# On the Complexity of Decision using Destinies in $H$-bounded Structures

A. Chateau

*LLAIC, IUT des Cézeaux - BP 86, 63172 Aubière Cedex, France*

**Abstract**

In this paper, two classes of decidable structures are compared. The first class is the class of $H$-bounded structures, in which quantifiers can be bounded by some recursive function $H$. The second class is the class of structures with recursive destinies, where destinies are finite organized trees embodying all Fraïsséan $k$-equivalence types of the structure. We show that $H$-bounded structures have recursive destinies, and we compare two decision algorithms in $H$-bounded structures: Ferrante-Rackoff's algorithm, which uses the function $H$ and elimination of quantifiers, and Nézondet's algorithm, which precomputes destinies and evaluates sentences on these finite trees. We also prove that the inclusion between the two classes is strict, by exhibiting a structure with recursive destinies which is not $H$-bounded. Finally, we characterize the class of $H$-bounded structures as the class of structures with strongly recursive destinies, *i.e.* for which there exists an algorithm constructing the destinies of any tuple of elements.

*Key words:* decidable structures, $H$-bounded, destinies, decision algorithm, complexity

## 1 Introduction

The original classical decision problem was pointed out by Hilbert ([HB34]), and has been widely studied since that time ([BGG97]). It can be formulated as follows: given a first-order sentence, decide whether it is consistent. However, the requirements of the decision problem considered in this paper are more strict, and involve a logical structure. Instead of deciding whether a first-order sentence has a model, we have to decide whether a sentence is true

*Email address:* chateau@llaic3.u-clermont1.fr (A. Chateau).
*URL:* http://llaic3.u-clermont1.fr/~chateau/ (A. Chateau).

in a given structure. Decidability of structures and its complexity have been widely studied for arithmetical structures, for instance $\langle \mathbb{N}, +, = \rangle$ (see [Pre29]), $\langle \mathbb{N}, \times, = \rangle$ (see [Mos52]), or $\langle \mathbb{N}, | \rangle$ (see [Mic81]). For a survey on decidability in arithmetical structures, we refer to [Bès02].

More precisely, a structure $\mathcal{M}$ on a signature $\sigma$ is decidable if there exists an algorithm giving the answer to "Is $\varphi$ true in $\mathcal{M}$?", on input a first-order $\sigma$-sentence $\varphi$. In the sequel, we consider only decidable structures on finite relational signatures. Two classes of decidable structures are studied. The first class was introduced by Ferrante and Rackoff in [FR79], and is the class of $H$-bounded structures. In those structures, bounding quantifiers by a recursive function $H$ leads to a decision algorithm using quantifier elimination, called Ferrante-Rackoff's algorithm. The second class of structures is the class of structures with recursive destinies. Destinies were introduced by Nézondet in [Néz97]. For a given structure $\mathcal{M}$, a $k$-destiny of $\mathcal{M}$ is a structured tree summing up all Fraïsséan $k$-equivalence types appearing in the structure $\mathcal{M}$. The main point is that a destiny of a structure can always be turned into a finite one, and that it is possible to recursively decide the satisfaction of a sentence in a finite destiny. In structures with recursive destinies, there exists an algorithm that constructs destinies. This algorithm can easily be extended to a decision algorithm, called Nézondet's algorithm, as detailled below.

The purpose of this paper is to compare both classes, and both corresponding algorithms.
Section 2 recalls the definition of $H$-bounded structures, and presents Ferrante-Rackoff's algorithm.
Section 3 provides a definition of destinies, explains the deep link between destinies, Ehrenfeucht games and Fraïssé's $k$-equivalence, and presents Nézondet's algorithm. An algorithm to construct destinies is given, which can be used under some conditions, and the inclusion of the class of $H$-bounded structures in the class of structures with recursive destinies is proven.
Section 4 compares, in $H$-bounded structures, the time complexity of Ferrante-Rackoff's and Nézondet's algorithms. Both algorithms have similar complexity, but Nézondet's one is more efficient when one wants to decide several sentences with same quantifier rank, instead of a single one.
Section 5 presents an example of a structure which has recursive destinies, but is not $H$-bounded. Hence, we prove that the inclusion of the class of $H$-bounded structures in the class of structures with recursive destinies is strict.
Finally, Section 6 characterizes $H$-bounded structures in terms of destinies, by enforcing the condition "to have recursive destinies" into "to have strongly recursive destinies". A structure has strongly recursive destinies if there exists an algorithm that constructs the destinies of all tuples of elements of the structure. The equality between the classes of $H$-bounded structures and structures with strongly recursive destinies is proved.

## 2 $H$-bounded structures

### 2.1 Definition

$H$-bounded structures were introduced by Ferrante and Rackoff in [FR79]. We use in this paper a slightly different definition, presented in [Mau94] or [Dub95].

**Definition 1 (Normed structure)**
*Let $\mathcal{M}$ be a structure on a finite relational signature $\sigma$. A* **norm** *on $\mathcal{M}$ is a mapping $||.||$ from the domain $M$ of $\mathcal{M}$ to a well-ordered set (e.g. $\mathbb{N}$). The normed structure is denoted by $(\mathcal{M}, ||.||)$.*

**Definition 2 ($H$-bounded structure)**
*Let $H : \mathbb{N}^3 \to \mathbb{N}$ be an increasing function of three variables (each increase of one variable makes the function increase too), and let $n, k, m \geq 0$ be arbitrary integers. Let $\mathcal{M}$ be a structure on a finite relational signature $\sigma$, and $(a_1, \ldots, a_k) \in M^k$ be such that $||a_i|| \leq m$ for each $i \in \{1, \ldots, k\}$. Let $\Phi(x_1, \ldots, x_k, y)$ be a formula of quantifier rank at most $n$ involving the free variables $x_1, \ldots, x_k$ and $y$, and let $\varphi(x_1, \ldots, x_k) := (\exists y)\Phi(x_1, \ldots, x_k, y)$.*
*The structure $\mathcal{M}$ is $H$-***bounded** *if $(\mathcal{M}, a_1, \ldots, a_k) \models \varphi(x_1, \ldots, x_k)$ if and only if there exists an element $a_{k+1} \in M$ with $||a_{k+1}|| \leq H(n, k, m)$ such that $(\mathcal{M}, a_1, \ldots, a_k, a_{k+1}) \models \Phi(x_1, \ldots, x_k, y)$.*

In what follows, we assume that there are a finite number of elements of $M$ with norm less than a given integer, and that the function $H$ is recursive. Similar restrictions are done in [Mau94] or [FR79]. Some examples of $H$-bounded structures with the corresponding function $H$ appear on Figure 1 (with canonical norm, *e.g.* the absolute value). Those results appear in [Mic81] and [FR79].

| Structure | Function $H(n, k, m)$ |
|---|---|
| $\langle \mathbb{N}, \leq \rangle$ | $m + 2^n$ |
| $\langle \mathbb{Z}, +, <, 0 \rangle$ | $(m+1) \times 2^{2^{2^{c(n+k)}}}$ |
| $\langle \mathbb{N}, \mid \rangle$ | $m + 2^{\frac{1}{2}(n+k+1)(n+1)(2k+n+2)}$ |

Fig. 1. Some examples of $H$-bounded structures

Under the above assumptions, we denote by $(\exists x \leq m)\Phi(x)$ the finite disjunction $\bigvee_{||x|| \leq m} \Phi(x)$ and by $(\forall x \leq m)\Phi(x)$ the finite conjunction $\bigwedge_{||x|| \leq m} \Phi(x)$.

**Definition 3 ($H$-relativized formula)**
*Let $H : \mathbb{N}^3 \to \mathbb{N}$ and $n, k \in \mathbb{N}$. Let $F(y_1, \dots, y_n)$ be a formula of quantifier rank at most $k$, and involving the free variables $y_1, \dots, y_n$. We suppose that the bound variables are denoted by $x_1, \dots, x_k$, where $x_k$ is the innermost. Let $m_0, \dots, m_k$ be a sequence of integers such that $m_0 \leq \dots \leq m_k$ and such that $H(k-i, n+i-1, m_{i-1}) \leq m_i$ for each $i \in \{1, \dots, k\}$. The* **$H$-relativized form of** *$F$, denoted by $\bar{F}_{\vec{m}}$, is produced from $F$, by replacing each quantifier of the form $Q_i x_i G(y_1, \dots, y_n, x_1, \dots, x_i)$ by $(Q_i x_i \leq m_i) G(y_1, \dots, y_n, x_1, \dots, x_i)$.*

We must notice that this relativization is compatible with Boolean operations on formulas. Using this relativized form, we can prove a quantifier elimination theorem in $H$-bounded structures. The proof of Theorem 1 appears in Appendix A. This theorem is an alternative version of the original theorem proved by Ferrante and Rackoff in [FR79]. Our version avoids a prenexisation step in the corresponding decision algorithm.

**Theorem 1**
*Let $\mathcal{M}$ be a $H$-bounded structure. Let $n, k \in \mathbb{N}$, and let $F(y_1, \dots, y_n)$ be a formula of quantifier rank at most $k$, and involving the free variables $y_1, \dots, y_n$. Let $(a_1, \dots, a_n)$ be a $n$-tuple of elements of $M$, and $m_0, \dots, m_k$ be a sequence of integers such that $m_0 \leq \dots \leq m_k$ and $H(k-i, n+i-1, m_{i-1}) \leq m_i$ for each $i \in \{1, \dots, k\}$, and $||a_j|| \leq m_0$ for each $j \in \{1, \dots, n\}$. Then $(\mathcal{M}, a_1, \dots, a_n) \models F(y_1, \dots, y_n)$ if and only if $(\mathcal{M}, a_1, \dots, a_n) \models \bar{F}_{\vec{m}}(y_1, \dots, y_n)$.*

### 2.3 Ferrante-Rackoff's algorithm

Under usual restrictions on the function $H$ and the norm, Theorem 1 provides a decision algorithm, called Ferrante-Rackoff's algorithm.

**Input:** a first-order $\sigma$-sentence $F$.
**Output:** `TRUE` if the sentence is true in $\mathcal{M}$, `FALSE` otherwise
(1) Compute a sequence of integers $m_0 \leq \dots \leq m_k$ such that $H(k-i, i-1, m_{i-1}) \leq m_i$ for each $i \in \{1, \dots, k\}$ (with an arbitrary chosen $m_0$)
(2) Evaluate the formula $\bar{F}_{\vec{m}}$, where all quantifications are relativized to a finite subset of $\mathcal{M}$'s universe.

## 3 Structures with recursive destinies

Destinies were introduced by Nézondet in [Néz97], and have been studied in [Cha03]. They are labelled trees summing up the $k$-equivalence types (on a Fraïssean meaning) in a structure. This section is quite long, but the careful definitions about destinies and their properties are useful to understand Nézondet's algorithm. First (Paragraph 3.1), a definition of the *complete destiny* of a structure is given. Paragraph 3.2 describes an isomorphism between subtrees of destinies, and states the essential connection between destinies, Fraïssé's equivalence and Ehrenfeucht's games. Paragraph 3.3 explains how to reduce the complete destiny to a finite isomorphic destiny called *simplified destiny*. Paragraph 3.4 exposes the equivalence between the satisfaction of sentences in a structure and the satisfaction of transformed sentences in simplified destinies of the structure. Then, Nézondet's algorithm is detailed in Paragraph 3.5. Paragraph 3.6 defines a bound on children of nodes in a destiny, which is used in Paragraph 3.7 to prove that $H$-bounded structures have recursive destinies.

Let $\mathcal{M}$ be a structure on a finite relational signature $\sigma$. Since we aim to describe a tree, we extend the signature $\sigma$ to the signature $\sigma' = \sigma \cup \{P, \emptyset\}$. The binary predicate $P$ is intended as the parenthood between nodes in the tree, and the constant $\emptyset$ as the root of the tree. A destiny $T$ of height $k$ of $\mathcal{M}$, also called $k$-destiny of $\mathcal{M}$, is a tree of root $\emptyset$ whose nodes are labelled with elements of $\mathcal{M}$, such that all the branches have length $k+1$, and such that the nodes on a same branch satisfy some predicate of $\sigma$ in $T$ if and only if their labels satisfy the same predicate in $\mathcal{M}$. If there are some constants in $\sigma$, we add to the domain as many elements as required to be the interpretations of these constants. We denote by $l(u)$ the label of a node $u$, and we suppose that the rank of a node is its distance to the root, so that the root has rank 0, its children have rank 1, and so on. We present in this paper a simplified version of the definition of destinies, designed for our aim. For a general presentation of destinies, we refer the reader to [Cha03].

### *3.1 Definition of destinies*

**Definition 4 (Complete destiny of a structure)**
*Let $k \geq 1$, and $n$ be integers. Let $\mathcal{M}$ be a structure on a finite relational signature $\sigma$, which contains $n$ symbols of constant $c_1, \ldots, c_n$. The* **complete destiny of height $k$ of $\mathcal{M}$**, *or* **complete $k$-destiny of $\mathcal{M}$**, *denoted by $T_c^k(\mathcal{M})$, is a $\sigma'$-structure such that :*

- *the domain of the complete $k$-destiny is of the form $U \cup \{a_1 \ldots, a_n\}$, where* *the constants $c_i$ are interpreted by the $a_i$'s, and the labelling function is extended such that $l(c_i^{T_c^k(\mathcal{M})}) = c_i^{\mathcal{M}}$;*

- *the constant $\emptyset$ is interpreted by an element of $U$, noted $\emptyset$ too;*
- *$(U, P, \emptyset)$ is a tree with all branches of length $k+1$, such that each non-leaf node has children labelled with all elements of $\mathcal{M}$;*
- *for each branch $B$ of $T_c^k(\mathcal{M})$, and for each predicate $R$ of $\sigma$ with arity $p$, and for each sequence $(x_1, \ldots, x_p)$ of nodes in $B$ with rank at least 1 or constants:*

$$T_c^k(\mathcal{M}) \models R^{T_c^k(\mathcal{M})}(x_1, \ldots, x_p) \text{ if and only if } \mathcal{M} \models R^{\mathcal{M}}(l(x_1), \ldots, l(x_p)).$$

- *the interpretations of the relations of $\sigma$ do not contain any tuples containing nodes of different branches, or containing the root $\emptyset$.*

Example: Let us consider the structure $\langle \mathbb{N}, \perp, = \rangle$. The complete 2-destiny of this structure, visible on Figure 2 is an infinite tree, so we cannot draw it completely.

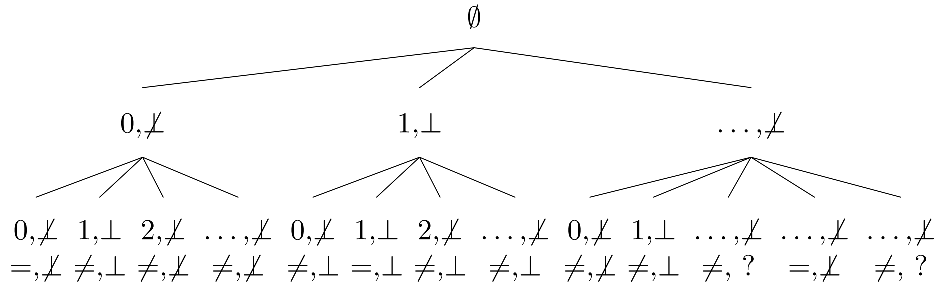


Fig. 2. The complete 2-destiny of the structure $\langle \mathbb{N}, \perp, = \rangle$.

A structure has several destinies, as we shall see later. However, the complete $k$-destiny of a structure is of particular importance, because it illustrates all Fraïssé's $k$-equivalence classes in $\mathcal{M}$, as we shall see in Theorem 2.

**Definition 5 (Subtree of root $u$ in a destiny)**
*Let $T$ be a $k$-destiny of domain $U \cup \{a_1, \ldots, a_n\}$, on the extended signature $\sigma'$, where $\sigma$ is a finite relational signature containing $n$ constant symbols. Let $u$ be a node of $T$ of rank $p \leq k$. The subtree of $u$, denoted by $ST(u)$, is the substructure of $T$ on the signature $\sigma'$ with domain* $\mathrm{Subtree}(u) \cup \{a_1, \ldots, a_n\} \cup \{\emptyset\}$, *where* $\mathrm{Subtree}(u)$ *is the set of nodes having $u$ as an ancestor (including $u$).*

**Definition 6 (Destiny of a tuple)**
*Let $\mathcal{M}$ be a $\sigma$-structure, and $(a_1, \ldots, a_n)$ be a tuple of elements of $\mathcal{M}$. A* **$k$-destiny of** *$(a_1, \ldots, a_n)$ is a $k$-destiny of the extended structure $(\mathcal{M}, a_1, \ldots, a_n)$.*

If the structure $\mathcal{M}$ is infinite, the complete $k$-destiny of $\mathcal{M}$ is also infinite. In order to obtain a finite equivalent object, we shall simplify the complete $k$-destiny by eliminating superfluous branches. We define a notion of isomorphism between subtrees of destinies, which induces an equivalence relation on subtrees of same rank. By deleting all but one subtree in each equivalence-class, we obtain a finite destiny as desired. Let us remark that this notion of isomorphism between subtrees is different from the classical one.

**Definition 7 (Isomorphism between subtrees of destinies)**
*Let $\sigma$ be a finite relational signature, and $k \geq 1$ be an integer. Let $T$ and $T'$ be two $k$-destinies on the extended signature $\sigma'$.*

- *Two leaves $u_k$ of $T$ and $u'_k$ of $T'$, with respective ancestors $u_{k-1}, \dots, u_1, \emptyset$ and $u'_{k-1}, \dots, u'_1, \emptyset$, are isomorphic if the tuples $(u_k, \dots, u_1)$ and $(u'_k, \dots u'_1)$ satisfy the same quantifier free formulas on the signature $\sigma'$;*
- *Two subtrees of $T$ and $T'$ with respective roots $u_p$ and $u'_p$ of rank $p < k$, with respective ancestors $u_{p-1}, \dots, u_1, \emptyset$ and $u'_{p-1}, \dots, u'_1, \emptyset$ are isomorphic if:*

*(1) the tuples $(u_p, \dots, u_1)$ and $(u'_p, \dots u'_1)$ satisfy the same quantifier free formulas on the signature $\sigma'$;*
*(2) supposing that the children of $u_p$ are the $(v_i)_{i \in I}$ and the children of $u'_p$ are the $(v'_j)_{j \in J}$, we have:*
  - *for each $i \in I$, there exists $j \in J$ such that $ST(v_i)$ and $ST(v'_j)$ are isomorphic;*
  - *for each $j \in J$, there exists $i \in I$ such that $ST(v_i)$ and $ST(v'_j)$ are isomorphic;*

We say that two destinies are isomorphic if the subtrees of their respective roots are isomorphic.

Example: if we consider the structure $\langle \mathbb{N}, \perp, = \rangle$, where $\perp$ is the coprimality predicate, and one of its 2-destinies presented on Figure 3, the leaves 0 and 6 in the subtree of root 2 are isomorphic. So are the subtrees of roots 0 and 7.

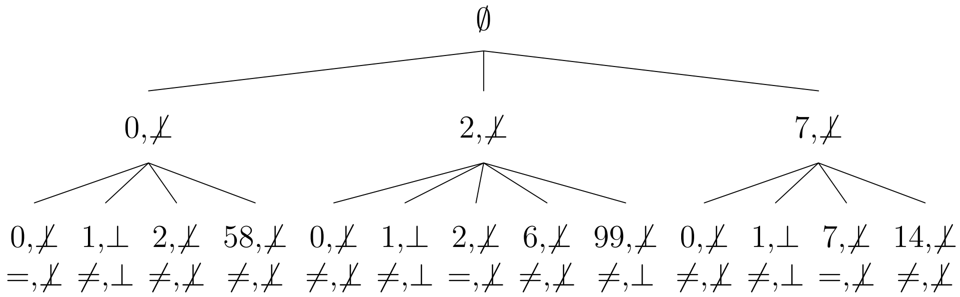


Fig. 3. A 2-destiny of the structure $\langle \mathbb{N}, \perp, = \rangle$ with isomorphic subtrees.

We said above that a complete destiny is a characterization of Fraïssé equivalence classes. The following theorem illustrates the meaning of this statement. For definitions of Fraïssé's equivalence, Ehrenfeucht's games or Hintikka's formulas, and the proof of the four first equivalences, one can refer to [EF95] or [EFT94]. The proof of the last equivalence is detailed in [Cha03] (the bases of the proof rely on Proposition 3, proven in Appendix C).

**Theorem 2**
*Let $\mathcal{A}$ and $\mathcal{B}$ be two $\sigma$-structures, $(a_1, \ldots, a_n)$ and $(b_1, \ldots, b_n)$ be two $n$-tuples of elements of $\mathcal{A}$ and $\mathcal{B}$ respectively. Let $k \geq 1$ be an integer. The following assertions are equivalent.*

*(1) $(a_1, \ldots, a_n)$ and $(b_1, \ldots, b_n)$ are $k$-equivalent (according to Fraïssé's meaning).*
*(2) Duplicator has a winning strategy for the Ehrenfeucht game in $k$ moves between the structures $(\mathcal{A}, a_1, \ldots, a_n)$ and $(\mathcal{B}, b_1, \ldots, b_n)$.*
*(3) $\mathcal{B} \models \varphi^k_{(a_1,\ldots,a_n)}(b_1, \ldots, b_n)$ (where $\varphi^k_{(a_1,\ldots,a_n)}$ denotes the Hintikka formula of rank $k$ of $(a_1, \ldots, a_n)$).*
*(4) $(a_1, \ldots, a_n)$ and $(b_1, \ldots, b_n)$ satisfy the same formulas of quantifier rank $k$ with $n$ free variables in their respective structures.*
*(5) The destinies $T^k_c((\mathcal{A}, a_1, \ldots, a_n))$ and $T^k_c((\mathcal{B}, b_1, \ldots, b_n))$ are isomorphic.*

Since there are only a finite number of Fraïssé's $k$-equivalence classes in a given structure, it can be deduced from this theorem that there are only a finite number of classes of isomorphism of subtrees in a complete $k$-destiny.

### *3.3 Simplified and reduced destinies*

We are now in a position to use the following simplification procedure on the complete $k$-destiny of $\mathcal{M}$:

- In each subtree of height 2, delete all but one representative for each class of isomorphism of leaves.
- In each subtree of height 3, delete all but one representative for each class of isomorphism of subtrees of height 2.
- ...
- In each subtree of height $k$, delete all but one representative for each class of isomorphism of subtrees of height $k-1$.
- Delete all but one representative for each class of isomorphism of subtrees of height $k$.

We obtain a destiny which is finite, as announced above.

**Definition 8 (Simplified destiny of a structure)**

*Let $k \geq 1$ be an integer. Let $\mathcal{M}$ be a structure on a finite relational signature $\sigma$. A* **simplified $k$-destiny of $\mathcal{M}$** *is a finite $\sigma'$-structure obtained by the simplification of $T_c^k(\mathcal{M})$ described above.*

Example: let us consider the structure $\langle \mathbb{N}, \mathcal{P}, <, = \rangle$, where $\mathcal{P}$ is the primality predicate. A simplified 2-destiny of this structure is presented on Figure 4. Prime labels are boxed.

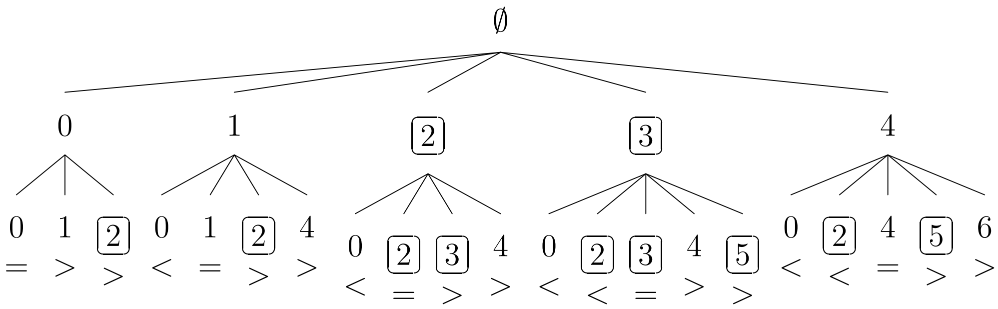

Fig. 4. A simplified 2-destiny of the structure $\langle \mathbb{N}, \mathcal{P}, <, = \rangle$.

In normed structures, we can define a reduced destiny, which is simplified and has minimal labels on its nodes.

**Definition 9 (Reduced destiny)**
*Let $T$ be a simplified $k$-destiny of a normed structure $\mathcal{M}$. The destiny $T$ is* **reduced** *if for each of its subtrees, the root of which is denoted by $u$, there does not exist in the complete $k$-destiny of $\mathcal{M}$ any isomorphic subtree with root denoted by $v$ such that $||l(v)|| < ||l(u)||$.*

It means that, at every step of the simplification procedure described above, we select a subtree with root of minimal label among all isomorphic subtrees. From now on, we consider only simplified destinies, and if the structures are normed, only reduced destinies.

### *3.4 Satisfaction in destinies*

We show that a sentence $\varphi$ of quantifier rank $k$ is true in $\mathcal{M}$ if and only if it is true in a simplified $k$-destiny of $\mathcal{M}$. Since relations between nodes of a destiny are relevant only when the nodes are on a same branch, we must transform a bit the sentence. We call this transformed sentence the *destinal form* of $\varphi$.

Let $F(y_1, \ldots, y_n)$ be a $\sigma$-formula of quantifier rank at most $k$. We suppose without any loss of generality that the bound variables of $F$ are called $x_1, \ldots, x_k$, where $x_k$ is the innermost. We define the quantifier relativization "~" on $F$ as follows. For $i$ from 1 to $k$,

- sub-formulas of $F$ of the form $\forall x_j G(y_1, \ldots, y_n, x_1, \ldots x_j)$ turn into $\forall x_j (P(x_{j-1}, x_j) \rightarrow G(y_1, \ldots, y_n, x_1, \ldots x_j))$;
- sub-formulas of $F$ of the form $\exists x_j G(y_1, \ldots, y_n, x_1, \ldots x_j)$ turn into $\exists x_j (P(x_{j-1}, x_j) \wedge G(y_1, \ldots, y_n, x_1, \ldots x_j))$.

In case where $j = 1$, we set $x_0 = \emptyset$. We obtain a $\sigma'$-formula with same quantifier rank and free variables as $F$.

**Definition 10 (Destinal form)**
*Let $F(y_1, \ldots, y_n)$ a $\sigma$-formula. The* **destinal form of** *$F$ is the $\sigma'$-formula $\tilde{F}(y_1, \ldots, y_n)$.*

Example: we consider the signature $\sigma = \{C, R\}$, where $C$ is a unary predicate, and $R$ is a binary predicate. We consider the following formula $F$:

$$R(x_1, x_2) \wedge \forall x_3 [\neg C(x_3) \vee \exists x_4 (R(x_3, x_4) \vee C(x_2))].$$

The destinal form of $F$ is the formula on the extended signature $\sigma \cup \{\emptyset, P\}$:

$$R(x_1, x_2) \wedge \forall x_3 (P(x_2, x_3) \rightarrow [\neg C(x_3) \vee \exists x_4 (P(x_3, x_4) \wedge (R(x_3, x_4) \vee C(x_2)))]).$$

The following theorem establishes a link between the satisfaction of sentences in a structure $\mathcal{M}$ and the satisfaction of their destinal form in a simplified destiny. It can by proved by induction on the structure of formulas (see Appendix B).

**Theorem 3**
*Let $\sigma$ be a finite relational signature, $\mathcal{M}$ be a $\sigma$-structure and $F$ be a $\sigma$-sentence of quantifier rank $k \geq 1$. Let $T$ be a simplified $k$-destiny of $\mathcal{M}$. Then $\mathcal{M} \models F$ if and only if $T \models \tilde{F}$, where $\tilde{F}$ is the destinal form of the sentence $F$.*

### *3.5 Nézondet's decision algorithm*

This theorem provides a decision algorithm: assuming that a simplified $k$-destiny $T$ of $\mathcal{M}$ is known, for any $\sigma$-sentence $F$ of quantifier rank $k$, we have to evaluate its destinal form $\tilde{F}$ on the finite tree $T$. Thus each variable ranges over a finite number of values. The algorithm is presented on Figure 5.

This algorithm requires the knowledge of a simplified $k$-destiny of the structure. But it is not always possible to construct such a destiny recursively. For instance, in case where the structure is not decidable, such an algorithm cannot exist. On the other hand, even if the structure is decidable, we have no general method to exhibit an algorithm constructing a simplified $k$-destiny for

**Input:** a $\sigma$-sentence $F$ of quantifier rank $k$
**Output:** `TRUE` if the sentence is true in $\mathcal{M}$, `FALSE` else
**Hypothesis:** a simplified $k$-destiny $T$ of $\mathcal{M}$ is known

(1) Current node is set to $\emptyset$.
(2) If $F$ is $G \wedge H$, then $\tilde{F} = \tilde{G} \wedge \tilde{H}$: evaluate $\tilde{G}$ and $\tilde{H}$ in $T$. Return `TRUE` if $\tilde{G}$ and $\tilde{H}$ are true in $T$, else return `FALSE`.
(3) If $F$ is $G \vee H$, then $\tilde{F} = \tilde{G} \vee \tilde{H}$: evaluate $\tilde{G}$ and $\tilde{H}$ in $T$. Return `TRUE` if $\tilde{G}$ or $\tilde{H}$ is true in $T$, else return `FALSE`.
(4) If $F$ is $\neg G$, then $\tilde{F} = \neg\tilde{G}$: evaluate $\tilde{G}$ in $T$. Return `TRUE` if $\tilde{G}$ is false in $T$, else return `FALSE`.
(5) If $F$ is $\forall x_i G(a_1, \ldots, a_{i-1}, x_i)$, then its destinal form is $\tilde{F} = \forall x_i(P(a_{i-1}, x_i) \rightarrow \tilde{G}(a_1, \ldots, a_{i-1}, x_i))$: current node is $a_{i-1}$, with ancestors $a_{i-2}, \ldots, a_1, \emptyset$. For each child $a_i$ of current node, evaluate $\tilde{G}(a_1, \ldots, a_{i-1}, a_i)$ in $T$ (current node is set to $a_i$ during this step). Return `TRUE` if for each of these evaluations $\tilde{G}(a_1, \ldots, a_{i-1}, a_i)$ is true in $T$, else return `FALSE`.
(6) If $F$ is $\exists x_i G(a_1, \ldots, a_{i-1}, x_i)$, then its destinal form is $\tilde{F} = \exists x_i(P(a_{i-1}, x_i) \wedge \tilde{G}(a_1, \ldots, a_{i-1}, x_i))$: current node is $a_{i-1}$, with ancestors $a_{i-2}, \ldots, a_1, \emptyset$. For each child $a_i$ of current node, evaluate $\tilde{G}(a_1, \ldots, a_{i-1}, a_i)$ in $T$ (current node is set to $a_i$ during this step). Return `TRUE` if for at least one of these evaluations $\tilde{G}(a_1, \ldots, a_{i-1}, a_i)$ is true in $T$, else return `FALSE`.
(7) If $F$ is atomic, then $\tilde{F} = F$: return `TRUE` if this formula is true in $T$, else return `FALSE`.

Fig. 5. Decision algorithm

each $k$. However, we will show that such an algorithm exists in $H$-bounded structures.

The definition of structures with recursive destinies was introduced by Cegielski in [Ceg03].

**Definition 11 (Structure with recursive destinies)**
*A $\sigma$-structure $\mathcal{M}$ is said to* **have recursive destinies** *if there exists an algorithm that produces a simplified $k$-destiny of $\mathcal{M}$ on input $k$.*

Let us now assume that $\mathcal{M}$ is a structure with recursive destinies. Then we have a full decision algorithm. If a sentence of quantifier rank $k$ is to be decided:

- construct a simplified $k$-destiny of $\mathcal{M}$,
- evaluate the sentence on this destiny.

Note that, if we have a lot of sentences of quantifier rank $k$ to decide, only the second step has to be repeated, because the $k$-destiny can be stored after the first computation.

### *3.6 Bounds on the children of a node and constructing algorithm*

A theoretical way to compute destinies of any height is, for a given $k$, to construct the complete $k$-destiny of the structure, and then to simplify isomorphic subtrees as described in Paragraph 3.3. The problem is that, when the structure is infinite, the complete $k$-destiny is infinite too. So we must truncate the construction of the complete $k$-destiny, since we have to work on finite objects. The proper notion for this operation is the *bound on the children*.

**Definition 12 (Bound on the children of a node)**
*Let $(\mathcal{M}, ||.||)$ be a normed $\sigma$-structure. Let $k \geq 1$ be an integer and $T$ be a $k$-destiny of $\mathcal{M}$. Let $u$ be a node of $T$ of rank $p < k$. Let $(u_i)_{i \in I}$ be the children of $u$. The* **bound on the children of** *$u$, denoted by $\mathrm{Sup}_c^T(u)$, is defined by $\mathrm{Sup}_c^T(u) = \sup_{i \in I}(||l(u_i)||)$.*

Let us assume now that, considering $T$ a reduced $k$-destiny, this bound is recursive, and that there are a finite number of elements of $\mathcal{M}$ with norm less than a given integer, as previously stated (see the definition of $H$-bounded structures). In this case, the following algorithm constructs a reduced $k$-destiny for each $k$.

**Input:** $k$
**Output:** a reduced $k$-destiny of $\mathcal{M}$
(1) Construct a finite tree with “sufficiently many nodes”
(2) Construct the lists of relations between the nodes and their ancestors
(3) Simplify the destiny

Let us detail the first step: we construct a finite tree with branches of length $k+1$ and root $\emptyset$ in the following way.

- First the tree is reduced to the root $\emptyset$. We add to this root as many children as necessary so that they can be labelled by all the elements of $\mathcal{M}$ with norm less than or equal to $\mathrm{Sup}_c^T(\emptyset)$.
- To each node $x_1$, we add as many children as necessary so that they can be labelled by all the elements of $\mathcal{M}$ with norm less than or equal to $\mathrm{Sup}_c^T(x_1)$.
- ...
- If all nodes of rank $p < k$ have been added, the rank $p+1$ is constructed by adding to each node $u$ of rank $p$ as many children as necessary so that they can be labelled by all the elements of $\mathcal{M}$ with norm less than or equal

to $\mathrm{Sup}_c^T(u)$.

- ...
- If all nodes of rank $k-1$ have been added, the rank $k$ is constructed by adding to each node $u$ of rank $k-1$ as many children as necessary so that they can be labelled by all the elements of $\mathcal{M}$ with norm less than or equal to $\mathrm{Sup}_c^T(u)$.

This construction guarantees that the corresponding destiny is isomorphic to a reduced destiny. We simplify it, in order to obtain a reduced $k$-destiny. The construction can be used as soon as a recursive upper bound of $\mathrm{Sup}_c^T$ is known. Instead of adding children with norm less than or equal to $\mathrm{Sup}_c^T$, we add children with norm less than or equal to the upper bounding function, and it runs as before.

Let us remark that it is not necessarily the only way to compute a reduced $k$-destiny, but we show in Paragraph 3.7 that this method can be applied in $H$-bounded structures, by computing an upper bound of $\mathrm{Sup}_c^T$ using the function $H$.

### *3.7 H-bounded structures have recursive destinies*

In this paragraph, we show that $H$-bounded structures have recursive destinies. More precisely, we prove that the bounds on the children of nodes of a reduced destiny are recursively upper bounded in $H$-bounded structures. The proof of Theorem 4 appears in Appendix C.

**Theorem 4**
*Let $\mathcal{M}$ be a $H$-bounded $\sigma$-structure. Let $k \geq 1$ and $T$ a reduced $k$-destiny of $\mathcal{M}$. Let $\alpha_p$ be a node of $T$ of rank $p < k$, with ancestors $\alpha_{p-1}, \ldots, \alpha_1, \emptyset$. We note $m = \max(||l(\alpha_p)||, \ldots, ||l(\alpha_1)||)$. Then $\mathrm{Sup}_c^T(\alpha_p) \leq H(k-p-1, p, m)$.*

Since the function $H$ is supposed to be recursive, so are the bounds on the children of nodes in a reduced destiny of $\mathcal{M}$.

| Structure | Function $H(n,k,m)$ | Upper bound of $\mathrm{Sup}_c^T(\alpha_p)$ |
|---|---|---|
| $\langle \mathbb{N}, \leq \rangle$ | $m + 2^n$ | $m + 2^{k-p-1}$ |
| $\langle \mathbb{Z}, +, <, 0 \rangle$ | $(m+1) \times 2^{2^{2^{c(n+k)}}}$ | $(m+1) \times 2^{2^{2^{c(k-1)}}}$ |
| $\langle \mathbb{N}, \vert \rangle$ | $m + 2^{\frac{1}{2}(n+k+1)(n+1)(2k+n+2)}$ | $m + 2^{\frac{1}{2}k(k-p)(p+k+1)}$ |

Fig. 6. Some examples of bound on the children of nodes in $H$-bounded structures

We have now two decision algorithms in $H$-bounded structures, whose complexities are compared in Section 4:

- the classical decision algorithm using quantifier elimination, that we called Ferrante-Rackoff's algorithm,
- the two-step decision algorithm using destinies, that we called Nézondet's algorithm.

## 4 Comparison between Ferrante-Rackoff's algorithm and Nézondet's algorithm

In order to determine the complexities of both algorithms, we have to index carefully each sub-formula of the input sentence, and each node of the reduced destiny. The details of this indexing and of the complexity computation can be found in [Cha03].

### *4.1 Complexity of Ferrante-Rackoff's algorithm*

We assume that:

- $k$ is the quantifier rank of the input sentence;
- $n$ is the length of the input sentence;
- $T_1(k)$ is the time required to decide an atomic formula with $k$ variables in the structure.
- $m_0 < \ldots < m_k$ is a sequence of integers such that for $i \in \{0, \ldots, k-1\}$, we have $m_{i+1} = H(i, k-i-1, m_i)$;
- as we supposed that there are only a finite number of elements of the structure with norm less than a given integer, we can define the function $K : \mathbb{N} \to \mathbb{N}$ such that $K(p)$ is the number of elements of the structure with norm less than or equal to $p$.

By a careful use of the indexing of sub-formulas of the input sentence, we obtain that the complexity of Ferrante-Rackoff's algorithm (in the classical model) is:

$$\mathcal{O}(\prod_{i=0}^{k} K(m_i) \times T_1(k) \times n).$$

Assuming that there are many sentences of same quantifier rank $k$ to decide, say $q$ sentences of total length $N$, the complexity of the total decision procedure is

$$\mathcal{O}(\prod_{i=0}^{k} K(m_i) \times T_1(k) \times N).$$

### 4.2 Complexity of Nézondet's algorithm

The complexity of Nézondet's algorithm is to be divided into two parts:

- the time of construction of a reduced $k$-destiny,
- the time of evaluation of the sentence on this $k$-destiny.

As we use the function H to compute an upper bound of the bounds on children of nodes, the construction of a reduced $k$-destiny has complexity

$$\mathcal{O}(\prod_{i=0}^{k} K(m_i) \times T_1(k)).$$

We obtain a reduced $k$-destiny with $N_k < \prod_{i=0}^{k} K(m_i)$ nodes. The evaluation time of a sentence with length $n$ and quantifier rank $k$ on this $k$-destiny is

$$\mathcal{O}(N_k \times n \times \Gamma_k(\sigma)),$$

where $\Gamma_k(\sigma)$ is the time needed to read in the list of relations satisfied by a leaf and its ancestors on the reduced $k$-destiny.

So, the total complexity of decision of one sentence with length $n$ and quantifier rank $k$ is:

$$\mathcal{O}(\prod_{i=0}^{k} K(m_i) \times T_1(k) + N_k \times n \times \Gamma_k(\sigma)).$$

Assuming that there are many sentences of same quantifier rank $k$ to decide, say $q$ sentences of total length $N$, the complexity of the total decision procedure is

$$\mathcal{O}(\prod_{i=0}^{k} K(m_i) \times T_1(k) + N_k \times N \times \Gamma_k(\sigma)).$$

### 4.3 Comparison

First, we have to notice that complexities are essentially the same. This is not surprising, since both algorithms consist in a quantifier elimination using the same bounds on quantifications.

However, in the case where there is a lot of sentences to decide, Nézondet's algorithm is better. The precomputation of the $k$-destiny avoids to decide atomic formulas in the structure each time we have to decide a sentence: this computation is done just once, and then is stored in the $k$-destiny. Both algorithms also do not rely on the same philosophy: Ferrante-Rackoff's algorithm uses a "lazy strategy", meaning that a predicate is evaluated on a tuple only when needed, whereas Nézondet's algorithm uses a "mass strategy", meaning that it precomputes a finite $k$-destiny that can be used to decide all the sentences with quantifier rank $k$.

Sometimes, this efficiency of Nézondet's algorithm, compared to that of Ferrante-Rackoff's algorithm, is even more improved by another phenomenon, due to the structure itself. Indeed, in most cases, the $k$-destiny is "lacunar": the number of nodes in the reduced destiny divided by the number of instanciations of variables in Ferrante-Rackoff algorithm is sometimes very small. During the construction of the reduced destinies, all redundant configurations have been eliminated, which is not the case in Ferrante-Rackoff's algorithm. So the instanciations of variables only occur on the smallest possible number of values. For instance, in the 3-destiny of the integers with successor and coprimality $\langle \mathbb{N}, S, \perp \rangle$, there is no node of rank 1 with label between 33554431 and $2^{227} - 1$ (see [Cha03]). With Ferrante-Rackoff's method, such gaps are not detected, so the variable $x_1$ would take all values between 33554431 and $2^{227} - 1$, even if it is not necessary. So, the fact that $N_k$ can be very smaller than $\prod_{i=0}^{k} K(m_i)$ has to be considered in the choice of the decision algorithm.

## 5 Not all structures with recursive destinies are $H$-bounded

Since we showed that the class of $H$-bounded structures is included in the class of structures with recursive destinies, now it is natural to ask the question of the converse inclusion. The answer is that the inclusion is strict, that is to say: there exists a normed $\sigma$-structure, which is not $H$-bounded for any recursive function $H$, but has recursive destinies. The main idea, inspired by Yuri Matiyasevich [1], consists in embedding an undecidable set of integers into a decidable structure with trivial destinies.

### *5.1 Presentation of the structure*

Let $\mathcal{U}$ be an undecidable set of integers with a diophantine characterization, which means that there exists a polynomial $P$ with $k+1$ variables ($k \geq 2$), such that $a \in \mathcal{U}$ if and only if $\exists x_1 \dots \exists x_k \; P(a, x_1, \dots, x_k) = 0$. Moreover, we can assume, without any loss of generality, that:

- the variables $x_1, \dots, x_k$ range over $\mathbb{N}$;
- $k = 2$, and $P$ is an exponential polynomial with three variables only.

[1] Steklov Institute, St. Petersbourg

The results allowing those restrictions appear in [Mat93] (sections 1.3 and 8.2 respectively). An exponential polynomial is defined as an expression built with variables and integers, using addition, multiplication and exponentiation. So, there exists an exponential polynomial $Q$ with three variables, such that:

$$a \in \mathcal{U} \text{ if and only if } \exists x_1 \geq 0 \; \exists x_2 \geq 0 \; Q(a, x_1, x_2) = 0.$$

The set $\mathcal{U}$ is countable, and its ordered elements are noted $a_0 < \ldots < a_n, \ldots$

We are now in a position to begin the description of the structure. The domain is the set of integers $\mathbb{Z}$, which contains the set $\mathcal{U}$. The signature consists in a unique predicate $\mathcal{Q}$ with arity 3. The predicate $\mathcal{Q}$ is interpreted as an extension of the set of roots of the polynomial $Q$, as we shall see in Paragraph 5.2.

We construct this interpretation in such a way that:

- all syntactically consistent atomic formulas are satisfiable in the structure $\langle \mathbb{Z}, \mathcal{Q} \rangle$, so that destinies are easy to describe (all possible branches appear);
- for all $a \in \mathcal{U}$, every pair $(x_1, x_2)$ satisfying $\mathcal{Q}(a, x_1, x_2)$ is "further" (in a meaning defined later) than the smallest pair of integers providing a root to the polynomial $Q(a, ., .)$.

Let us precise both properties.

**Definition 13 (Full destinies)**
*Let $T$ be a $k$-destiny of the structure $\langle \mathbb{Z}, \mathcal{Q} \rangle$. The destiny is said to be* **full** *if for each node $x_p$ with ancestors $x_{p-1}, \ldots, x_1$, the children of the node $x_p$ are such that:*

- *considering all terms $\mathcal{Q}(a, b, c)$, with $a$, $b$, and $c$ belonging to $\{y, x_1, \ldots, x_p\}$, and with at least one of them equal to $y$,*
- *considering all conjunctions of all of these terms, negated or not,*

*for each of these conjunctions, there exists a child of $x_p$ satisfying it.*

Example: let us construct the formal full 2-destiny (where "formal" means "without instantiation of variables"). The root is $\emptyset$.

- We consider all terms $\mathcal{Q}(a, b, c)$, with $a$, $b$, and $c$ belonging to $\{y\}$, and with at least one of them equal to $y$: there is only one, $\mathcal{Q}(y, y, y)$.
- We consider conjunctions of all of these terms, with a negation or not: there are two, $\mathcal{Q}(y, y, y)$ and $\neg\mathcal{Q}(y, y, y)$.

So there are two nodes of rank 1.

For each of these nodes of rank 1:

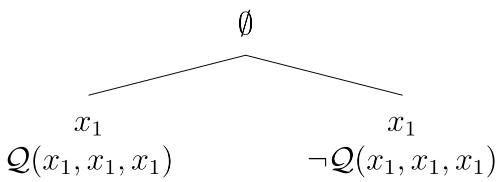


Fig. 7. Formal full 1-destiny

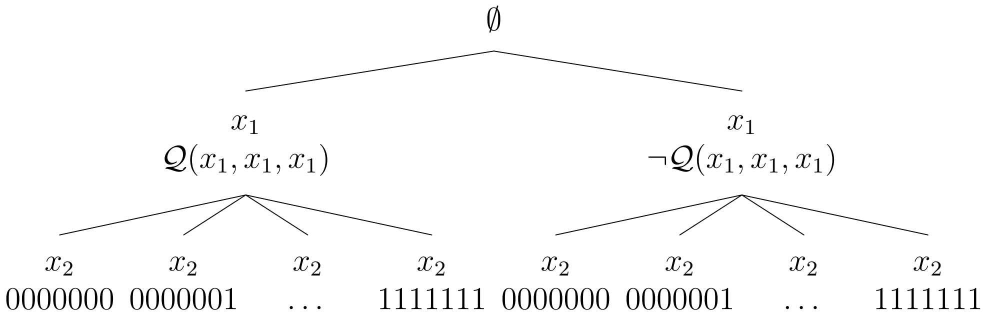


Fig. 8. Formal full 2-destiny

- We consider all terms $\mathcal{Q}(a, b, c)$, with $a$, $b$, and $c$ belonging to $\{y, x_1\}$, and with at least one of them equal to $y$: there are seven, $\mathcal{Q}(y, y, y)$, $\mathcal{Q}(y, y, x_1)$, $\mathcal{Q}(y, x_1, y)$, $\mathcal{Q}(x_1, y, y)$, $\mathcal{Q}(y, x_1, x_1)$, $\mathcal{Q}(x_1, y, x_1)$, $\mathcal{Q}(x_1, x_1, y)$.
- We consider conjunctions of all of these terms, with a negation or not: there are $2^7 = 128$, of the form
$$\varepsilon_1\mathcal{Q}(y, y, y) \wedge \varepsilon_2\mathcal{Q}(y, y, x_1) \wedge \varepsilon_3\mathcal{Q}(y, x_1, y) \wedge \varepsilon_4\mathcal{Q}(x_1, y, y) \wedge$$
$$\varepsilon_5\mathcal{Q}(y, x_1, x_1) \wedge \varepsilon_6\mathcal{Q}(x_1, y, x_1) \wedge \varepsilon_7\mathcal{Q}(x_1, x_1, y),$$
where the $\varepsilon_i$'s are $\neg$ or nothing. It is possible to encode such a conjunction into a 7-bit word: $w_1 \dots w_7$, where $w_i = 1$ if and only if $\varepsilon_i$ is nothing.

There are 256 nodes of rank 2. Notice that the size of full destinies grows exponentially with their height.

The second property involves the polynomial $Q$. We fix $a \in \mathcal{U}$. We consider the pairs $(x_1, x_2)$ of $\mathbb{N}^2$ such that $Q(a, x_1, x_2) = 0$. There exists at least one such pair, by characterization of the set $\mathcal{U}$ using the roots of $Q$. So let $R(a)$ be the non-empty set $R(a) = \{(x_1, x_2) \in \mathbb{N}^2 / Q(a, x_1, x_2) = 0\}$. Let $||.||$ be the Euclidian norm on $\mathbb{Z}^2$:
$$\begin{array}{rcl} \mathbb{Z}^2 & \to & \mathbb{N} \\ (u, v) & \mapsto & u^2 + v^2 \end{array}$$
We denote by $\mu(a)$ the minimum of $||.||$ on $R(a)$.

**Definition 14 (Distance property for $\mathcal{Q}$)** *The predicate $\mathcal{Q}$ has the* **distance property** *in $\mathbb{Z}$ if, for each $a \in \mathcal{U}$, and each pair $(u, v) \in \mathbb{Z}^2$ such that $\mathcal{Q}(a, u, v)$, then: $||(u, v)|| \geq \mu(a)$.*

This definition means that, if $\mathcal{Q}$ has the distance property, all pairs "near" $(0, 0)$ satisfy $\neg\mathcal{Q}(a, u, v)$. This property is essential to prove that the structure is not $H$-bounded, as we shall see in Paragraph 5.3.

### *5.2 Interpretation of the predicate $\mathcal{Q}$*

We now define the interpretation of $\mathcal{Q}$ in $\mathbb{Z}$ such that the destinies are full and $\mathcal{Q}$ satisfies the distance property. So doing, we can deduce an algorithm to construct a simplified $k$-destiny for all $k$. The method consists in defining the interpretation of $\mathcal{Q}$ on different subdomains in the same time that we construct destinies of all elements of these subdomains.

**Interpretation on $\mathbb{N}^3$**

On $\mathbb{N}^3$, the predicate $\mathcal{Q}$ corresponds to the characterization of the set $\mathcal{U}$. Namely, for all $(a,b,c) \in \mathbb{N}^3$, $\mathcal{Q}(a,b,c)$ if and only if $Q(a,b,c) = 0$.

**Interpretation on negative integers $\mathbb{Z}_-^{*3}$**

We use a diagonal and recursive method, by constructing successively:

- a simplified 0-destiny of $-1$,
- a simplified 0-destiny of $-2$,
- a simplified 1-destiny of $-1$,
- a simplified 0-destiny of $-3$,
- a simplified 1-destiny of $-2$,
- a simplified 2-destiny of $-1$,
- and so on,

and every time we need a new element to satisfy a conjunction of atomic and negatomic terms, we take the smallest (in absolute value) suitable integer.

Let us follow the first steps:

*Construction of a simplified 0-destiny of* $-1$: we set $\mathcal{Q}(-1,-1,-1)$. We have thus defined the interpretation of $\mathcal{Q}$ on $\{(-1,-1,-1)\}$.
*Construction of a simplified 0-destiny of* $-2$: we set $\neg\mathcal{Q}(-2,-2,-2)$. We have thus defined the interpretation of $\mathcal{Q}$ on $\{(-2,-2,-2)\}$.
*Construction of a simplified 1-destiny of* $-1$: we have $2^7$ possible configurations for the conjunctions of positive or negative terms with one variable and $-1$. So we need 128 elements, each fulfilling one of these conditions.

- Configuration 0000000: $-2$ is not convenient because $\neg\mathcal{Q}(-2,-2,-2)$ contradicts the first 0. So, we take $-3$, and fix:
  $\mathcal{Q}(-3,-3,-3)$, $\mathcal{Q}(-3,-3,-1)$, $\mathcal{Q}(-3,-1,-3)$, $\mathcal{Q}(-1,-3,-3)$,
  $\mathcal{Q}(-3,-1,-1)$, $\mathcal{Q}(-1,-3,-1)$, $\mathcal{Q}(-1,-1,-3)$.
- Configuration 1000000: $-2$ is convenient, so we take it, and fix:

$\neg\mathcal{Q}(-2,-2,-2)$, $\mathcal{Q}(-2,-2,-1)$, $\mathcal{Q}(-2,-1,-2)$, $\mathcal{Q}(-1,-2,-2)$, $\mathcal{Q}(-2,-1,-1)$, $\mathcal{Q}(-1,-2,-1)$, $\mathcal{Q}(-1,-1,-2)$.

- Configuration 0100000: we take $-4$, and fix: $\mathcal{Q}(-4,-4,-4)$, $\neg\mathcal{Q}(-4,-4,-1)$, $\mathcal{Q}(-4,-1,-4)$, $\mathcal{Q}(-1,-4,-4)$, $\mathcal{Q}(-4,-1,-1)$, $\mathcal{Q}(-1,-4,-1)$, $\mathcal{Q}(-1,-1,-4)$.
- and so on...

We have defined the interpretation of $\mathcal{Q}$ on the tuples containing $-1$ and elements of the set $\{-2,\ldots,-129\}$.

*Construction of a simplified 0-destiny of* $-3$: we already have fixed the interpretation of $\mathcal{Q}$ on the tuple $(-3,-3,-3)$, so this construction has already been done.

*Construction of a simplified 1-destiny of* $-2$: there are again $2^7$ possible configurations for the conjunctions of positive or negative terms with one variable and $-2$. So we need 128 elements, each fulfilling one of these conditions. It is possible to re-use some elements of the set $\{-1,-3,\ldots,-129\}$, but relationships between $-1$ and $-2$ have been already fixed, as well as relationships between the elements of the set with themselves. Every time a new element is needed, we choose the smallest (in absolute value) suitable one, and fix the corresponding interpretations of the predicate $\mathcal{Q}$.

This construction is step by step recursive, but has infinite length.
If there are tuples $(a,b,c)\in\mathbb{Z}_{-}^{*3}$ such that this construction has not defined $\mathcal{Q}$ on them, we fix $\neg\mathcal{Q}(a,b,c)$.

**Interpretation when the first element of the tuple is non-negative and belongs to $\mathcal{U}$**

We suppose that $a_0 < a_1 < \ldots < a_n < \ldots$ is the ordered list of all elements of $\mathcal{U}$. The construction is similar to the previous one, except that we add the following requirement: if $a\in\mathcal{U}$, and $u,v$ are elements of $\mathbb{Z}$ such that $||(u,v)|| < \mu(a)$, then we impose $\neg\mathcal{Q}(a,u,v)$. This case involves only a finite number of pairs $(u,v)$, so that it is always possible to find a "new" element of $\mathbb{Z}$ in order to satisfy a conjunction, if needed. We construct a simplified 0-destiny of $a_0$, then a simplified 0-destiny of $a_1$, then a simplified 1-destiny of $a_0$, and so on. If there are tuples $(a,b,c)$ with $a\in\mathcal{U}$ such that this construction has not defined $\mathcal{Q}$ on them, we fix $\neg\mathcal{Q}(a,b,c)$. This construction is not constructive.

**Interpretation in other cases**

We construct destinies of any other element as before. Suppose that $\mathbb{N}\backslash\mathcal{U} = \{b_0,\ldots b_n,\ldots\}$. We construct a simplified 0-destiny of $b_0$, then a simplified 0-destiny of $b_1$, then a simplified 1-destiny of $b_0$, and so on. If there are tuples $(a,b,c)$ with $a\notin\mathcal{U}$ such that this construction has not defined $\mathcal{Q}$ on them, we fix $\neg\mathcal{Q}(a,b,c)$.

Notice that only the construction on $\mathbb{Z}_-^{*3}$ is recursive. The predicate $\mathcal{Q}$, with this interpretation, satisfies both conditions: full destinies property, and distance property. Moreover, the construction on $\mathbb{Z}_-^{*3}$ provides an algorithm that produces a simplified $k$-destiny for every height $k$: the construction has to be stopped when a $(k-1)$-destiny of $-1$ and a $(k-1)$-destiny of $-2$ have been constructed. Then one has to set them as sons of the root $\emptyset$ in order to obtain a simplified $k$-destiny.

### *5.3 The structure is not $H$-bounded*

We have to prove that the structure $\langle \mathbb{Z}, \mathcal{Q}\rangle$ is not $H$-bounded. Assuming that there exists a recursive function $H : \mathbb{N}^3 \to \mathbb{N}$, such that the structure $\langle \mathbb{Z}, \mathcal{Q}\rangle$ is $H$-bounded, we prove that the set $\mathcal{U}$ is decidable, a contradiction. For each $a \in \mathcal{U}$, the following formula is satifiable in $\mathbb{Z}$, since destinies are full:

$$\exists x_1 \exists x_2 \mathcal{Q}(a, x_1, x_2).$$

If $\langle \mathbb{Z}, \mathcal{Q}\rangle$ is $H$-bounded, the following formula is also satisfiable in $\mathbb{Z}$:

$$(\exists x_1 \leq H(1,1,a))(\exists x_2 \leq H(0,2,H(1,1,a)))\mathcal{Q}(a, x_1, x_2).$$

It means that there exists a pair of elements of $\mathbb{Z}$, say $(x_1, x_2)$, such that $||(x_1, x_2)|| \leq H(1,1,a)^2 + H(0,2,H(1,1,a))^2$. We denote by $f(a)$ the expression $H(1,1,a)^2 + H(0,2,H(1,1,a))^2$.
By the distance property, we have $||(x_1, x_2)|| \geq \mu(a)$, so that $f(a) \geq \mu(a)$.

By definition of $\mu(a)$, it means that there exists a pair $(n_1, n_2) \in \mathbb{N}^2$ such that $Q(a, n_1, n_2) = 0$ and $\mu(a) = ||(n_1, n_2)|| \leq f(a)$. Since $H$ is recursive, so is the function $f$, and we can decide $\mathcal{U}$ by the following algorithm.

**Input:** $a \in \mathbb{N}$
**Output:** The answer to "Does $a$ belong to $\mathcal{U}$?"
(1) Compute $f(a)$
(2) For each pair $(n_1, n_2) \in \mathbb{N}^2$ such that $n_1^2 + n_2^2 \leq f(a)$
(3) If $Q(a, n_1, n_2) = 0$ Return "$a \in \mathcal{U}$".
Endfor
(4) Return "$a \notin \mathcal{U}$".

Thus the set $\mathcal{U}$ is decidable, which contradicts our first hypothesis.

The example we produced of this structure which is not $H$-bounded but has recursive destinies is ad-hoc. As pointed out by an anonymous referee, it is possible to construct another example than seems somehow more natural, because one avoids the difficult construction of the interpretation of the predicate ($Q$). This example uses the undecidability of the halting problem for Turing machines on empty inputs ([Tur37]) instead of using the undecidability of a set of integers. The (decidable) structure is $\langle \mathbb{N} \times \mathbb{N}, =, <, Halt \rangle$ where $(i,j) < (i',j')$ if and only if $i = i'$ and $j < j'$, and $Halt(i,j)$ holds if and only if the $i$-th Turing machine does not halt after $j$ steps of computation on empty input. The destinies of such a structure are not full destinies, but one may verify that they are easy to describe (they look like the destinies of a partial ordering with an additional coloration due to the $Halt$ predicate).

## 6 Strongly recursive destinies and characterization of $H$-bounded structures

We have shown that the class of structures with recursive destinies strictly contains the class of $H$-bounded structures (with $H$ recursive). However, it it possible to characterize $H$-bounded structures via destinies. For this purpose, we introduce structures with *strongly recursive destinies.*

**Definition 15 (Strongly recursive destinies)**
*Let $\mathcal{M}$ be a structure on a finite relational signature $\sigma$. The structure $\mathcal{M}$ has* **strongly recursive destinies** *if there exists an algorithm that produces simplified $k$-destinies of all tuples, for all heights $k$.*

We respectively denote by **RD, SRD** and **HB** the class of structures with recursive destinies, the class of structures with strongly recursive destinies, and the class of $H$-bounded structures. Obviously, **SRD** is included in **RD**. We prove that **SRD** and **HB** coincide, under the usual restrictions on the norm.

**Theorem 5 (SRD = HB)**
*Let $(\mathcal{M}, ||.||)$ be a normed structure on a finite relational signature $\sigma$. The structure $\mathcal{M}$ is $H$-bounded with $H$ recursive if and only if it has strongly recursive destinies.*

**Proof:** Assume that $\mathcal{M}$ is $H$-bounded, with $H$ recursive. Let $(a_1, \ldots, a_p)$ be a tuple of elements of $\mathcal{M}$. Then the structure $(\mathcal{M}, a_1, \ldots, a_p)$ is $H_{m_0,p}$-bounded, where $m_0 = \max(||a_1||, \dots ||a_p||)$, and:

$$H_{m_0,p}(n,k,m) = H(n, k+p, \max(m, m_0)).$$

The algorithm that constructs simplified $k$-destinies of each tuple is the following:

**Input:** a $p$-tuple $(a_1, \dots, a_p)$, a quantifier rank $k$
**Output:** a simplified $k$-destiny of $(a_1, \dots, a_p)$
(1) Compute $m_0 = \max(||a_1||, \dots, ||a_p||)$
(2) Construct a tree of height $k$ with "sufficiently many nodes", using as bound on the children the function $H_{m_0,p}(k-j-1, j, m)$ (where $j$ is the rank of the node)
(3) Add relations on nodes
(4) Simplify

Conversely, assume now that $\mathcal{M}$ has strongly recursive destinies. Let $m, k, n$ be integers. The set:

$$U_{k,m} = \{(a_1, \dots, a_k) \in A^k / \text{ For each } i \in \{1, \dots, k\},\ ||a_i|| \leq m\}$$

is finite, due to restrictions on the norm. Let $D_n(a_1, \dots, a_k)$ be a simplified $n$-destiny of the $k$-tuple $(a_1, \dots, a_k)$, recursively produced. We fix the function:

$$H(n,k,m) = \max_{U_{k,m}}(\mathrm{Sup}_c^{D_n(a_1,\dots,a_k)}(\emptyset)).$$

This function is clearly recursive. Let us show that $\mathcal{M}$ is $H$-bounded. Let $(a_1, \dots, a_k)$ be elements of $\mathcal{M}$ such that $||a_i|| \leq m$ for each $i \in \{1, \dots, k\}$, and $F(x_1, \dots, x_k, y)$ be a formula of quantifier rank at most $n$. We suppose that $\mathcal{M} \models \exists x_{k+1} F(a_1, \dots, a_k, x_{k+1})$. Then, by Theorem 3, $D_n(a_1, \dots, a_k) \models \exists x_{k+1}(P(\emptyset, x_{k+1}) \wedge \tilde{F}(a_1, \dots, a_k, x_{k+1}))$.

By definition of $\mathrm{Sup}_c^{D_n(a_1,\dots,a_k)}$, there exists a node $\alpha$ in $D_n(a_1, \dots, a_k)$, child of $\emptyset$, such that $||l(\alpha)|| \leq \mathrm{Sup}_c^{D_n(a_1,\dots,a_k)}(\emptyset) \leq H(n,k,m)$, and

$$D_n(a_1, \dots, a_k) \models \tilde{F}(a_1, \dots, a_k, \alpha),$$

which implies

$$\mathcal{M} \models F(a_1, \dots, a_k, l(\alpha)).$$

Consequently, there exists an element of the structure $a_{k+1} = l(\alpha)$, such that $||a_{k+1}|| \leq H(n,k,m)$ and $\mathcal{M} \models F(a_1, \dots, a_k, a_{k+1})$. This proves that $\mathcal{M}$ is $H$-bounded, with $H$ recursive.

□

Finally, let us notice that another class of structures, coinciding with the class of $H$-bounded structures, is the class of structures presented by Cegielski in [Ceg03]. It is defined below, and denoted by **CEG**.

**Definition 16 (Cegielski's structures)**
*Let $R_1, \ldots, R_k$ be predicates. The structure $\langle \mathbb{N}, R_1, \ldots, R_k \rangle$ is a* **Cegielski structure** *if the complete theory of the structure $\langle \mathbb{N}, R_1, \ldots, R_k, (i)_{i \in \mathbb{N}} \rangle$ is decidable (every element of $\mathbb{N}$ is considered as a constant),*

**Proposition 1 (CEG = SRD = HB)**
*Let $(\mathcal{M}, ||.||)$ be a normed structure on a finite relational signature $\sigma$. The structure $\mathcal{M}$ has strongly recursive destinies if and only if it is isomorphic to a Cegielski structure.*

Cegielski proves in [Ceg03] that **CEG**$\subseteq$**RD**, but this proof can easily be adapted to show that Cegielski's structures have strongly recursive destinies. Conversely, a structure with strongly recursive destinies with usual requirements on the norm is a countable structure, which can be easily converted into a structure with domain $\mathbb{N}$. Since the construction of destinies of all tuples of elements is recursive, it is clear that the complete theory of the structure $\langle \mathbb{N}, R_1, \ldots, R_k, (i)_{i \in \mathbb{N}} \rangle$ is decidable.

## 7 Conclusion and further developments

We have presented an original decision algorithm using destinies, and defined the class of structures with recursive destinies. We have shown that, under some natural conditions, the class of $H$-bounded structures is equivalent to the class of structures with strongly recursive destinies. Some additional questions remain open. Is it possible to characterize other classes of decidable structures with destinies? In all likelihood, there are some decidable structures that do not have recursive destinies. Now, is it possible to describe those structures? Does the form of destinies (lacunarity, fullness, and so on) have an influence on the "decidability degree" of the structure?

Another way to extend this work is to look at second-order theories. Indeed, we defined destinies only in the case of first-order logic, but one can easily imagine to extend the definition to second-order logic. Then, we have to notice that destinies are very close to the notion of $\overline{k}$-theories of structures, introduced by Läuchli ([Läu68]) as an extension of Ehrenfeucht games and Fraïssé $k$-equivalence to monadic logics. The $\overline{k}$-theories of structures were used by Shelah ([She75]) and Gurevich ([Gur85]) to study decidability of monadic second-order theories. A clear survey of these works appear in [Tho97], where $\overline{k}$-theories are used to study the decidability of the monadic theory of labelled ordering. A sequence $\overline{k} = (k_1, \ldots, k_n)$ of natural numbers determines the prenex quantification of sentences as follows: a block of $k_1$ quantifiers of the same kind, then $k_2$ quantifiers of the same kind and so on. The $\overline{k}$-theory of a structure describes the behaviour of prenex sentences of this form, and can be seen as a finite tree. The differences between $k$-destinies and $\overline{k}$-theories are the following:

- $k$-destinies give witnesses of $k$-equivalence classes, instead of formal $k$-types. In Section 5.1, Figures 7 and 8 present "formal" destinies, but we can see that the knowledge of their form is not sufficient to exhibit a simplified $k$-destiny, as done in the construction of Section 5.2. In the case of $\overline{k}$-theories, only formal $\overline{k}$-types are studied, which leads to the uniqueness of the $\overline{k}$-theory of a structure. So we can say that $k$-destinies are "closer" to the structures, but that it creates more difficulties to link computability of destinies and decidability of the structures.
- The $\overline{k}$-theory definition is based on prenex formulas, whereas prenex formulas are not a relevant tool to construct $k$-destinies.
- The notion of alternation of quantifiers is preferred to the notion of quantifier depth in the case of $\overline{k}$-theories, due to technical point in the proof of the Composition Theorem ([She75], [Gur79]).

So, an interesting perspective could be a further comparison between both notions.

As we can see, the alternative perspective on decidable structures brought by destinies raises new questions, and hopefully new answers.


## Acknowledgements

I would like to thank Malika More [2] for her careful reading and her very useful advices.

I would like to underline the fact that this article was partially supported by Intas 2000 - 447.


## References


[Bès02] A. Bès. A survey of arithmetical definability. *A tribute to Maurice Boffa, Special Issue of Belgian Mathematical Society*, pages 1–54, 2002.


[2] LLAIC1, Clermont-Ferrand

[BGG97] E. Börger, E. Grädel, and Y. Gurevich. *The classical decision problem.* Springer-Verlag, 1997.

[Ceg03] P. Cegielski. Decidability and p-destinies. *Zapiski Nauchnyh Seminarov POMI, to appear in Journal of Mathematical Sciences*, 304, 2003.

[Cha03] A. Chateau. *Utilisation des destinées pour la décision et sa complexité dans le cas de formules à profondeur de quantification bornée sur des structures finies ou infinies.* PhD thesis, Université d'Auvergne, 2003.

[Dub95] D. P. Dubhashi. Complexity of Logical Theories. Technical report, BRICS (Basic Research In Computer Science), Departement of Computer Science, University of Aarhus, Denmark, 1995.

[EF95] H.-D. Ebbinghaus and J. Flum. *Finite Model Theory.* Springer-Verlag, 1995.

[EFT94] H.-D. Ebbinghaus, J. Flum, and W. Thomas. *Mathematical Logic.* Springer-Verlag, 1994.

[FR79] J. Ferrante and C. W. Rackoff. The Computational Complexity of Logical Theories. In *Lecture Notes in Mathematics*, volume 718. Springer-Verlag, 1979.

[Gur79] Y. Gurevich. Modest theories of short chains I. *Journal of Symbolic Logic*, 44:481–490, 1979.

[Gur85] Y. Gurevich. Monadic second-order theories. In J. Barwise and S. Feferman, editors, *Model-Theoretic Logics*, pages 479–506. Springer-Verlag, 1985.

[HB34] D. Hilbert and P. Bernays. *Grundlagen der Mathematik.* Springer-Verlag, 1934.

[Läu68] H. Läuchli. A decision procedure for the weak second order theory of linear order. *Contributions to Mathematical Logic*, 1968.

[Mat93] Y. Matiyasevich. *Hilbert's tenth problem.* The MIT press, 1993.

[Mau94] F. Maurin. *Complexité de l'addition ordinale.* PhD thesis, Université de Caen, 1994.

[Mic81] P. Michel. Borne supérieure de la complexité de la théorie de $\mathbb{N}$ muni de la relation de divisibilité. In *Lecture Notes in Mathematics*, volume 890. Springer-Verlag, 1981.

[Mos52] A. Mostowski. On direct products of theories. *Journal of Symbolic Logic*, 17:1–31, 1952.

[Néz97] F. Nézondet. *p-destinées et applications à la théorie du successeur et de la coprimarité sur les entiers.* PhD thesis, Université d'Auvergne, 1997.

[Pre29] M Presburger. Über die Vollständigkeit eines gewissen Systems der Arithmetik ganzer Zahlen, in welchem die Addition als einzige Operation hervortritt. *C.R. 1er congrès des Mathématiciens des pays slaves*, pages 92–101, 1929.

[She75] S. Shelah. The monadic theory of order. *Annals of Mathematics*, 102:379–419, 1975.

[Tho97] W. Thomas. Ehrenfeucht games, the composition method, and the monadic theory of ordinal words. In *Structures in Logic and Computer Science*, pages 118–143, 1997.

[Tur37] A. M. Turing. On Computable Numbers, with an Application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society*, 42:230–265, 1937.

## A Proof of Theorem 1

**Proof:**(by induction on $k$)
**Case $k = 0$:** let $F(y_1, \ldots, y_n)$ be a quantifier-free $\sigma$-formula. In this case, we have $\bar{F}(y_1, \ldots, y_n) = F(y_1, \ldots, y_n)$, so that the result is clear.

**Let us show that the induction hypothesis at rank $k$ implies the induction hypothesis at rank $k + 1$.**
Let $F(y_1, \ldots, y_n)$ be a $\sigma$-formula of quantifier rank $k + 1$. Let $(a_1, \ldots, a_n)$ be a sequence of elements of $\mathcal{M}$. Without any loss of generality, we assume that $F$ is of the form $Q_1 x_1 G(y_1, \ldots, y_n, x_1)$, with $G$ a $\sigma$-formula of quantifier rank $k$.

If $Q_1 = \exists$ and $\mathcal{M} \models F(a_1, \ldots, a_n)$, then, since the structure $\mathcal{M}$ is $H$-bounded, we have $\mathcal{M} \models (\exists x_1 \leq H(k, n, m_0))G(a_1, \ldots, a_n, x_1)$, thus

$$\mathcal{M} \models (\exists x_1 \leq m_1)G(a_1, \ldots, a_n, x_1).$$

It means that there exists an element $a_{n+1}$ of $\mathcal{M}$, such that $||a_{n+1}|| \leq m_1$ and $\mathcal{M} \models G(a_1, \ldots, a_n, a_{n+1})$. Let us apply the induction hypothesis to $G$ (the sequence $m_1, \ldots, m_k$ is convenient), which gives us:

$$\mathcal{M} \models \bar{G}_{(m_2,\ldots,m_k)}(a_1, \ldots, a_n, a_{n+1}).$$

Since $||a_{n+1}|| \leq m_1$: $\mathcal{M} \models (\exists x_1 \leq m_1)\bar{G}_{(m_2,\ldots,m_k)}(a_1, \ldots, a_n, x_1)$, we have:

$$\mathcal{M} \models \bar{F}_{(m_1,\ldots,m_k)}(a_1, \ldots, a_n).$$

Conversely, if $\mathcal{M} \models \bar{F}_{(m_1,\ldots,m_k)}(a_1,\ldots,a_n)$, then we have:

$$\mathcal{M} \models (\exists x_1 \leq m_1)\bar{G}_{(m_2,\ldots,m_k)}(a_1,\ldots,a_n,x_1),$$

which implies the existence of an element $a_{n+1}$ of $\mathcal{M}$, such that $||a_{n+1}|| \leq m_1$ and $\mathcal{M} \models \bar{G}_{(m_2,\ldots,m_k)}(a_1,\ldots,a_n,a_{n+1})$. Let us apply the induction hypothesis:

$$\mathcal{M} \models G(a_1,\ldots,a_n,a_{n+1}).$$

Then $\mathcal{M} \models (\exists x_1 \leq m_1)G(a_1,\ldots,a_n,x_1)$, which clearly implies:

$$\mathcal{M} \models \exists x_1 G(a_1,\ldots,a_n,x_1),$$

which can also be written:

$$\mathcal{M} \models F(a_1,\ldots,a_n).$$

In the case where $Q_1 = \forall$, we consider $\neg F$ and it runs as before.

□

## B Proof of Theorem 3

This theorem is a straightforward consequence of the following proposition. Let $\sigma$ be a finite relational signature, $\mathcal{M}$ be a $\sigma$-structure and $T$ be a simplified $k$-destiny of $\mathcal{M}$. Let $F(x_1,\ldots,x_n)$ be a formula of quantifier rank $k-n$. Let $(a_1,\ldots,a_n)$ be a $n$-tuple of elements of $\mathcal{M}$. There exists a node $b_n$ with ancestors $b_{n-1},\ldots,b_1,\emptyset$ in the complete $k$-destiny of $\mathcal{M}$ such that for each $i \in \{1,\ldots,n\}$, we have $l(b_i) = a_i$. The subtree $ST(b_n)$ has an isomorphic subtree in any simplified destiny, by construction of a simplified destiny.

**Proposition 2**
*For each node $b'_n$ of a simplified destiny $T$ of $\mathcal{M}$, with ancestors $b'_{n-1},\ldots,b_1,\emptyset$, such that $ST(b'_n)$ is isomorphic to $ST(b_n)$, we have $\mathcal{M} \models F(a_1,\ldots,a_n)$ if and only if $T \models \tilde{F}(b'_1,\ldots,b'_n)$.*

**Proof:** We prove the proposition by decreasing induction on $n \in \{1,\ldots,k\}$.
$\boxed{\Rightarrow}$ (Case $\Leftarrow$ is very similar, so we do not detail it)

**Case** $n = k$**:** let $F(x_1,\ldots,x_n)$ be a quantifier free formula. We have $\tilde{F} = F$. Let $(a_1,\ldots,a_k)$ be a $k$-tuple of elements of $\mathcal{M}$, and $b_k$ be a node in the complete $k$-destiny of $\mathcal{M}$, with ancestors $b_{k-1},\ldots,b_1,\emptyset$ such that, for each $i \in \{1,\ldots k\}$, we have $l(b_i) = a_i$. Let $b'_k$ be a node of $T$, with ancestors $b'_{k-1},\ldots,b'_1,\emptyset$, such that $ST(b'_k)$ and $ST(b_k)$ are isomorphic (such a node exists since the destiny $T$ is simplified). By the definition of isomorphism between leaves, the $k$-tuples $(l(b_1), \ldots, l(b_k))$ and $(l(b'_1), \ldots, l(b'_k))$ satisfy the same quantifier free formulas. Thus, $\mathcal{M} \models F(l(b'_1), \ldots, l(b'_p))$. By definition of a destiny, we have $T \models F(b'_1, \ldots, b'_k)$, then $T \models \tilde{F}(b'_1, \ldots, b'_k)$.

**Let us show that the induction hypothesis at rank $1 \leq n+1 \leq k$ implies the induction hypothesis at rank $n$.**
Let $F(x_1, \ldots, x_n)$ be a formula of quantifier rank $k-n$. Let $(a_1, \ldots, a_n)$ be a $n$-tuple of elements of $\mathcal{M}$. Let $b_n$ and $b'_n$ be nodes as above. Without any loss of generality, we suppose that $F(x_1, \ldots, x_n) = Q_{n+1}x_{n+1}G(x_1, \ldots, x_{n+1})$.

- If $Q_{n+1} = \exists$: then $\tilde{F}(x_1, \ldots, x_n) = \exists x_{n+1}(P(x_n, x_{n+1}) \wedge \tilde{G}(x_1, \ldots, x_{n+1}))$. Assuming that $\mathcal{M} \models F(a_1, \ldots, a_n)$, then $\mathcal{M} \models \exists x_{n+1}G(a_1, \ldots, a_n, x_{n+1})$, so that there exists an element $a_{n+1}$ of $\mathcal{M}$ such that $\mathcal{M} \models G(a_1, \ldots, a_n, a_{n+1})$. Since $ST(b'_n)$ and $ST(b_n)$ are isomorphic, there exists a child $b'_{n+1}$ of $b'_n$, such that $ST(b'_{n+1})$ and $ST(b_{n+1})$ are isomorphic, where $b_{n+1}$ is the child of $b_n$ with label $a_{n+1}$ in the complete $k$-destiny of $\mathcal{M}$. We apply the induction hypothesis to $G$: $T \models \tilde{G}(b'_1, \ldots, b'_{n+1})$. Since $b'_{n+1}$ is a child of $b'_n$, we have

$$T \models P(b'_n, b'_{n+1}) \wedge \tilde{G}(b'_1, \ldots, b'_{n+1}),$$

  which implies

$$T \models \exists x_{n+1}(P(b'_n, x_{n+1}) \wedge \tilde{G}(b'_1, \ldots, b'_n, x_{n+1})),$$

  thus

$$T \models \tilde{F}(b'_1, \ldots, b'_n).$$

- If $Q_{n+1} = \forall$: then $\tilde{F}(x_1, \ldots, x_n) = \forall x_{n+1}(P(x_n, x_{n+1}) \rightarrow \tilde{G}(x_1, \ldots, x_{n+1}))$. Assuming that $\mathcal{M} \models F(a_1, \ldots, a_n)$, then $\mathcal{M} \models \forall x_{n+1}G(a_1, \ldots, a_n, x_{n+1})$, so that for each element $a_{n+1}$ of $\mathcal{M}$, we have $\mathcal{M} \models G(a_1, \ldots, a_n, a_{n+1})$. For each child $b'_{n+1}$ of $b'_n$, there exists a child $b_{n+1}$ of $b_n$ in the complete $k$-destiny of $\mathcal{M}$, such that $ST(b'_{n+1})$ and $ST(b_{n+1})$ are isomorphic. We have $\mathcal{M} \models G(a_1, \ldots, a_n, l(b_{n+1}))$. We apply the induction hypothesis to $G$: $T \models \tilde{G}(b'_1, \ldots, b'_{n+1})$. This holds for each child $b'_{n+1}$ of $b'_n$, so that

$$T \models \forall x_{n+1}(P(b'_n, x_{n+1}) \rightarrow \tilde{G}(b'_1, \ldots, b'_n, x_{n+1})),$$

  thus

$$T \models \tilde{F}(b'_1, \ldots, b'_n).$$

$\square$

## C Proof of Theorem 4

We use Proposition 3, which characterizes a class of isomorphism between subtrees in a destiny by a formula (which is similar to Hintikka's formula).

**Proposition 3**
*Let us fix $k \geq 1$. Let $\mathcal{M}$ and $\mathcal{M}'$ be two $\sigma$-structures on a finite relational signature, and $T$ and $T'$ be two simplified $k$-destinies, respectively of $\mathcal{M}$ and $\mathcal{M}'$. For each node $\alpha_p$ of rank $p \leq k$, with ancestors $\alpha_{p-1}, \ldots, \alpha_1, \emptyset$ in $T$, there exists a $\sigma$-formula $F_{(\alpha_1,\ldots,\alpha_p)}(x_1, \ldots, x_p)$ of quantifier rank $k-p$, such that for each node $\beta_p$ with ancestors $\beta_{p-1}, \ldots, \beta_1, \emptyset$ in $T'$, the subtrees $ST(\beta_p)$ and $ST(\alpha_p)$ are isomorphic if and only if $T' \models \tilde{F}_{(\alpha_1,\ldots,\alpha_p)}(\beta_1, \ldots, \beta_p)$.*

**Proof:** We prove the proposition by decreasing induction on $p \in \{1, \ldots, k\}$.

**Case $p = k$:** let $\alpha_k$ be a leaf in $T$, with ancestors $\alpha_{k-1}, \ldots, \alpha_1, \emptyset$. The formula $F_{(\alpha_1,\ldots,\alpha_k)}(x_1, \ldots, x_k)$ denotes the conjunction of all atomic and negatomic $\sigma$-formulas which are satisfied by the $k$-tuple $(l(\alpha_1), \ldots, l(\alpha_k))$ in the structure $\mathcal{M}$. Let us show that it is convenient: let $\beta_k$ be a leaf in $T'$, with ancestors $\beta_{k-1}, \ldots, \beta_1, \emptyset$.

- If $\beta_k$ and $\alpha_k$ are isomorphic, then the tuples $(\alpha_1, \ldots, \alpha_k)$ and $(\beta_1, \ldots, \beta_k)$ satisfy the same quantifier free formulas.
  Since $\mathcal{M} \models F_{(\alpha_1,\ldots,\alpha_k)}(l(\alpha_1), \ldots, l(\alpha_k))$, then $T \models \tilde{F}_{(\alpha_1,\ldots,\alpha_k)}(\alpha_1, \ldots, \alpha_k)$, which implies $T' \models \tilde{F}_{(\alpha_1,\ldots,\alpha_k)}(\beta_1, \ldots, \beta_k)$.
- If $T' \models \tilde{F}_{(\alpha_1,\ldots,\alpha_k)}(\beta_1, \ldots, \beta_k)$, then the tuples $(\alpha_1, \ldots, \alpha_k)$ and $(\beta_1, \ldots, \beta_k)$ satisfy the same quantifier free $\sigma$-formulas. Since they also satisfy the same quantifier free formulas involving the predicate $P$ and the constant $\emptyset$, they satisfy the same quantifier free $\sigma'$-formulas. Thus $\alpha_k$ and $\beta_k$ are isomorphic.

**Let us show that the induction hypothesis at rank $1 \leq p+1 \leq k$ implies the induction hypothesis at rank $p$.**
Let $\alpha_p$ be a node in $T$ with ancestors $\alpha_{p-1}, \ldots, \alpha_1, \emptyset$. The node $\alpha_p$ has a finite number of children with non-isomorphic subtrees.
Let $y_1, \ldots, y_{n_p}$ be children of $\alpha_p$ such that the subtrees $ST(y_i)$ are non-isomorphic and meet all the classes of isomorphism between subtrees with root being a child of $\alpha_p$.

By induction hypothesis, there exists for each $i \in \{1, \ldots, n_p\}$ a formula, denoted by $F_{(\alpha_1,\ldots,\alpha_p,y_i)}(x_1, \ldots, x_p, x_{p+1})$, of quantifier rank $k-p-1$, which characterizes the class of isomorphism of the subtree $ST(y_i)$.

Let $G(x_1, \ldots, x_p)$ be the conjunction of all atomic and negatomic formulas satisfied by the tuple $(l(\alpha_1), \ldots, l(\alpha_p))$ in $\mathcal{M}$.

We fix the formula $F_{(\alpha_1,\ldots,\alpha_p)}(x_1,\ldots,x_p)$:

$$G(x_1,\ldots,x_p) \wedge \bigwedge_{i=1}^{n_p} \exists x_{p+1} F_{(\alpha_1,\ldots,\alpha_p,y_i)}(x_1,\ldots,x_{p+1})$$
$$\wedge \forall x_{p+1} \bigvee_{i=1}^{n_p} F_{(\alpha_1,\ldots,\alpha_p,y_i)}(x_1,\ldots,x_{p+1}).$$

It is a formula of quantifier rank $k-p$. Let us show that it is convenient. Let $\beta_p$ a node in $T'$ with ancestors $\beta_{p-1},\ldots,\beta_1,\emptyset$.

- If $ST(\beta_p)$ and $ST(\alpha_p)$ are isomorphic, then:
  - the tuples $(\alpha_1,\ldots,\alpha_p)$ and $(\beta_1,\ldots,\beta_p)$ satisfy the same quantifier free formulas on $\sigma'$. since $\mathcal{M} \models G(l(\alpha_1),\ldots,l(\alpha_p))$, we have

    $$T \models \tilde{G}(\alpha_1,\ldots,\alpha_p),$$

    then, by isomorphism,

    $$T' \models \tilde{G}(\beta_1,\ldots,\beta_p).$$
  - For each child $y$ of $\alpha_p$, there exists a child $z$ of $\beta_p$ , such that $ST(z)$ and $ST(y)$ are isomorphic. Let $i \in \{1,\ldots,n_p\}$ be the integer such that $ST(y)$ and $ST(y_i)$ are isomorphic. By induction hypothesis,

    $$T' \models \tilde{F}_{(\alpha_1,\ldots,\alpha_p,y_i)}(\beta_1,\ldots,\beta_p,z).$$

    Then, for each $i \in \{1,\ldots,n_p\}$,

    $$T' \models \exists x_{p+1}(P(\beta_p,x_{p+1}) \wedge \tilde{F}_{(\alpha_1,\ldots,\alpha_p,y_i)}(\beta_1,\ldots,\beta_p,x_{p+1})),$$

    therefore

    $$T' \models \bigwedge_{i=1}^{n_p} \exists x_{p+1}(P(\beta_p,x_{p+1}) \wedge \tilde{F}_{(\alpha_1,\ldots,\alpha_p,y_i)}(\beta_1,\ldots,\beta_p,x_{p+1})).$$
  - For each child $z$ of $\beta_p$, there exists a child $y$ of $\beta_p$ , such that $ST(z)$ and $ST(y)$ are isomorphic. Let $i \in \{1,\ldots,n_p\}$ be the integer such that $ST(y)$ and $ST(y_i)$ are isomorphic. By induction hypothesis,

    $$T' \models \tilde{F}_{(\alpha_1,\ldots,\alpha_p,y_i)}(\beta_1,\ldots,\beta_p,z).$$

    Then for each child $z$ of $\beta_p$, there exists an integer $i \in \{1,\ldots,n_p\}$ such that

    $$T' \models \tilde{F}_{(\alpha_1,\ldots,\alpha_p,y_i)}(\beta_1,\ldots,\beta_p,z),$$

    which implies

    $$T' \models \forall x_{p+1}(P(\beta_p,x_{p+1}) \rightarrow \bigvee_{i=1}^{n_p} \tilde{F}_{(\alpha_1,\ldots,\alpha_p,y_i)}(\beta_1,\ldots,\beta_p,x_{p+1})).$$

Finally:

$$
\begin{aligned}
T' \models & \tilde{G}(\alpha_1, \ldots, \alpha_p) \\
& \wedge \forall x_{p+1}(P(\beta_p, x_{p+1}) \rightarrow \bigvee_{i=1}^{n_p} \tilde{F}_{(\alpha_1,\ldots,\alpha_p,y_i)}(\beta_1, \ldots, \beta_p, x_{p+1})) \\
& \wedge \bigwedge_{i=1}^{n_p} \exists x_{p+1}(P(\beta_p, x_{p+1}) \wedge \tilde{F}_{(\alpha_1,\ldots,\alpha_p,y_i)}(\beta_1, \ldots, \beta_p, x_{p+1})),
\end{aligned}
$$

*i.e.*

$$T' \models \tilde{F}_{(\alpha_1,\ldots,\alpha_p)}(\beta_1, \ldots, \beta_p).$$

- If $T' \models \tilde{F}_{(\alpha_1,\ldots,\alpha_p)}(\beta_1, \ldots, \beta_p)$:
  - · Since $T' \models \tilde{G}(\beta_1, \ldots, \beta_p)$, then the tuples $(\alpha_1, \ldots, \alpha_p)$ and $(\beta_1, \ldots, \beta_p)$ satisfy the same quantifier free formulas on $\sigma$. Since they also satisfy the same quantifier free formulas involving the predicate $P$ and the constant $\emptyset$, they satisfy the same quantifier free formulas on $\sigma'$.
  - · Since $T' \models \bigwedge_{i=1}^{n_p} \exists x_{p+1}(P(\beta_p, x_{p+1}) \wedge \tilde{F}_{(\alpha_1,\ldots,\alpha_p,y_i)}(\beta_1, \ldots, \beta_p, x_{p+1}))$, then for each $i \in \{1, \ldots, n_p\}$, there exists a child $z$ of $\beta_p$ such that $T' \models \tilde{F}_{(\alpha_1,\ldots,\alpha_p,y_i)}(\beta_1, \ldots, \beta_p, z)$. By induction hypothesis, it means that $ST(z)$ and $ST(y_i)$ are isomorphic. For each child $y$ of $\alpha_p$, there exist an integer $i \in \{1, \ldots, n_p\}$ such that $ST(y)$ and $ST(y_i)$ are isomorphic. Then, for each child $y$ of $\alpha_p$, there exists a child $z$ of $\beta_p$ such that $ST(y)$ and $ST(z)$ are isomorphic.
  - · Since $T' \models \forall x_{p+1}(P(\beta_p, x_{p+1}) \rightarrow \bigvee_{i=1}^{n_p} \tilde{F}_{(\alpha_1,\ldots,\alpha_p,y_i)}(\beta_1, \ldots, \beta_p, x_{p+1}))$, then for each child $z$ of $\beta_p$, there exists an integer $i \in \{1, \ldots, n_p\}$ such that $T' \models \tilde{F}_{(\alpha_1,\ldots,\alpha_p,y_i)}(\beta_1, \ldots, \beta_p, z)$. By induction hypothesis, it means that $ST(z)$ and $ST(y_i)$ are isomorphic. Then, for each child $z$ of $\beta_p$, there exists a child $y$ of $\alpha_p$ such that $ST(y)$ and $ST(z)$ are isomorphic.

  Consequently, $ST(\alpha_p)$ and $ST(\beta_p)$ are isomorphic.

□

**Proof:** (of Theorem 4) We use Proposition 3.
Let $\alpha_p$ be a node of rank $p \in \{0, \ldots, k-1\}$ with ancestors $\alpha_{p-1}, \ldots, \alpha_1, \emptyset$ in the reduced destiny $T$. We note $m = \max(||l(\alpha_1)||, \ldots, ||l(\alpha_p)||)$ ($m = 0$ if $\alpha_p$ is the root $\emptyset$). Each child $\alpha_{p+1}$ of $\alpha_p$ is the root of a unique representative of a class of isomorphism between subtrees. Each of them is characterized by a formula $F_{(\alpha_1,\ldots,\alpha_{p+1})}(x_1, \ldots, x_{p+1})$ of quantifier rank $k-p-1$ (by Proposition 3). Fixing $\alpha_{p+1}$, we apply the definition of a $H$-bounded structure to the formula $\exists y F_{(\alpha_1,\ldots,\alpha_{p+1})}(\alpha_1, \ldots, \alpha_p, y)$. We know that

$$T \models \exists y(P(\alpha_p, y) \wedge \tilde{F}_{(\alpha_1,\ldots,\alpha_{p+1})}(\alpha_1, \ldots, \alpha_p, y))$$

thus

$$\mathcal{M} \models \exists y F_{(\alpha_1,\ldots,\alpha_{p+1})}(l(\alpha_1), \ldots, l(\alpha_p), y).$$

Then, by definition of a $H$-bounded structure,

$$\mathcal{M} \models (\exists y \leq H(k-p-1,p,m))F_{(\alpha_1,\ldots,\alpha_{p+1})}(l(\alpha_1),\ldots,l(\alpha_p),y).$$

Since $T$ is a reduced destiny, and the node $\alpha_{p+1}$ has minimal label, we have: $||l(\alpha_{p+1})|| \leq H(k-p-1,p,m)$. This holds for each child of $\alpha_p$, so that

$$\mathrm{Sup}_c^T(\alpha_p) \leq H(k-p-1,p,m).$$

$\square$