variables: $\mathcal{V}^{FO}$ of first-order variables (denoted using lowercase letters $x, y, \dots$) and $\mathcal{V}^{MSO}$ of set variables (denoted using capital letters $X, Y, \dots$). Atomic formulae are

- $R(x_1, \dots, x_n)$, where R is a relation name of arity n (coming from a fixed signature Σ), and $x_1, \dots, x_n$ are first-order variables, and
- $x \in X$, where x is a first-order variable, and X a set variable.

Formulae of the *monadic second-order logic*, MSO, are built out of atomic formulae using the Boolean connectives $\vee, \wedge, \neg$, first-order quantifiers $\exists x$ and $\forall x$ for $x \in \mathcal{V}^{FO}$, and set quantifiers $\exists X$ and $\forall X$ for $X \in \mathcal{V}^{MSO}$. We use the standard notion of *free variables*.

In order to evaluate an MSO formula φ over a signature Σ in a relational structure $\mathcal{S}$ over the same signature, we also need a *valuation* ν, which is a partial function that maps

- variables $x \in \mathcal{V}^{FO}$ to elements of the universe of $\mathcal{S}$, and
- variables $X \in \mathcal{V}^{MSO}$ to subsets of the universe of $\mathcal{S}$.

The valuation should be defined at least for all free variables of φ. We write $\mathcal{S}, \nu \models \varphi$ when φ is *satisfied* in $\mathcal{S}$ with respect to the valuation ν; this is defined by induction on the structure of φ, in the expected way.

We write $\varphi(x_1, \dots, x_n)$ to denote that the free variables of φ are among $x_1, \dots, x_n$. Then, given elements $u_1, \dots, u_n$ in the universe of a structure $\mathcal{S}$, we say that $\varphi(u_1, \dots, u_n)$ is satisfied in $\mathcal{S}$ if φ is satisfied in $\mathcal{S}$ under the valuation mapping x_i to u_i for all $i \in \{1, \dots, n\}$.

An MSO-*interpretation* $I = (\delta, (\varphi_R)_{R \in \Sigma_2})$ from Σ_1 to Σ_2 consists of an MSO-formula $\delta(x)$ over Σ_1, and of MSO-formulae $\varphi_R(x_1, \dots, x_n)$ over Σ_1, for every relation name $R \in \Sigma_2$, where n is the arity of R. Having such an MSO-interpretation, we can apply it to a structure $\mathcal{S}$ over Σ_1; we obtain a structure $I(\mathcal{S})$ over Σ_2, where the universe $U^{I(\mathcal{S})}$ consists of those elements v of the universe of $\mathcal{S}$ for which $\delta(v)$ is satisfied in $\mathcal{S}$, and where every relation $R^{I(\mathcal{S})}$ consists of the tuples $(v_1, \dots, v_n) \in (U^{I(\mathcal{S})})^n$ for which $\varphi_R(v_1, \dots, v_n)$ is satisfied in $\mathcal{S}$.

Iterations and hierarchies. For a set A, by A^* (or A^+) we denote the set of all finite words (or all nonempty finite words, respectively) over alphabet A. In the sequel, we write $[a_1 a_2 \dots a_k]$ for a word consisting of letters $a_1, a_2, \dots, a_k$, and we use $\circ$ to denote concatenation of words (this notation allows us to unambiguously describe words of words).

Let $\mathcal{S} = (U^{\mathcal{S}}, R_1^{\mathcal{S}}, \dots, R_r^{\mathcal{S}})$ be a relational structure over a signature Σ, and let $\sharp, \& \not\in \Sigma$ be new relation names, where $\sharp$ is binary and $\&$ unary. The *basic (a.k.a. Shelah-Stupp's) iteration of* $\mathcal{S}$, denoted $\mathcal{S}^\sharp$, is a relational structure over $\Sigma \cup \{\sharp\}$, where

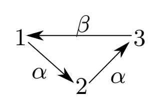
- the universe $U^{\mathcal{S}^\sharp}$ is $(U^{\mathcal{S}})^+$ (i.e., the set of nonempty words whose letters are elements of $U^{\mathcal{S}}$),
- the relation $R_i^{\mathcal{S}^\sharp}$ contains all tuples of the form $(w \circ [a_1], \dots w \circ [a_n])$ such that $(a_1, \dots, a_n) \in R_i^{\mathcal{S}}$ (where $w \in (U^{\mathcal{S}})^*$ and $a_1, \dots, a_n \in U^{\mathcal{S}}$), for every $i \in \{1, \dots, r\}$, and
- the relation $\sharp^{\mathcal{S}^\sharp}$ contains all pairs of the form $(w, w \circ [a])$ (where $w \in (U^{\mathcal{S}})^+$ and $a \in U^{\mathcal{S}}$).

The *Muchnik's iteration of* $\mathcal{S}$, denoted $\mathcal{S}^{\sharp,\&}$, is a relational structure over $\Sigma \cup \{\sharp, \&\}$, where

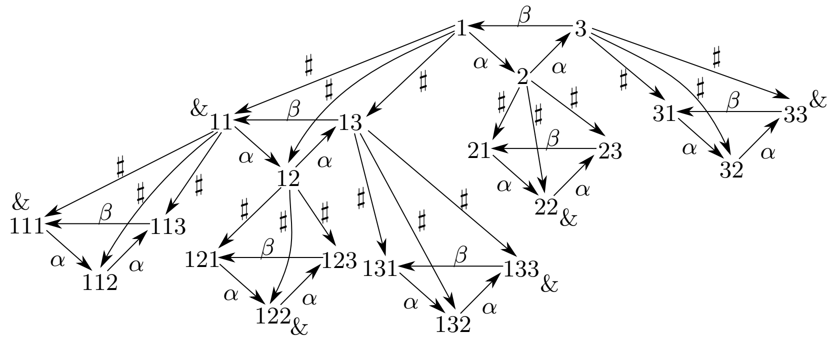
- the universe $U^{\mathcal{S}^{\sharp,\&}}$ and the relations $R_i^{\mathcal{S}^{\sharp,\&}}$ and $\sharp^{\mathcal{S}^{\sharp,\&}}$ are defined as in $\mathcal{S}^\sharp$,
- the relation $\&^{\mathcal{S}^{\sharp,\&}}$ contains all elements of the form $w \circ [aa]$ (where $w \in (U^{\mathcal{S}})^*$ and $a \in U^{\mathcal{S}}$).

In the sequel, we also use \$ as an alternative for the $\sharp$ symbol (and then we write $\mathcal{S}^\$$ instead of $\mathcal{S}^\sharp$).

Example 2.1. This example is borrowed from Caucal and Knapik [6]. Consider a structure $\mathcal{S}$ with universe $\{1, 2, 3\}$, and with the following binary relations α, β, depicted by arrows:



A fragment of Muchnik's iteration $\mathcal{S}^{\sharp,\&}$ has the following shape (the basic iteration $\mathcal{S}^\sharp$ looks similarly, except that the $\&$ predicate should be removed):



□

For every $n \in \mathbb{N}$ we define two families of relational structures, $\mathfrak{hgr}_n$ and $\mathfrak{hgr}_n^{\text{basic}}$, as follows:

$$\begin{aligned}
\mathfrak{hgr}_0 = \mathfrak{hgr}_0^{\text{basic}} &= \text{finite relational structures},\\
\mathfrak{hgr}_{n+1} &= \{I(\mathcal{S}^{\sharp,\&}) \mid \mathcal{S} \in \mathfrak{hgr}_n \wedge I \text{ is an MSO-interpretation}\},\\
\mathfrak{hgr}_{n+1}^{\text{basic}} &= \{I(\mathcal{S}^{\sharp}) \mid \mathcal{S} \in \mathfrak{hgr}_n^{\text{basic}} \wedge I \text{ is an MSO-interpretation}\}.
\end{aligned}$$

More formally, the classes contain also all structures isomorphic to the structures present in the above definition.

Recall that the composition of two MSO-interpretations is again an MSO-interpretation [10], that there exists an identity MSO-interpretation, and that a structure MSO-interpreted in a finite structure is again finite. It follows that the above definition (where iterations and MSO-interpretations appear alternatingly, starting from an iteration) covers any sequence of iterations and MSO-interpretations applied to a finite structure; then the level n counts the number of iterations.

3 Equality on level 1

In this section we concentrate on the first main result of the paper:

Theorem 3.1 (cf. Caucal and Knapik [6, Corollary 14]). *The classes $\mathfrak{hgr}_1$ and $\mathfrak{hgr}_1^{\text{basic}}$ coincide.*

In order to prove this result, Caucal and Knapik use a passage through prefix-recognizable structures. Below, we give a straightforward, direct proof.

Proof (Theorem 3.1). Clearly $\mathfrak{hgr}_1^{\text{basic}}$ is contained in $\mathfrak{hgr}_1$. In order to prove the other inclusion, consider a structure in $\mathfrak{hgr}_1$. It is of the form $I(\mathcal{S}^{\sharp,\&})$ for some finite structure $\mathcal{S}$ and MSO-interpretation I. We are going to prove that there is a finite structure $\mathcal{T}$ and an MSO-interpretation J such that $\mathcal{S}^{\sharp,\&} = J(\mathcal{T}^{\sharp})$. This shows that $I(\mathcal{S}^{\sharp,\&})$, which equals $I(J(\mathcal{T}^{\sharp}))$, belongs to $\mathfrak{hgr}_1^{\text{basic}}$, because a composition of two MSO-interpretations is again an MSO-interpretation.

As $\mathcal{T}$ we take $\mathcal{S}$ enriched with additional predicates (i.e., unary relations). Namely, for each element a of the universe, we have a predicate P_a that holds only in this element (it is important that the universe is finite, so we need only finitely many new predicates). The interpretation J leaves unchanged the universe, all the relations from the signature of $\mathcal{S}$, and the relation $\sharp$; they are simply inherited from $\mathcal{T}^{\sharp}$. We only need to define in $\mathcal{T}^{\sharp}$ the "clone" predicate &. It should hold in elements of the form $w \circ [aa]$. But such equality of the last two letters can be easily expressed in an MSO formula, by taking a disjunction over all possible elements a of the universe. Indeed, recall that the new P_a predicates check in $\mathcal{T}^{\sharp}$ whether the last letter is a, and that the $\sharp$ relation allows us to cut off the last letter. $\square$

4 Closure under basic iteration

We now come to the second main result of the paper:

Theorem 4.1 (cf. Caucal and Knapik [6, Theorem 15]). *For every $n \geq 1$, the class $\mathfrak{hgr}_n$ is closed under basic iteration.*

In particular, because $\mathfrak{hgr}_1^{\text{basic}} = \mathfrak{hgr}_1$ (cf. Theorem 3.1), and because a composition of two MSO-interpretations is an MSO-interpretation, we obtain that $\mathfrak{hgr}_2^{\text{basic}} = \mathfrak{hgr}_1$, and likewise $\mathfrak{hgr}_n^{\text{basic}} = \mathfrak{hgr}_1$ for all $n \geq 1$.

The key point in our proof of Theorem 4.1 is that a composition of a Muchnik's iteration with a basic iteration can be encoded in a single Muchnik's iteration. As already said in the introduction, this amounts to encoding a word of words in a single word. In this encoding, we need a separator to be inserted between concatenated words. We thus consider an operation of adding a distinguished element (which will become the separator) to an arbitrary structure.

Let $\mathcal{S} = (U^{\mathcal{S}}, R_1^{\mathcal{S}}, \dots, R_r^{\mathcal{S}})$ be a relational structure over a signature Σ, and let $\dagger \not\in \Sigma$ be a new unary relation name. The *single-element extension of* $\mathcal{S}$, denoted $\mathcal{S}_\dagger$, is a relational structure over $\Sigma \cup \{\dagger\}$, where

- the universe $U^{\mathcal{S}_\dagger}$ is $U^{\mathcal{S}} \uplus \{a_\dagger\}$, for some fresh element $a_\dagger$,
- $R_i^{\mathcal{S}_\dagger} = R_i^{\mathcal{S}}$ for all $i \in \{1, \dots, r\}$, and
- $\dagger^{\mathcal{S}_\dagger} = \{a_\dagger\}$.

In other words, we add a new element to the universe of $\mathcal{S}$, but not to any of the relations; the new unary predicate $\dagger$ holds only in the new element.

Below, the $\equiv$ symbol stands for isomorphism of structures.

Lemma 4.2. *Fix a signature Σ. There exists an* MSO*-interpretations I such that for every relational structure $\mathcal{S}$ over Σ,*

$$(\mathcal{S}^{\sharp,\&})^{\$} \equiv I((\mathcal{S}_\dagger)^{\sharp,\&}).$$

Proof. Let $\mathcal{S} = (U^{\mathcal{S}}, R_1^{\mathcal{S}}, \dots, R_r^{\mathcal{S}})$. Recall that the universe of the double iteration $(\mathcal{S}^{\sharp,\&})^{\$}$ is $((U^S)^+)^+$, and the universe of $(\mathcal{S}_\dagger)^{\sharp,\&}$ is $(U^{\mathcal{S}} \uplus \{a_\dagger\})^+$. We define an injective mapping $\mathsf{flat}\colon ((U^S)^+)^+ \to (U^{\mathcal{S}} \uplus \{a_\dagger\})^+$ by

$$\mathsf{flat}([w_1 w_2 \dots w_k]) = w_1 \circ [a_\dagger] \circ w_2 \circ [a_\dagger] \circ \dots \circ [a_\dagger] \circ w_k.$$

It thus concatenates the words $w_1, \dots, w_k$ being letters of $[w_1 w_2 \dots w_k]$, inserting the separator $a_\dagger$ between them.

The universe-restricting formula δ of the interpretation I should select elements of the image of flat. These are words such that there are no two $a_\dagger$ letters in a row, and the first and the last letter are not $a_\dagger$. This property can be easily expressed in MSO (recall that the $\dagger$ predicate in $(\mathcal{S}_\dagger)^{\sharp,\&}$ checks whether the last letter is $a_\dagger$, and the $\sharp$ relation allows to cut off the last letter of a word).

The interpretation I should not change relations R_i in any way, because the last letter of the last letter of an element of $((U^S)^+)^+$ (taken into account by the relations R_i in $(\mathcal{S}^{\sharp,\&})^{\$}$) is mapped by flat to the last letter of an element of $(U^{\mathcal{S}} \uplus \{a_\dagger\})^+$ (taken into account by the relations R_i in $(\mathcal{S}_\dagger)^{\sharp,\&}$).

The relations $\sharp$ and & should remain unchanged as well. Indeed, the $\sharp$ relation in $(\mathcal{S}^{\sharp,\&})^{\$}$ contains pairs of the form $(v \circ [w], v \circ [w \circ [a]])$. They are mapped by flat to $(\mathsf{flat}(v) \circ [a_\dagger] \circ w, \mathsf{flat}(v) \circ [a_\dagger] \circ w \circ [a])$ (or just $(w, w \circ [a])$, if v is empty), which are exactly pairs contained in the relation $\sharp$ in $(\mathcal{S}_\dagger)^{\sharp,\&}$, while restricted to the image of flat. Likewise, the & predicate in $(\mathcal{S}^{\sharp,\&})^{\$}$ holds in elements of the form $(v \circ [w \circ [aa]])$. They are mapped by flat to $(\mathsf{flat}(v) \circ [a_\dagger] \circ w \circ [aa])$ (or just $(w \circ [aa])$, if v is empty), which are exactly the elements of the image of flat for which the & predicate holds in $(\mathcal{S}_\dagger)^{\sharp,\&}$.

Finally, the relation \$ in $(\mathcal{S}^{\sharp,\&})^{\$}$ contains pairs of the form $(v, v \circ [w])$, which are mapped by flat to $(\mathsf{flat}(v), \mathsf{flat}(v) \circ [a_\dagger] \circ w)$. Thus, the formula $\varphi_{\$}(x, y)$ in I should say that x is obtained from y by cutting off the suffix starting from the last $a_\dagger$. As for δ, this can be easily expressed in MSO. $\square$

Remark 4.3. In Lemma 4.2 it is important that Muchnik's iteration is applied first, that is, that we take $(\mathcal{S}^{\sharp,\&})^{\$}$ and not $(\mathcal{S}^{\sharp})^{\$,\&}$. Indeed, the "clone" predicate of the second iteration would say that the last two words w_{k-1}, w_k in an encoding $w_1 \circ [a_\dagger] \circ \cdots \circ [a_\dagger] \circ w_{k-1} \circ [a_\dagger] \circ w_k$ are equal. We are unable to say this in MSO.

Lemma 4.2 eliminates a composition of iterations, at the cost of using the single-element extension. We thus need to prove that the classes $\mathfrak{hgr}_n$ are closed under the latter operation:

Lemma 4.4. *For every $n \geq 1$, the class $\mathfrak{hgr}_n$ is closed under taking single-element extensions.*

To this end, we need the following lemma:

Lemma 4.5. *Fix a signature Σ. There exists an* MSO*-interpretation I such that for every relational structure $\mathcal{S}$ over Σ,*

$$(\mathcal{S}^{\sharp,\&})_\dagger \equiv I((\mathcal{S}_\dagger)^{\sharp,\&}).$$

Proof. Let $\mathcal{S} = (U^{\mathcal{S}}, R_1^{\mathcal{S}}, \ldots, R_r^{\mathcal{S}})$. Recall that $(\mathcal{S}^{\sharp,\&})_\dagger$ extends $\mathcal{S}^{\sharp,\&}$ by a single fresh element, while in $(\mathcal{S}_\dagger)^{\sharp,\&}$ the fresh element can be used anywhere as a letter of a word. We define an injective mapping $\mathsf{inj}\colon (U^{\mathcal{S}})^+ \uplus \{a_\dagger\} \to (U^{\mathcal{S}} \uplus \{a_\dagger\})^+$ (i.e., from the universe of $(\mathcal{S}^{\sharp,\&})_\dagger$ to the universe of $(\mathcal{S}_\dagger)^{\sharp,\&}$) by

$$\mathsf{inj}(w) = \begin{cases} [a_\dagger] & \text{if } w = a_\dagger, \\ w & \text{otherwise.} \end{cases}$$

The universe-restricting formula δ of the interpretation I should select elements of the image of inj. These are words not using $a_\dagger$ as a letter, plus the length-1 word $[a_\dagger]$. Of course we can select such words in MSO (where we can use the $\dagger$ predicate to check whether the last letter is $a_\dagger$, and the $\sharp$ relation to cut off the last letter of a word). All the relations R_i, as well as $\sharp$ and &, should remain unchanged by the interpretation I. $\square$

The next two lemmata allow us to swap an MSO-interpretation with the operations of iteration or single-element extension.

Lemma 4.6. *Let I be an* MSO*-interpretation from a signature Σ to a signature Π. There exists an* MSO*-interpretation J such that for every relational structure $\mathcal{S}$ over Σ,*

$$(I(\mathcal{S}))^\sharp = J(\mathcal{S}^\sharp).$$

Proof. First, observe that we can write an MSO formula $\psi(X)$ saying that X is one of the copies in the iteration, (i.e., for some word w, the set X contains all words of the form $w \circ [a]$).

Let $I = (\delta, (\varphi_R)_{R\in\Sigma})$. Let $\delta'(x, X)$ be a formula obtained from $\delta(x)$ by relativizing to the set X (by saying that, we mean that all quantified objects should come from the set X, as well as x should belong to X). Then, let $\delta''(x) \equiv \exists X.(\psi(X) \wedge \delta'(x, X))$; this formula says that the last letter of x satisfies δ. Using δ'' we can easily write the universe-restricting formula of J, which should say that all letters of the considered word satisfy δ.

Likewise, we relativize every formula $\varphi_R(x_1, \dots, x_n)$ to a set X (saying in particular that all $x_1, \dots, x_n$ belong to X), obtaining a formula $\varphi'_R(x_1, \dots, x_n, X)$. Then, we take $\varphi''_R(x_1, \dots, x_n) \equiv \exists X.(\psi(X) \wedge \varphi'_R(x_1, \dots, x_n, X))$. This formula says that all $x_1, \dots, x_n$ belong to the same copy in the iteration, and that their last letters satisfy the formula φ_R; this is exactly the definition of R in $(I(\mathcal{S}))^\sharp$, so we can take φ''_R to the interpretation J.

Finally, J should leave the relation $\sharp$ unchanged. $\square$

The above lemma was also given by Caucal and Knapik [6, Lemma 16], but their proof contains a flaw. Namely, they propose to just make a conjunction of $\varphi_R(x_1, \dots, x_n)$ with a formula saying that all $x_1, \dots, x_n$ belong to the same copy in the iteration. This is not enough (for example, φ_R may contain a subformula, unrelated to the arguments $x_1, \dots, x_n$, saying that all elements of the structure are connected by some relation α, which is true in some $\mathcal{S}$, but not in $\mathcal{S}^\sharp$); all quantifiers in a formula have to be relativized to the same copy, as in our proof.

Lemma 4.7. *Let I be an* MSO*-interpretation from a signature Σ to a signature Π. There exists an* MSO*-interpretation J such that for every relational structure $\mathcal{S}$ over Σ,*

$$(I(\mathcal{S}))_\dagger = J(\mathcal{S}_\dagger).$$

Proof. Let $I = (\delta, (\varphi_R)_{R\in\Sigma})$. We relativize the formulae δ and φ_R to elements not satisfying $\dagger$ (i.e., elements of the original structure $\mathcal{S}$), obtaining δ' and φ'_R; in particular, these formulae say that their arguments do not satisfy $\dagger$. As the universe-restricting formula of J we take $\delta'(x) \vee \dagger(x)$, and we use formulae φ'_R to define relations. Additionally, we leave the $\dagger$ predicate unchanged. $\square$

We can now finish the proofs of Lemma 4.4 and Theorem 4.1.

Proof (Lemma 4.4). Induction on n. The base case of $n = 0$ is trivial: the single-element extension of a finite structure is again finite. For the induction step, assume that $\mathfrak{hgr}_n$ is closed under taking single-element extensions, and consider a structure from $\mathfrak{hgr}_{n+1}$; we have to prove that the single-element extension of this structure also belongs to $\mathfrak{hgr}_{n+1}$. The structure is of the form $I(\mathcal{S}^{\sharp,\&})$ for some $\mathcal{S} \in \mathfrak{hgr}_n$, and for some MSO-interpretation I. First, by Lemma 4.7, we can write $(I(\mathcal{S}^{\sharp,\&}))_\dagger = J((\mathcal{S}^{\sharp,\&})_\dagger)$, for some MSO-interpretation J. Then, by Lemma 4.5, we have $(\mathcal{S}^{\sharp,\&})_\dagger \equiv K((\mathcal{S}_\dagger)^{\sharp,\&})$, for some MSO-interpretation K. By the induction hypothesis, $\mathcal{S}_\dagger \in \mathfrak{hgr}_n$, so $J(K((\mathcal{S}_\dagger)^{\sharp,\&})) \in \mathfrak{hgr}_{n+1}$. $\square$

Proof (Theorem 4.1). Consider a structure from $\mathfrak{hgr}_n$, where $n \geq 1$; we have to prove that the basic iteration of this structure also belongs to $\mathfrak{hgr}_n$. The structure is of the form $I(\mathcal{S}^{\sharp,\&})$ for some $\mathcal{S} \in \mathfrak{hgr}_{n-1}$, and for some MSO-interpretation I. First, by Lemma 4.6, we can write $(I(\mathcal{S}^{\sharp,\&}))^\$ = J((\mathcal{S}^{\sharp,\&})^\$)$, for some MSO-interpretation J. Then, by Lemma 4.2, we have $(\mathcal{S}^{\sharp,\&})^\$ \equiv K((\mathcal{S}_\dagger)^{\sharp,\&})$, for some MSO-interpretation K. By Lemma 4.4 we know that $\mathcal{S}_\dagger \in \mathfrak{hgr}_{n-1}$, so $J(K((\mathcal{S}_\dagger)^{\sharp,\&})) \in \mathfrak{hgr}_n$. $\square$

Remark 4.8. In Lemma 4.2 we have shown that a composition of Muchnik's iteration with basic iteration can be encoded in the Muchnik's iteration of the single-element extension. Notice that by removing the "clone" predicate &, the same proof gives us another statement: a composition of two basic iterations can be encoded in a single basic iteration of the single-element extension.

Caucal and Knapik [6, Section 5] ask whether inside every $\mathfrak{hgr}_n$ class there exists a finer hierarchy, where one climbs up from one layer to the next layer via basic iteration. By the above, we know that two basic iterations can be rewritten using a single one; thus the answer to this question is negative, assuming that the classes would be closed under taking single-element extensions (which is a very natural assumption).

5 Conclusions

Caucal and Knapik [6] have proved that the $\mathfrak{hgr}_n^{\text{basic}}$ hierarchy, involving basic iterations, collapses at level 1, where it coincided with $\mathfrak{hgr}_1$, the first level of the hierarchy involving Muchnik's iteration. We have done the same, using much simpler methods. Moreover, we have given an additional insight on the nature of the two kinds of iterations: a composition of two basic iterations boils down to a single basic iteration, and a composition of Muchnik's iteration with basic iteration boils down to just a Muchnik's iteration. Simultaneously, Caucal's hierarchy (being a graph version of $\mathfrak{hgr}_n$) is strict, which implies that a composition of two Muchnik's iteration cannot be reduced to a single Muchnik's iteration, even if we only consider directed graphs instead of arbitrary relational structures.

References

1. Arnborg, S., Lagergren, J., Seese, D.: Easy problems for tree-decomposable graphs. J. Algorithms **12**(2), 308–340 (1991). https://doi.org/10.1016/0196-6774(91)90006-K

2. Büchi, J.R.: On a decision method in restricted second order arithmetic. In: Proceedings of the 1960 International Congress on Logic, Methodology and Philosophy of Science. pp. 1–11. Stanford University Press (1962)
3. Carayol, A., Wöhrle, S.: The Caucal hierarchy of infinite graphs in terms of logic and higher-order pushdown automata. In: Pandya, P.K., Radhakrishnan, J. (eds.) FST TCS 2003: Foundations of Software Technology and Theoretical Computer Science, 23rd Conference, Mumbai, India, December 15-17, 2003, Proceedings. Lecture Notes in Computer Science, vol. 2914, pp. 112–123. Springer (2003). https://doi.org/10.1007/978-3-540-24597-1_10
4. Caucal, D.: On infinite terms having a decidable monadic theory. In: Diks, K., Rytter, W. (eds.) Mathematical Foundations of Computer Science 2002, 27th International Symposium, MFCS 2002, Warsaw, Poland, August 26-30, 2002, Proceedings. Lecture Notes in Computer Science, vol. 2420, pp. 165–176. Springer (2002). https://doi.org/10.1007/3-540-45687-2_13
5. Caucal, D.: On infinite transition graphs having a decidable monadic theory. Theor. Comput. Sci. **290**(1), 79–115 (2003). https://doi.org/10.1016/S0304-3975(01)00089-5
6. Caucal, D., Knapik, T.: Shelah-Stupp's iteration and Muchnik's iteration. Fundam. Informaticae **159**(4), 327–359 (2018). https://doi.org/10.3233/FI-2018-1667
7. Courcelle, B.: The monadic second-order logic of graphs, II: infinite graphs of bounded width. Math. Syst. Theory **21**(4), 187–221 (1989). https://doi.org/10.1007/BF02088013
8. Courcelle, B.: The monadic second-order logic of graphs V: on closing the gap between definability and recognizability. Theor. Comput. Sci. **80**(2), 153–202 (1991). https://doi.org/10.1016/0304-3975(91)90387-H
9. Courcelle, B.: The monadic second-order logic of graphs VII: graphs as relational structures. Theor. Comput. Sci. **101**(1), 3–33 (1992). https://doi.org/10.1016/0304-3975(92)90148-9
10. Courcelle, B., Engelfriet, J.: Graph Structure and Monadic Second-Order Logic - A Language-Theoretic Approach, Encyclopedia of mathematics and its applications, vol. 138. Cambridge University Press (2012), http://www.cambridge.org/knowledge/isbn/item5758776/
11. Courcelle, B., Walukiewicz, I.: Monadic second-order logic, graph coverings and unfoldings of transition systems. Ann. Pure Appl. Log. **92**(1), 35–62 (1998). https://doi.org/10.1016/S0168-0072(97)00048-1
12. Elgot, C.C.: Decision problems of finite automata design and related arithmetics. Transactions of the American Mathematical Society **98**(1), 21–51 (1961). https://doi.org/10.2307/1993511
13. Elgot, C.C., Rabin, M.O.: Decidability and undecidability of extensions of second (first) order theory of (generalized) successor. J. Symb. Log. **31**(2), 169–181 (1966). https://doi.org/10.2307/2269808
14. Engelfriet, J.: A characterization of context-free NCE graph languages by monadic second-order logic on trees. In: Ehrig, H., Kreowski, H., Rozenberg, G. (eds.) Graph-Grammars and Their Application to Computer Science, 4th International Workshop, Bremen, Germany, March 5-9, 1990, Proceedings. Lecture Notes in Computer Science, vol. 532, pp. 311–327. Springer (1990). https://doi.org/10.1007/BFb0017397
15. Knapik, T., Niwiński, D., Urzyczyn, P.: Higher-order pushdown trees are easy. In: Nielsen, M., Engberg, U. (eds.) Foundations of Software Science and Computation Structures, 5th International Conference, FOSSACS 2002. Held as Part of the Joint

European Conferences on Theory and Practice of Software, ETAPS 2002 Grenoble, France, April 8-12, 2002, Proceedings. Lecture Notes in Computer Science, vol. 2303, pp. 205–222. Springer (2002). https://doi.org/10.1007/3-540-45931-6_15
16. Muller, D.E., Schupp, P.E.: The theory of ends, pushdown automata, and second-order logic. Theor. Comput. Sci. **37**, 51–75 (1985). https://doi.org/10.1016/0304-3975(85)90087-8
17. Ong, C.L.: On model-checking trees generated by higher-order recursion schemes. In: 21th IEEE Symposium on Logic in Computer Science (LICS 2006), 12-15 August 2006, Seattle, WA, USA, Proceedings. pp. 81–90. IEEE Computer Society (2006). https://doi.org/10.1109/LICS.2006.38
18. Rabin, M.O.: Decidability of second-order theories and automata on infinite trees. Transactions of the American Mathematical Society **141**, 1–35 (1969). https://doi.org/10.1090/S0002-9947-1969-0246760-1
19. Semenov, A.L.: Decidability of monadic theories. In: Chytil, M., Koubek, V. (eds.) Mathematical Foundations of Computer Science 1984, Praha, Czechoslovakia, September 3-7, 1984, Proceedings. Lecture Notes in Computer Science, vol. 176, pp. 162–175. Springer (1984). https://doi.org/10.1007/BFb0030296
20. Shelah, S.: The monadic theory of order. Annals of Mathematics **102**(3), 379–419 (1975). https://doi.org/10.2307/1971037
21. Siefkes, D.: Decidable extensions of monadic second order successor arithmetic. In: Automatentheorie und Formale Sprachen, Tagung, Math. Forschungsinst, Oberwolfach, 1969. pp. 441–472. Bibliograph. Inst., Mannheim (1970)
22. Stupp, J.: The lattice-model is recursive in the original model. Institute of Mathematics, The Hebrew University, Jerusalem (1975)
23. Thomas, W.: Das Entscheidungsproblem für einige Erweiterungen der Nachfolger-Arithmetik. Ph.D. thesis, Universität Freiburg (1975)
24. Trakhtenbrot, B.: Finite automata and the logic of monadic predicates. Doklady Akademii Nauk SSSR **140**, 326–329 (1961)
25. Walukiewicz, I.: Monadic second-order logic on tree-like structures. Theor. Comput. Sci. **275**(1-2), 311–346 (2002). https://doi.org/10.1016/S0304-3975(01)00185-2