GUARDED SECOND-ORDER LOGIC, SPANNING TREES, AND NETWORK FLOWS

ACHIM BLUMENSATH

TU Darmstadt, Germany
e-mail address: blumensath@mathematik.tu-darmstadt.de

ABSTRACT. According to a theorem of Courcelle monadic second-order logic and guarded second-order logic (where one can also quantify over sets of edges) have the same expressive power over the class of all countable k-sparse hypergraphs. In the first part of the present paper we extend this result to hypergraphs of arbitrary cardinality. In the second part, we present a generalisation dealing with methods to encode sets of vertices by single vertices.

INTRODUCTION

Guarded second-order logic (GSO) is the variant of monadic second-order logic (MSO) where one can not only quantify over sets of vertices but also over sets of edges. This modification results in a large increase of expressive power. Statements that can be expressed in guarded second-order logic, but not in monadic second-order logic, include the existence of certain minors in a graph and the existence of Hamiltonian paths.

The high expressive power of guarded second-order logic means that most GSO-theories are quite complicated. In [7] Seese has shown that every class of graphs with infinite tree width has an undecidable GSO-theory. This result immediately generalises to hypergraphs. It follows that all classes of hypergraphs with a decidable GSO-theory are *k-sparse,* for some k, which roughly means that their members have few edges. For classes of countable k-sparse hypergraphs, Courcelle [2] has shown that every GSO-formula is equivalent to an MSO-formula over such a class. It follows that over every class of countable hypergraphs with a decidable GSO-theory guarded second-order logic and monadic second-order logic have the same expressive power. Unfortunately, the proof of Theorem 1.4 in [2] contains an error. In the first part of the present article we give a new proof of this theorem. In addition, we extend the result from countable hypergraphs to hypergraphs of arbitrary cardinality.

When we look at the results of the first part we see that most of them concern the coding of sets of vertices by single vertices. In the abstract, this problem can be stated as follows: given a set $F \subseteq \mathscr{P}(V)$ of finite sets of vertices, find a definable function $f : F \to V$ that is injective. In our concrete case, $F := E$ is the set of edges. In the second part of the paper we consider more general instances of this problem where F can be arbitrary. This generalisation is inspired by a result of Colcombet and Löding [1] on set interpretations. Their

main technical result is a method to transform a definable finite-to-one function $F \to V$ into an injective one. Colcombet and Lödung consider as background structure only the infinite binary tree. Below we show that using guarded second-order parameters, i.e., sets of edges, we can extend some of their results to arbitrary graphs.

The overview of the article is a follows. We start in Section 1 with basic definitions and a survey of results on definable orientations of sparse hypergraphs. In Section 2 we prove the general version of the one technical result of [2] whose proof does not extend to arbitrary cardinalities. In Section 3 we summarise the consequences for the expressive power of guarded second-order logic on sparse hypergraphs.

Section 4 contains the second part of the article. We study network flow problems and we show how to use flows to transform definable finite-to-one maps into injective ones.

1. ORIENTATIONS OF SPARSE HYPERGRAPHS

Let us fix our terminology regarding graphs and hypergraphs. When we say 'graph' we will mean an undirected one. Undirected graphs will always be simple and loop free, whereas directed graphs will be simple, but they may contain loops. When dealing with hypergraphs we will sometimes allow multiple edges. Such a hypergraph is a two-sorted structure $\langle V, E, I \rangle$ where V is the set of vertices, E the set of edges, and $I \subseteq V \times E$ the incidence relation. Using sloppy notation we will tacitly identify an edge $e \in E$ of such a hypergraph with the set $\{ v \in V \mid \langle v, e \rangle \in I \}$ of its vertices and we write $v \in e$ instead of $\langle v, e \rangle \in I$. Similarly, if $F \subseteq E$ is a set of edges then the union $\bigcup F$ consists of all vertices incident with at least one edge of F. We will use this notation even if there are multiple edges.

Monadic second-order logic (MSO) extends first-order logic by variables and quantifiers that range over sets of vertices. Similarly, *guarded second-order logic* (GSO) extends first-order logic by variables and quantifiers ranging over sets of vertices or sets of edges (for detailed definitions see [4]). We will also consider *weak monadic second-order logic* (WMSO) where quantification is restricted to *finite* sets of vertices.

Definition 1.1. Let $\mathfrak{H} = \langle V, E \rangle$ be a hypergraph.

(a) We say that $\mathfrak{H}$ has *rank* m if every edge of $\mathfrak{H}$ has at most m vertices.
(b) A *subhypergraph* of $\mathfrak{H}$ is a hypergraph $\mathfrak{H}_0 = \langle V_0, E_0 \rangle$ with $V_0 \subseteq V$ and $E_0 \subseteq E$.
(c) Let $C \subseteq V$. The subhypergraph *induced by* C is

$$\mathfrak{H}|_C := \langle C, E|_C \rangle \quad \text{with} \quad E|_C := \{ e \in E \mid e \subseteq C \} .$$

In order to translate GSO-formulae into MSO-formulae, we have to encode sets of edges by sets of vertices. A simple way to do so consists in choosing an *orientation* of the hypergraph, i.e., a function assigning to each edge one of its vertices.

Definition 1.2. Let $\mathfrak{H} = \langle V, E \rangle$ be a hypergraph.

(a) An *orientation*[1] of $\mathfrak{H}$ is a function $f : E \to V$ with $f(e) \in e$, for all $e \in E$. We say that a formula $\varphi(x, Y)$ *defines* an orientation f of $\mathfrak{H}$ if we have

$$\mathfrak{H} \models \varphi(a, e) \quad \text{iff} \quad f(e) = a, \quad \text{for all } a \in A \text{ and } e \in E .$$

(b) An orientation f is *bounded by* k if

$$|f^{-1}(a)| \leq k, \quad \text{for all } a \in A .$$

---

[1]This is called a *semi-orientation* in [2].

(c) We call $\mathfrak{H}$ MSO-*orientable* if there exist an MSO-formula $\varphi(x, Y; \bar{P})$ with parameters $P_i \subseteq V$ defining an orientation of $\mathfrak{H}$. Similarly, we say that $\mathfrak{H}$ is GSO-*orientable* if there exist a GSO-formula $\varphi(x, Y; \bar{P}, \bar{Q})$ with parameters $P_i \subseteq V$ and $Q_i \subseteq E$ defining an orientation of $\mathfrak{H}$.

In this paper we are mainly interested in *sparse* hypergraphs, i.e., hypergraphs with few edges.

Definition 1.3. A hypergraph $\mathfrak{H} = \langle V, E \rangle$ is *k-sparse*[2] if
$$\big| E|_X \big| \leq k \cdot |X|, \quad \text{for every finite set } X \subseteq V.$$

Lemma 1.4. *Let* $\mathfrak{G} = \langle V, E \rangle$ *be a graph.*

(a) *If the degree of* $\mathfrak{G}$ *is at most* $2k$, *then* $\mathfrak{G}$ *is k-sparse.*

(b) *If* $\mathfrak{G}$ *is planar, then it is 3-sparse.*

Proof.

(a) If $\mathfrak{X} = \langle X, F \rangle$ is a finite induced subgraph of $\mathfrak{G}$ then
$$2 \cdot |F| = \sum_{v \in X} \deg(v) \leq 2k \cdot |X|.$$

(b) This follows from the fact that every planar graph with n vertices has at most $3n - 6$ edges (see, e.g., Corollary 4.2.10 of [3]). $\square$

In the next section we will prove that every hypergraph of bounded rank is GSO-orientable. In the remainder of this section we show that k-sparse hypergraphs are even MSO-orientable. For countable hypergraphs these results are all due to Courcelle [2]. The only thing new in the present section are two applications of the compactness theorem for first-order logic to extend the results to uncountable hypergraphs. The proofs in Section 2, on the other hand, are mostly new.

Lemma 1.5. *A hypergraph* $\mathfrak{H} = \langle V, E \rangle$ *(possibly with multiple edges) of finite rank is k-sparse if and only if there exists an orientation of* $\mathfrak{H}$ *that is bounded by k.*

Proof. For ($\Leftarrow$), let $X \subseteq V$ be finite. Then
$$\big| E|_X \big| \leq \sum_{a \in X} |f^{-1}(a)| \leq k \cdot |X|.$$

($\Rightarrow$) First, let us consider the case where $\mathfrak{H}$ is finite. If f is an arbitrary orientation of $\mathfrak{H}$ then
$$\sum_{a \in V} |f^{-1}(a)| = |E| \leq k \cdot |V|.$$

Hence, if there is some element $a \in V$ with $|f^{-1}(a)| > k$ then there must be some other element $b \in V$ with $|f^{-1}(b)| < k$. Let us define the *weight* of an orientation by
$$w(f) := \sum \big\{ |f^{-1}(a)| - k \mid a \in V, \ |f^{-1}(a)| > k \big\}.$$

We have to construct an orientation of weight 0. To do so we transform an orientation f with $w(f) > 0$ into one with smaller weight. Given f, fix an element $a \in V$ with $|f^{-1}(a)| > k$. Let $F \subseteq E$ be the smallest subset of E such that a belongs to the set $U := \bigcup F$ and we have

---

[2]In [2] such hypergraphs are called *uniformly k-sparse*. Courcelle also introduces a notion of a *k-sparse* graph. Since uniform sparsity is the more robust notion, and the only one we will use in this paper, we have changed terminology for brevity. A related notion is the *arboricity* of a graph (see, e.g., Section 2.4 of [3]).

$f^{-1}(c) \subseteq F$, for every element $c \in U$. The subhypergraph $\mathfrak{H}|_U$ induced by U is k-sparse. Hence, there exists some element $b \in U$ with $|f^{-1}(b)| < k$. By choice of F we can find a sequence of edges $e_0, \ldots, e_n \in F$ with

$$b \in e_0, \quad f(e_i) \in e_{i+1}, \quad \text{and} \quad f(e_n) = a.$$

We define a new orientation g by setting

$$g(e) := \begin{cases} b & \text{if } e = e_0, \\ f(e_{i-1}) & \text{if } e = e_i, \ i > 0, \\ f(e) & \text{otherwise}. \end{cases}$$

It follows that

$$|g^{-1}(x)| = \begin{cases} |f^{-1}(a)| - 1 & \text{if } x = a, \\ |f^{-1}(b)| + 1 & \text{if } x = b, \\ |f^{-1}(x)| & \text{otherwise}. \end{cases}$$

Hence, $w(g) < w(f)$. Repeating this construction we obtain an orientation f with $w(f) = 0$.

The general case where $\mathfrak{H}$ may be infinite can be proved using the compactness theorem for first-order logic. Let Δ be the elementary diagram of $\mathfrak{H}$ (i.e., the set of all first-order formulae with parameters that hold in $\mathfrak{H}$; see [5] for details) where we consider $\mathfrak{H}$ as a two-sorted structure $\langle V, E, I \rangle$ with a binary incidence relation I. We can write down a formula φ stating that $f : E \to V$ is a function such that

- $\langle f(e), e \rangle \in I$, for all $e \in E$,
- $|f^{-1}(a)| \leq k$, for all $a \in V$.

By assumption and the first part of the proof, every finite subset of $\Delta \cup \{\varphi\}$ is satisfiable. Therefore, according to the compactness theorem, there exists a model $\mathfrak{H}^+ = \langle V^+, E^+, I^+, f^+ \rangle$ of $\Delta \cup \{\varphi\}$. By the Diagram Lemma (see, e.g., [5]), we can find an elementary embedding $h : \mathfrak{H} \to \mathfrak{H}^+$ (i.e., an embedding preserving every first-order formula). Since every edge of $\mathfrak{H}$ has only finitely many vertices it follows that

$$\langle a, h(e) \rangle \in I^+ \quad \text{implies} \quad a = h(v), \quad \text{for some } v \in e.$$

Hence, we can define the desired orientation of $\mathfrak{H}$ by $f := h^{-1} \circ f^+ \circ h$. $\square$

It turns out that the orientation obtained via the preceding lemma is MSO-definable. The following sequence of lemmas shows how we can encode such an orientation by a finite set of unary predicates.

Definition 1.6. Let $\mathfrak{H} = \langle V, E \rangle$ be a directed graph and $\mathfrak{G}$ an undirected one.

(a) Every orientation f of $\mathfrak{G}$ induces an directed graph $\mathfrak{G}_f$ by orienting every edge e of $\mathfrak{G}$ such that it points to the vertex $f(e)$.

(b) An $\mathfrak{H}$-*orientation* of $\mathfrak{G}$ consists of a pair $\langle f, h \rangle$ where f is an orientation of $\mathfrak{G}$ and h is a homomorphism $\mathfrak{G}_f \to \mathfrak{H}$.

We say that an $\mathfrak{H}$-orientation $\langle f, h \rangle$ is *bounded by* k if f is bounded by k.

(c) We say that a family $(P_v)_{v \in V}$ of unary predicates *encodes* an $\mathfrak{H}$-orientation $\langle f, h \rangle$ of $\mathfrak{G}$ if $P_v = h^{-1}(v)$, for all $v \in V$.

Lemma 1.7. *For every finite graph* $\mathfrak{H}$, *there exists a first-order formula* $\varphi_{\mathfrak{H}}(\bar{X})$ *such that*

$$\mathfrak{G} \models \varphi_{\mathfrak{H}}(\bar{P}) \quad \text{iff} \quad \text{the tuple } \bar{P} \text{ encodes an } \mathfrak{H}\text{-orientation of } \mathfrak{G}.$$

Proof. Let $u_0, \ldots, u_{n-1}$ be an enumeration of the vertices of $\mathfrak{H}$. All $\varphi_{\mathfrak{H}}(\bar{X})$ has to say is that the X_i form a partition of the vertices (some X_i may be empty) and that there is no edge $\{v, w\}$ of $\mathfrak{G}$ such that $v \in X_i$, $w \in X_k$ and $\langle u_i, u_k \rangle$ is *not* an edge of $\mathfrak{H}$. □

Theorem 1.8 (Nešetřil, Sopena, Vignal [6])**.** *For every $k < \omega$, there exists a finite loop-free directed graph $\mathfrak{T}_k$ with antisymmetric edge relation that has the following property. For every finite directed graph $\mathfrak{G}$, with irreflexive and antisymetric edge relation and indegree at most k, there exists a homomorphism $\mathfrak{G} \to \mathfrak{T}_k$.*

Corollary 1.9. *Every k-sparse undirected graph has a $\mathfrak{T}_k$-orientation which is bounded by k.*

Proof. In Lemma 1.5, we have shown that such a graph $\mathfrak{G} = \langle V, E \rangle$ has an orientation $f : E \to V$ that is bounded by k. It follows that $\mathfrak{G}_f$ has indegree at most k. By the theorem, there exists a homomorphism $h : \mathfrak{G}_f \to \mathfrak{T}_k$. Thus, $\langle f, h \rangle$ is the desired $\mathfrak{T}_k$-orientation. □

Lemma 1.10. *For every $k < \omega$, there exists a first-order formula $\eta_k(\bar{X})$ such that*

$$\mathfrak{G} \models \eta_k(\bar{P}) \quad \text{iff} \quad \bar{P} \text{ encodes a } \mathfrak{T}_k\text{-orientation of } \mathfrak{G} \text{ that is bounded by } k \,.$$

Proof. Note that the homomorphism h of a $\mathfrak{T}_k$-orientation $\langle f, h \rangle$ uniquely determines the orientation f since the edge relation of $\mathfrak{T}_k$ is antisymmetric. In particular, the parameters $\bar{P}$ encoding $\langle f, h \rangle$ tell us whether f is bounded by k. Hence, we can obtain $\eta_k(\bar{X})$ by adding a check for boundedness to the formula $\varphi_{\mathfrak{T}_k}(\bar{X})$ of Lemma 1.7. □

Corollary 1.11. *The class of all k-sparse undirected graphs is finitely MSO-axiomatisable.*

Proof. By Lemma 1.5 and Corollary 1.9 it follows that a graph $\mathfrak{G}$ is k-sparse if and only if it has a $\mathfrak{T}_k$-orientation that is bounded by k. Hence, we can use the formula $\exists \bar{X} \eta_k(\bar{X})$ where η_k is the formula from Lemma 1.10. □

In order to apply these results to hypergraphs we use the following construction associating a graph with every hypergraph.

Definition 1.12. Let $\mathfrak{H} = \langle V, E \rangle$ be a hypergraph with orientation f. We define a directed graph $\mathcal{O}_f(\mathfrak{H}) := \langle V, F \rangle$ with edge relation

$$F := \{ \langle a, b \rangle \mid a \neq b \text{ and there is some edge } e \in E \text{ with } a \in e \text{ and } f(e) = b \} \,.$$

Lemma 1.13. *Let $\mathfrak{H} = \langle V, E \rangle$ be a k-sparse hypergraph of rank m where $0 < k < \omega$ and $1 < m < \omega$. Then $\mathfrak{H}$ has an orientation f that is bounded by mk^2 such that the edge relation of $\mathcal{O}_f(\mathfrak{H})$ is antisymmetric.*

Proof. First, we consider the case that $\mathfrak{H}$ is finite. We call an element $a \in V$ *bad* for an orientation f of $\mathfrak{H}$ if there is some element $b \in V$ such that $\mathcal{O}_f(\mathfrak{H})$ contains both edges $\langle a, b \rangle$ and $\langle b, a \rangle$. Note that this implies that the vertex b is also bad.

We construct a sequence of orientations $(f_n)_n$ such that

$$|f_n^{-1}(a)| \leq \begin{cases} k & \text{if } a \text{ is bad for } f_n \,, \\ mk^2 & \text{otherwise} \,, \end{cases}$$

and the number of bad elements decreases at every step. We start with an arbitrary orientation f_0 bounded by k.

Given an orientation f_n with the above properties we construct a new orientation f_{n+1} with fewer bad elements as follows. Let a be a bad element, set $X := f_n^{-1}(a)$, and let

$$Y := \{ e \mid a \in e \text{ and } f_n(e) \in \bigcup X \setminus \{a\} \} \,.$$

Since a is bad we have

$$|X| \leq k \quad \text{and} \quad |\bigcup X| \leq k(m-1)\,.$$

Note that every element of the form $b := f_n(e)$ with $e \in Y$ is also bad since, by definition of X, there is an edge $e' \in X$ with

$$b \in e' \quad \text{and} \quad f_n(e') = a\,.$$

Consequently, $\mathcal{O}_{f_n}(\mathfrak{H})$ contains the edges $\langle b, a \rangle$ (since $f_n(e') = a$) and $\langle a, b \rangle$ (since $f_n(e) = b$). It follows that

$$|Y| \leq k \cdot |\bigcup X \setminus \{a\}| \leq k^2(m-1)\,.$$

We define the new orientation f_{n+1} by

$$f_{n+1}(e) := \begin{cases} a & \text{if } e \in Y\,, \\ f_n(e) & \text{otherwise}\,. \end{cases}$$

Then we have

$$|f_{n+1}(x)^{-1}| \leq \begin{cases} k + k^2(m-1) & \text{if } x = a\,, \\ |f_n(x)^{-1}| & \text{otherwise}\,. \end{cases}$$

In particular, f_{n+1} is bounded by mk^2. By construction, the element a is not bad for f_{n+1}. Furthermore, if $\langle b, c \rangle$ is an edge in $\mathcal{O}_{f_{n+1}}(\mathfrak{H})$ with $b, c \neq a$ then this edge is induced by an edge e in $\mathfrak{H}$ with $e \notin X \cup Y$. Hence, $\langle b, c \rangle$ is also an edge of $\mathcal{O}_{f_n}(\mathfrak{H})$. Therefore, every element that is bad for f_{n+1} is also bad for f_n.

It remains to prove the claim for infinite hypergraphs $\mathfrak{H}$. Let Φ be the union of the elementary diagram of $\mathfrak{H}$ and formulae stating that f is an orientation of $\mathfrak{H}$ that is bounded by mk^2 and that $\mathcal{O}_f(\mathfrak{H})$ has an antisymmetric edge relation. If $\mathfrak{M}$ is a model of Φ then there exists an embedding $h : \mathfrak{H} \to \mathfrak{M}$ and the desired orientation of $\mathfrak{H}$ can be obtained via h from that of $\mathfrak{M}$. Hence, it is sufficient to show that Φ is satisfiable. Note that every finite subset $\Phi_0 \subseteq \Phi$ is satisfiable since every finite substructure of $\mathfrak{H}$ has an orientation of the desired form. By the compactness theorem it follows that Φ is satisfiable. $\square$

2. Depth-first spanning trees

While k-sparse hypergraphs are MSO-orientable there are hypergraphs without an MSO-definable orientation. For instance, the countably infinite clique is such a graph. In this section we will show that every hypergraph of bounded rank is at least GSO-orientable. A basic tool the proof below is based on is the notion of a spanning tree of a hypergraph. Before presenting the rather involved definition for hypergraphs let us start with considering the simpler case of graphs.

For a countable undirected graph $\mathfrak{G}$ we can define a *depth-first spanning tree* to be a spanning tree $\mathfrak{T}$ of $\mathfrak{G}$ where no edge of $\mathfrak{G}$ connects disjoint subtrees of $\mathfrak{T}$ (see [2, 3]; in [3] such trees are called *normal*). To generalise this definition to uncountable graphs we have to admit trees of arbitrary ordinal height. Such trees are necessarily *order trees,* i.e., partial orders $\langle T, \leq \rangle$ where $\leq$ is a *tree order,* that is, a partial order such that any two elements have an infimum and, for every element a, the set of all elements below a is well-ordered. Unfortunately, we cannot in general hope to have a spanning subgraph that is an order tree, since the partial order $\leq$ requires too many edges. Therefore, we will use a hybrid between

an ordinary tree and an order tree. The precise definition of a spanning tree $\mathfrak{T}$ of a graph $\mathfrak{G}$ is as follows. Instead of requiring $\mathfrak{T}$ to be a subgraph of $\mathfrak{G}$ we consider trees $\mathfrak{T}$ such that

- for every vertex w of $\mathfrak{T}$ with immediate predecessor v, the edge $\langle v, w \rangle$ belongs to $\mathfrak{G}$, and
- for every vertex w of $\mathfrak{T}$ without immediate predecessor, we can fix an increasing chain $(u_i)_{i<\alpha}$ of predecessors of w with limit w and a family $(\pi_i)_{i<\alpha}$ of paths from w to u_i.

Hence, every vertex w of $\mathfrak{T}$ is attached to its predecessors via some auxiliary graph F_w that is either a single edge or a tree with root w whose leaves form an increasing sequence of predecessors of w with limit w.

Example 2.1. Consider the complete graph $\mathfrak{K}_\kappa$, for an uncountable cardinal κ. We can enumerate the vertices of $\mathfrak{K}_\kappa$ as $(v_\alpha)_{\alpha<\kappa}$ where the index α ranges over all ordinals less than κ. As depth-first spanning tree of this graph we can use a chain of length κ as follows. We set $\mathfrak{T} := \langle T, E \rangle$ where

$$T := \{\, v_\alpha \mid \alpha < \kappa \,\}$$

is the set of all vertices and

$$E := \{\, \langle v_\alpha, v_{\alpha+1} \rangle \mid \alpha < \kappa \,\} \cup \{\, \langle v_\alpha, v_\delta \rangle \mid \delta \text{ a limit ordinal and } \alpha < \delta \,\}.$$

The first part of E consists of the successor edges, whereas the second part contains the auxiliary graphs F_{v_δ} attaching a limit vertex v_δ to its predecessors.

To generalise these ideas to hypergraphs we need a suitable replacement for the trees F_w. Unfortunately, not every hypergraph has a spanning tree. A typical example is the hypergraph
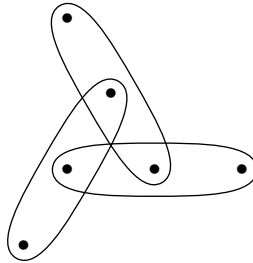
Instead, we will use certain tree-like hypergraphs called *priority trees.*

Definition 2.2. Let $\mathfrak{H} = \langle V, E \rangle$ be a hypergraph. A *hyperpath* in $\mathfrak{H}$ is a sequence $e_0 \ldots e_n$ of edges such that

$$e_i \cap e_k \neq \emptyset \quad \text{iff} \quad |i - k| \leq 1\,.$$

If $u \in e_0 \setminus e_1$ and $v \in e_n \setminus e_{n-1}$ then we say that the hyperpath *connects u and v.*

Definition 2.3. Let $\mathfrak{H} = \langle V, E \rangle$ be a hypergraph of rank at most m, $\langle T, F \rangle$ be a subhypergraph of $\mathfrak{H}$ with $T = \bigcup F$, and suppose that there are partitions

$$T = P_0 \uplus \cdots \uplus P_{m-1} \quad \text{and} \quad F = F_0 \uplus \cdots \uplus F_{m-1}\,.$$

(a) Suppose that $\mathfrak{T} = \langle T, F, L, (F_i)_{i<m}, (P_i)_{i<m}, v \rangle$ with $L \subseteq F$ and $v \in T$. We define by induction when such a tuple $\mathfrak{T}$ is a *priority tree.* The element v is called the *root* of $\mathfrak{T}$ and L is its set of *leaf edges.*

We start the induction with the case where F consists of a single hyperpath $e_0 \ldots e_n$ with $v \in e_0 \setminus e_1$, we have $L = \{e_n\}$, $F_0 = F$, and $P_0 = T$. Then $\mathfrak{T}$ is a priority tree. We also call

$\mathfrak{T}$ a priority tree if it can be obtained from a priority tree $\mathfrak{T}' = \langle T', F', L', (F'_i)_i, (P'_i)_i, v \rangle$ with the same root v by adding a hyperpath $e_0 \ldots e_n$ such that

$$e_i \cap T' \neq \emptyset \quad \text{iff} \quad i = 0 \,,$$
$$e_0 \nsubseteq T' \,,$$
$$L = L' \cup \{e_n\} \,,$$
$$e_0, \ldots, e_n \in F_k \,,$$
$$(e_0 \cup \cdots \cup e_n) \setminus T' \subseteq P_k \,,$$

where k is the minimal index such that $e_0 \cap P'_k = \emptyset$. This is the successor case of the induction step.

Finally, we also have a limit case. Suppose that

$$\mathfrak{T}^\alpha = \langle T^\alpha, F^\alpha, L^\alpha, (F_i^\alpha)_i, (P_i^\alpha)_i, v \rangle \,, \qquad \text{for } \alpha < \beta \,,$$

is an increasing chain of priority trees. That is, the sequences $(T^\alpha)_\alpha$, $(F^\alpha)_\alpha$, $(L^\alpha)_\alpha$, $(F_i^\alpha)_\alpha$, and $(P_i^\alpha)_\alpha$ are all increasing, and all trees $\mathfrak{T}^\alpha$ have the same root v. Then $\mathfrak{T}$ is a priority tree if it is the union of this chain, that is, if

$$T = \bigcup_{\alpha<\beta} T^\alpha, \quad F = \bigcup_{\alpha<\beta} F^\alpha, \quad L = \bigcup_{\alpha<\beta} L^\alpha, \quad F_i = \bigcup_{\alpha<\beta} F_i^\alpha, \quad P_i = \bigcup_{\alpha<\beta} P_i^\alpha.$$

(b) A *branch* of $\mathfrak{T}$ is a hyperpath $e_0 \ldots e_m \subseteq F$ satisfying the following conditions:
- $e_0 \setminus e_1$ contains the root v of T.
- Let k_i be the index such that $e_i \in F_{k_i}$. We have $e_{i+1} \setminus e_i \subseteq P_{k_{i+1}}$, for every $i < m$. Furthermore, if $k_i \neq k_{i+1}$ then $k_i := \min \{ l \mid e_{i+1} \cap P_l \neq \emptyset \}$.

(c) With each priority tree $\mathfrak{T}$ we associate two relations, an order $\leq$ on F defined by

$$e \leq f \quad :\text{iff} \quad \text{every branch containing } f \text{ also contains } e,$$

and an equivalence relation $\sim$ on T defined by

$$u \sim v \quad :\text{iff} \quad u, v \in P_k \,, \text{ for some } k \,, \text{ and there exists a hyperpath}$$
$$e_0 \ldots e_m \subseteq F_k \text{ connecting } u \text{ and } v \,.$$

Example 2.4. Consider the following priority tree with edges a, b, c, d, e, f where we have labelled each vertex in P_i by the index i. The edge colours are given by $F_0 = \{a, b, e\}$, $F_1 = \{c\}$, $F_2 = \{d, f\}$. The ordering $\leq$ is displayed to the right.
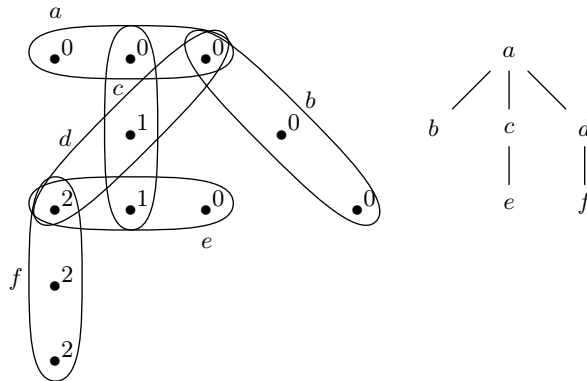
Recall that a *tree order* is a partial order such that any two elements have an infimum and, for every element a, the set of all elements below a is well-ordered. A *preorder* is a reflexive and transitive relation. Every preorder $\sqsubseteq$ induces an equivalence relation $\sqsubseteq \cap \sqsubseteq^{-1}$. The equivalence classes of this relation are called $\sqsubseteq$-*classes*.

Lemma 2.5. *Let $\mathfrak{T}$ be a priority tree. The order $\leq$ on the edges is a tree order.*

The proof consists of a straightforward but tedious induction following the construction of priority trees.

Lemma 2.6. *Let $\mathfrak{H} = \langle V, E \rangle$ be a connected hypergraph of rank at most m.*
(a) *For each vertex $v \in V$ and every set $L_0 \subseteq E$ of edges, there exists a priority tree $\mathfrak{T} = \langle T, F, L, (F_i)_i, (P_i)_i, v \rangle$ with root v such that $\bigcup L_0 \subseteq T$ and $L \subseteq L_0$.*
(b) *For every GSO-formula $\vartheta(x, y)$ (possibly with parameters), there exists a GSO-formula $\varphi(x, y)$ (with parameters) such that, if ϑ defines a well-order on L_0 and $\mathfrak{T}$ is a priority tree as in (a) then $\varphi(x, y)$ defines a linear order on T.*

Proof.

(a) Let $(e_i)_{i < \alpha}$ be an enumeration of L_0. For every $i < \alpha$, we fix a hyperpath $\pi_i = h_0^i \dots h_{m_i}^i$ connecting v with $h_{m_i}^i = e_i$. We construct $\mathfrak{T}$ by induction on i. We start with the hyperpath π_0. At step $i > 0$ we determine the shortest suffix $h_l^i \dots h_{m_i}^i$ of the path π_i that meets the tree constructed so far and we add this suffix to the tree. (If $e_i \subseteq T$ we leave the tree unchanged.) We choose the least index k with $h_l^i \cap P_k = \emptyset$ and we put the new edges into F_k and the new vertices into P_k. The limit of this construction is the desired priority tree.

(b) The equivalence relation $\sim$ associated with $\mathfrak{T}$ is MSO-definable in $\mathfrak{H}$ with the help of the parameters T, F, F_i, and P_i. We denote the $\sim$-class of a vertex u by $[u]$. Note that, by construction of $\mathfrak{T}$, $[u]$ is a hyperpath and $[u]$ contains a unique leaf edge which, furthermore, is one of the ends of the hyperpath. We denote by $\eta(u)$ the suffix of the hyperpath $[u]$ that connects u to the leaf edge in $[u]$.

To define the desired order on T we first construct a preorder on T by setting $x \sqsubseteq y$ if and only if one of the following conditions is satisfied:
- $x \in P_i$ and $y \in P_k$, for $i < k$.
- $x, y \in P_k$ and the leaf edge in $[x]$ is ϑ-smaller than the leaf edge in $[y]$.
- $x, y \in P_k$, $[x] = [y]$, and $\eta(x) \subseteq \eta(y)$.

Note that we have $x \sqsubseteq y$ and $y \sqsubseteq x$ if and only if $\eta(x) = \eta(y)$. In this case x and y belong to the same edge $e \in F$. Hence, every $\sqsubseteq$-class has size at most m. Adding m additional unary predicates $Q_0, \dots, Q_{m-1}$ such that each Q_i contains at most one element of each $\sqsubseteq$-class, we can define

$$x < y \quad :\text{iff} \quad x \sqsubseteq y \text{ and (either } y \not\sqsubseteq x, \text{ or we have}$$
$$x \in Q_i \text{ and } y \in Q_k, \text{ for } i < k). \qquad \square$$

We have seen that every k-sparse graph has an MSO-definable orientation f that is bounded by k. If we want to encode sets of edges via sets of vertices we can try to encode each edge e by a pair $\langle v, i \rangle$ consisting of the vertex $v := f(e)$ and a number $i < k$. This idea requires a way to linearly order the sets $f^{-1}(v)$. In [2] Courcelle uses depth-first spanning trees to obtain such linear orders. As remarked above one needs to adapt the definition of a depth-first spanning tree when one tries to extend these results to uncountable hypergraphs.

Definition 2.7. Let $\mathfrak{H} = \langle V, E \rangle$ be a hypergraph of rank m and suppose that $\mathfrak{T} = \langle T, \leq, (F_v)_{v \in T} \rangle$ is a structure where $\langle T, \leq \rangle$ is a tree (of ordinal height) with $T \subseteq V$ and with every vertex $v \in T$ we associate a set $F_v \subseteq E$ of edges. We assume that $F_u \cap F_v = \emptyset$, for $u \neq v$.

(a) The set of *auxiliary nodes* associated to a vertex $v \in T$ is

$$A_v := \{v\} \cup \bigcup F_v \setminus \bigcup_{x < v} A_x .$$

(b) For $X \subseteq V$, we define

$$B(X/T) := \{\, v \in T \mid X \cap A_v \neq \emptyset \,\} \quad \text{and} \quad \beta(X/T) := \max B(X/T) .$$

(c) $\mathfrak{T}$ is a *depth-first spanning tree* of $\mathfrak{H}$ if it satisfies the following conditions:
- For all $u \neq v$, $A_u \cap A_v = \emptyset$ and $A_u \cap T = \{u\}$.
- For each edge $e \in E$ the set $B(e/T)$ is nonempty and linearly ordered by $\leq$.
- The vertices $v \in T$ are partitioned into the following classes: (0) the root; $(s_l)_{l<m}$ a successor; $(t_l)_{l<m}$ a limit; where the successor and limit vertices are subdivided into m subclasses. This partition satisfies the following conditions:
  - (0) If v has type 0 then it is the root of T and $F_v = \emptyset$.
  - (s_l) If v has type s_l then it is the (immediate) successor of some vertex $u \in T$. We have $F_v = \{e\}$ with $v \in e$. Futhermore, v is the only vertex in $B(e/T)$ of type s_l and $\beta(e \setminus \{v\}/T) = u$.
  - (t_l) If v has type t_l then it is the limit of an increasing sequence $(u_i)_{i<\gamma}$ of vertices $u_i \in T$. F_v is (the set of edges of) a priority tree with root v. Furthermore,

$$\{\, \beta(e/T) \mid e \text{ a leaf edge of } F_v \,\}$$

    is a cofinal subset of $(u_i)_i$ and v is the only vertex in $B(\bigcup F_v/T)$ with type t_l.

Proposition 2.8. *Every connected hypergraph $\mathfrak{H}$ has a depth-first spanning tree.*

Proof. If in the definition of a depth-first spanning tree we drop the condition that $B(e/T) \neq \emptyset$, for every edge e, then we obtain a structure that we call a *partial* depth-first spanning tree. We construct an increasing sequence

$$\langle T_\alpha, \leq, (F_v)_{v \in T_\alpha} \rangle , \quad \alpha < \kappa ,$$

of such partial depth-first spanning trees with the property that, for every connected component C of $U_\alpha := V \setminus \bigcup_{v \in T_\alpha} A_v$, the set

$$N(C/T_\alpha) := \bigcup \{\, B(e/T_\alpha) \mid e \in E \text{ with } e \cap C \neq \emptyset \,\}$$

is linearly ordered by $\leq$. (A *connected component* of U_α is a maximal subset $C \subseteq U_\alpha$ such that the subhypergraph $\mathfrak{H}|_C$ is connected.) The limit of this sequence will be the desired depth-first spanning tree of $\mathfrak{H}$.

We start by choosing an arbitrary element $v \in V$ and setting $T_0 := \{v\}$ and $F_v := \emptyset$. For limit ordinals δ, we define $T_\delta := \bigcup_{\alpha < \delta} T_\alpha$. For the successor step, suppose that we have already defined T_α. Fix some connected component C of U_α. Note that $N(C/T_\alpha)$ is nonempty since $\mathfrak{H}$ is connected. We distinguish two cases.

(1) If $N(C/T_\alpha)$ has a maximal element u then we choose some edge e with $e \cap A_u \neq \emptyset$ and $e \cap C \neq \emptyset$, and we fix some vertex $v \in e \cap C$. We add v to T_α as immediate successor of u and we set $F_v := \{e\}$. It follows that $A_v = e \cap U_\alpha$. Since $B(e/T_\alpha)$ contains at most

$|e \setminus \{v\}| < m$ vertices there is some $l < m$ such that $B(e/T_\alpha)$ contains no vertex of type s_l. Hence, in the new tree $T_{\alpha+1}$ we can assign the type s_l to v.

(2) Suppose that $N(C/T_\alpha)$ has no maximal element. We choose a sequence $(e_i)_{i<\gamma}$ of edges with $e_i \cap C \neq \emptyset$ such that the sequence $(u_i)_{i<\gamma}$ defined by

$$u_i := \beta(e_i/T_\alpha)$$

is increasing and cofinal in $N(C/T_\alpha)$. By taking a suitable subsequence we may assume that the set of types appearing in $B(e_i/T_\alpha)$ is the same for every $i < \gamma$.

For each edge e_i, choose some edge $h_i \subseteq C$ with $h_i \cap e_i \neq \emptyset$ and set $L := \{\, h_i \mid i < \gamma \,\}$. We select a vertex $v \in C$ and a priority tree $\mathfrak{S} = \langle S, H, L, \bar{F}, \bar{P}, v \rangle$ such that $S \subseteq C$. We define $T_{\alpha+1} := T_\alpha \cup \{v\}$ where v is the limit of $N(C/T_\alpha)$ and we set $F_v := H \cup \{\, e_i \mid i < \gamma \,\}$. It follows that $A_v = S \cup \bigcup_i (e_i \cap U_\alpha)$.

It remains to show that the constructed tree $T_{\alpha+1}$ is a partial depth-first tree where all sets $N(C/T_{\alpha+1})$ are linearly ordered. We start by showing that each set $B(e/T_{\alpha+1})$ with $e \in E$ is linearly ordered. If $e \cap A_v = \emptyset$ then $B(e/T_{\alpha+1}) = B(e/T_\alpha)$ and we are done. Otherwise, we have $B(e/T_{\alpha+1}) = B(e/T_{\alpha+1}) \cup \{v\}$. Note that $A_v \subseteq C$ implies $e \cap C \neq \emptyset$. Therefore, we have $B(e/T_\alpha) \subseteq N(C/T_\alpha)$. Since v is larger than every element in $N(C/T_\alpha)$ the claim follows.

Let D be a connected component of $U_{\alpha+1} := V \setminus \bigcup_{x \in T_{\alpha+1}} A_x$. We have to show that $N(D/T_{\alpha+1})$ is linearly ordered. Since $U_{\alpha+1} \subseteq U_\alpha$ there is some connected component D' of U_α containing D. If $D' \neq C$ then $U_\alpha \setminus U_{\alpha+1} \subseteq C$ implies that $D = D'$ and the set

$$N(D/T_{\alpha+1}) = N(D'/T_\alpha)$$

is linearly ordered. If, on the other hand, $D \subseteq C$ then we have

$$N(D/T_{\alpha+1}) \subseteq N(C/T_\alpha) \cup \{v\}$$

and the latter set is linearly ordered since v is greater than every element of $N(C/T_\alpha)$. $\square$

Remark 2.9.

(a) If the hypergraph $\mathfrak{H}$ is countable then we can actually obtain a depth-first spanning tree of height at most ω as follows. In the above proof, if we are slightly more careful in choosing the vertex v that is added to the partial tree, then we can ensure that every vertex is chosen already after finitely many steps.

(b) Note that, strictly speaking, the above proposition is not a generalisation of Theorem 1.4 of [2] since we use a different notion of a depth-first spanning tree.

We use depth-first spanning trees to encode orientations of a hypergraph. First, we show that each depth-first spanning tree can be encoded by finitely many GSO-parameters.

Lemma 2.10. *For every $m < \omega$ we can construct* MSO*-formulae $\varphi(X; \bar{Z})$, $\vartheta(x, Y; \bar{Z})$, and $\chi(x, y; \bar{Z})$ such that, for every connected hypergraph $\mathfrak{H}$ of rank at most m and each depth-first spanning tree $\langle T, \leq, (F_v)_v \rangle$ of $\mathfrak{H}$, there are* GSO*-parameters $\bar{S}$ such that*

$$\mathfrak{H} \models \varphi(P; \bar{S}) \qquad \text{iff} \qquad P \subseteq T \text{ is downward } \leq\text{-closed,}$$
$$\mathfrak{H} \models \vartheta(v, P; \bar{S}) \qquad \text{iff} \qquad v \in T \text{ and } P = A_v,$$
$$\mathfrak{H} \models \chi(u, v; \bar{S}) \qquad \text{iff} \qquad u, v \in T \text{ and } u \leq v.$$

Proof. We will use the following parameters:

• Unary predicates T_0, T_{s_l}, T_{t_l}, for $l < m$, containing all vertices of the corresponding type.

- $F^\tau := \bigcup \{ F_v \mid v \in T_\tau \}$, for every type τ.
- $F := \bigcup_\tau F^\tau$.
- $A^\tau := \bigcup \{ A_v \mid v \in T_\tau \} \setminus T$, for every type τ.
- If v is of type t_l then the set F_v forms a priority tree. We use additional parameters $F_i^{t_l}$ and P_i, for $i < m$, encoding the corresponding partition

$$F^{t_l} = F_0^{t_l} \uplus \cdots \uplus F_{m-1}^{t_l} \quad \text{and} \quad \bigcup F^{t_l} = P_0 \uplus \cdots \uplus P_{m-1}.$$

Hence, for every vertex v of type t_l we have a priority tree

$$\left\langle \bigcup F_v, F_v, L_v, (F_i^{t_l} \cap F_v)_i, (P_i \cap \bigcup F_v)_i, v \right\rangle.$$

(1) First, we construct the formula ϑ. To simplify our task we define separate formulae $\vartheta_\tau(x, Y)$, for each type τ, such that

$$\mathfrak{H} \models \vartheta_\tau(v, P) \quad \text{iff} \quad v \in T_\tau \text{ and } P = A_v.$$

Then we can set $\vartheta := \bigvee_\tau \vartheta_\tau$.

If v has type 0 then $A_v = \{v\}$ and we can set

$$\vartheta_0(x, Y) := T_0 x \wedge Y = \{x\}.$$

If the type of v is s_l then $F_v = \{e\}$ and $A_v = \{v\} \cup (e \cap A^{s_l})$, where e is the unique edge in F^{s_l} containing v. Hence, we can define

$$\vartheta_{s_l}(x, Y) := T_{s_l} x \wedge (\exists e \in F^{s_l})[x \in e \wedge Y = \{x\} \cup (e \cap A^{s_l})].$$

Finally, if v has type t_l then A_v is the least subset of $\{v\} \cup A^{t_l}$ satisfying the following conditions:

- $v \in Y$
- If $e \in F^{t_l}$ and $e \cap Y \neq \emptyset$ then $e \cap A^{t_l} \subseteq Y$.

Hence, we can define

$$\vartheta_{t_l}(x, Y) := T_{t_l} x \wedge Y x \wedge (\forall e \in F^{t_l})(e \cap Y \neq \emptyset \to e \cap A^{t_l} \subseteq Y)$$
$$\wedge \forall Z [Z x \wedge (\forall e \in F^{t_l})(e \cap Z \neq \emptyset \to e \cap A^{t_l} \subseteq Z) \to Y \subseteq Z].$$

(2) Next, we define a formula $\alpha(x, Y)$ such that

$$\mathfrak{H} \models \alpha(v, Q) \quad \text{iff} \quad v \in T \text{ and } Q = B(\bigcup F_v / T).$$

The formula $\alpha(x, Y)$ should state that

$$Y = \{ u \in T \mid \text{there is some } e \in F_v \text{ with } e \cap A_x \neq \emptyset \text{ and } e \cap A_u \neq \emptyset \}.$$

Using the formulae ϑ_τ we can write α as

$$\alpha(x, Y) := \bigvee_\tau [T_\tau x \wedge (\exists e \in F^\tau)(e \cap A_x \neq \emptyset \wedge e \cap A_u \neq \emptyset)].$$

(3) With the help of α we can write down the desired formulae φ and χ.

$$\varphi(X) := \forall x \forall Y [X x \wedge \alpha(x, Y) \to Y \subseteq X],$$
$$\chi(x, y) := \forall X [\varphi(X) \to (X y \to X x)].$$

$\square$

Lemma 2.11. *We can construct* GSO-*formulae* $\psi_m(x, y; \bar{Z})$*, for* $m < \omega$*, such that, for every depth-first spanning tree* $\langle T, \leq, (F_v)_v \rangle$ *of a connected hypergraph* $\mathfrak{H}$ *of rank* m*, there are* GSO-*parameters* $\bar{S}$ *such that the formula* $\psi_m(x, y; \bar{S})$ *defines a preorder* $\sqsubseteq_0$ *with the following properties:*

- *The restriction of* $\sqsubseteq_0$ *to* T *coincides with* $\leq$*.*
- $\sqsubseteq_0$ *linearly preorderes every set* $X \subseteq V$ *such that* $B(X/T)$ *is linearly ordered by* $\leq$*.*
- *Each* $\sqsubseteq_0$*-class has at most* m *elements.*

Proof. Let $\chi(x, y)$ and $\vartheta(x, Y)$ be the formulae of Lemma 2.10. For each type τ, we define a formula $\eta_\tau(x, y)$ linearly preordering each set A_v where v is of type τ. Then the desired formula ψ_m states that either

(1) $x \in A_u$ and $y \in A_v$ for $u < v$, or
(2) $x, y \in A_v$, for some v of type τ, and $\eta_\tau(x, y)$ holds.

If v is of type 0 or s_l then A_v contains at most m elements and we can set

$$\eta_\tau(x, y) := \text{true}.$$

For vertices of type t_l we can use the formula from Lemma 2.6. $\square$

Corollary 2.12. *We can construct* GSO-*formulae* $\psi_m(x, y; \bar{Z})$*, for* $m < \omega$*, such that, for every depth-first spanning tree* $\langle T, \leq, (F_v)_v \rangle$ *of a connected hypergraph* $\mathfrak{H}$ *of rank* m*, there are* GSO-*parameters* $\bar{S}$ *such that the formula* $\psi_m(x, y; \bar{S})$ *defines a partial order* $\sqsubseteq$ *with the following properties:*

- *The restriction of* $\sqsubseteq$ *to* T *coincides with* $\leq$*.*
- $\sqsubseteq$ *linearly orderes every set* $X \subseteq V$ *such that* $B(X/T)$ *is linearly ordered by* $\leq$*.*

Proof. Let $\sqsubseteq_0$ be the preorder from Lemma 2.11. Since every $\sqsubseteq_0$-class contains at most m elements we can add m new unary predicates $P_0, \dots, P_{m-1}$ such that $P_0 \cup \dots \cup P_{m-1} = V$ and we have $|X \cap P_i| \leq 1$, for each $\sqsubseteq_0$-class X and all i. Then we can define

$$
\begin{aligned}
u \sqsubseteq v \quad &: \text{iff} \quad \text{either } u \sqsubset_0 v, \text{ or} \\
&\qquad u \sqsubseteq_0 v, \ v \sqsubseteq_0 u, \ u \in P_i, \ v \in P_k \text{ for } i < k.
\end{aligned}
$$
                              $\square$

Theorem 2.13. *We can construct* GSO-*formulae* $\varphi_m(x, Y; \bar{Z})$*, for* $m < \omega$*, such that for every hypergraph* $\mathfrak{H}$ *of rank* m*, there are* GSO-*parameters* $\bar{S}$ *such that, the formula* $\varphi_m(x, Y; \bar{S})$ *defines an orientation of* $\mathfrak{H}$*.*

Proof. Suppose that $\mathfrak{H}$ has κ connected components C_i, $i < \kappa$. For each component C_i we fix a depth-first spanning tree $\langle T^i, \leq^i, (F_v^i)_v \rangle$. Let $\bar{S}^i$ be the parameters from Lemma 2.10 and Lemma 2.11. For every edge $e \in E$, there exists a unique component C_i such that the intersection $X := e \cap \bigcup_v A_v^i$ is finite and nonempty. Furthermore, the set $B(X/T^i)$ is linearly ordered by $\leq$. Using the ordering $\sqsubseteq$ of Corollary 2.12 we can write down a formula $\varphi_m(v, e)$ stating that v is the $\sqsubseteq$-least element of this set X. $\square$

Corollary 2.14. *Every hypergraph of rank* $m < \omega$ *is* GSO-*orientable.*

Let us mention the following consequences of this result. For countable hypergraphs they are again due to Courcelle [2].

Definition 2.15.
(a) A formula $\varphi(x, y, Z)$ defines an *edge ordering* of a hypergraph $\mathfrak{H} = \langle V, E \rangle$ if, for every edge $e \in E$, the formula $\varphi(x, y, e)$ defines a linear ordering on the vertices of e.

(b) A formula $\varphi(x, y, z)$ defines an *neighbourhood ordering* of a directed graph $\mathfrak{G} = \langle V, E \rangle$ if, for every vertex $v \in V$, the formula $\varphi(x, y, v)$ defines a linear ordering on the set $\{\, u \in V \mid (u, v) \in E \,\}$.

Lemma 2.16. *There exist* GSO-*formulae* $\varphi_m(x, y, Z; \bar{U})$, *for* $m < \omega$, *such that, for every hypergraph* $\mathfrak{H} = \langle V, E \rangle$ *of rank* m, *there are* GSO-*parameters* $\bar{S}$ *such that the formula* $\varphi_m(x, y, Z; \bar{S})$ *defines an edge ordering of* $\mathfrak{H}$.

Lemma 2.17. *There exist* MSO-*formulae* $\varphi_m(x, y, z; \bar{U})$, *for* $m < \omega$, *such that, for every directed graph* $\mathfrak{G}$ *of indegree at most* m, *there are* MSO-*parameters* $\bar{P}$ *such that the formula* $\varphi_m(x, y, z; \bar{P})$ *defines a neighbourhood ordering of* $\mathfrak{G}$.

Proof. We can apply Lemma 2.16 to the hypergraph $\mathfrak{H} := \langle V, F \rangle$ where

$$F := \{\, I(v) \mid v \in V \,\} \quad \text{with} \quad I(v) := \{\, u \in V \mid (u, v) \in E \,\}\,.$$

Note that every subset $S \subseteq F$ can be encoded by the set

$$I^{-1}(F) := \{\, v \in V \mid I(v) \in F \,\} \subseteq V\,.$$

Hence, every GSO-formula over $\mathfrak{H}$ can be translated into an MSO-formula over $\mathfrak{G}$. $\qquad\square$

3. GSO versus MSO

In [2] Courcelle has shown that we can translate every GSO-formula φ into an MSO-formula ψ that is equivalent to φ on all countable k-sparse hypergraphs. Using the results of the previous sections we can lift the restriction to countable hypergraphs. The proof in [2] goes through unchanged since it relies only on the statements of Lemma 2.16 and Lemma 2.17, and on local modifications of hypergraphs.

Theorem 3.1. *For all numbers* $m, k < \omega$, *there exists a monadic second-order interpretation (with monadic parameters) that maps a* k-*sparse hypergraph of rank* m *to its incidence structure.*

Corollary 3.2. *For all* $m, k < \omega$ *and all formulae* $\varphi(\bar{x}, \bar{Y}, \bar{Z}) \in$ GSO *with first-order variables* $\bar{x}$, *monadic variables* $\bar{Y}$, *and guarded second-order variables* $\bar{Z}$, *there exists a formula* $\psi(\bar{x}, \bar{Y}, \bar{Z}) \in$ MSO *with the following property: for all* k-*sparse hypergraphs* $\mathfrak{H} = \langle V, E \rangle$ *of rank* m *and all parameters* $a_i \in V$, $P_i \subseteq V$, $R_i \subseteq E$, *there exist parameters* $Q_i \subseteq V$ *such that*

$$\mathfrak{H} \models \varphi(\bar{a}, \bar{P}, \bar{R}) \quad \text{iff} \quad \mathfrak{H} \models \psi(\bar{a}, \bar{P}, \bar{Q})\,.$$

4. Sparse distributions

The results so far concern ways to encode edges by vertices. In this last section we consider a more general problem. Let $\mathfrak{G} = \langle V, E \rangle$ be a graph. We denote by $\mathscr{P}_{\mathrm{fin}}(V)$ the set of all finite subsets of V. We would like to encode a given subset $F \subseteq \mathscr{P}_{\mathrm{fin}}(V)$ by a set of vertices, that is, we would like to find a definable function $h : F \to V$ that is injective. For $F = E$ this reduces to the problem considered in the preceding sections. For arbitrary F, such a function h does not always exist. But we will show that sometimes we can transform a given function $h_0 : F \to V$ into an injective one.

These results are inspired by work of Colcombet and Löding [1] on set interpretations. Colcombet and Löding consider a power set operation $\mathcal{P}$ on structures. One of their main results in a commutation theorem for interpretations and the power set operation. They show that, given a tree $\mathfrak{T}$ and an FO-interpretation $\mathcal{I}$ such that $\mathcal{I}(\mathcal{P}(\mathfrak{T}))$ is of the form $\mathcal{P}(\mathfrak{M})$, for some structure $\mathfrak{M}$, then there exists a WMSO-interpretation $\mathcal{J}$ such that $\mathfrak{M} \cong \mathcal{J}(\mathfrak{T})$. On ingredient in the proof of this result is a method to encode, in a definable way, finite subsets of the tree $\mathfrak{T}$ by single vertices.

Suppose we are given a function $h_0 : F \to V$ that we want to transform into an injective function $h : F \to V$. Let $\delta(v) := |h_0^{-1}(v)|$. The first step in the construction of h consists in finding a definable function $g : V \to V$ such that $|g^{-1}(v)| = \delta(v)$, for all v. Of course, this is not always possible. For instance, if the graph is finite and we have $\delta(v) > 1$, for all vertices v. Therefore, we consider only functions δ that are *sparse* in the sense of the following definition.

Definition 4.1. Let $\mathfrak{G} = \langle V, E \rangle$ be an undirected graph.
(a) The *border* of a subset $Z \subseteq V$ is the set
$$B_{\mathfrak{G}}(Z) := E \cap (V \setminus Z) \times Z$$
of all edges connecting a vertex in Z with a vertex outside of Z.
(b) A *distribution* of $\mathfrak{G}$ is a map $\delta : V \to \omega$. For $X \subseteq V$, we define the shorthand
$$\delta(X) := \sum_{v \in X} \delta(x) \,.$$
(c) Let $h : X \to V$ be an arbitrary mapping. The distribution *induced* by h is the function $\delta : V \to \omega$ with
$$\delta(v) := |h^{-1}(v)| \,.$$
(d) A distribution δ is *k-sparse* if
$$\delta(Z) \leq |Z| + k \cdot |B_{\mathfrak{G}}(Z)| \,, \quad \text{for every } Z \subseteq V \,.$$

Given a k-sparse distribution δ we will construct the desired function $g : V \to V$ by solving a network flow problem.

Definition 4.2. Let $\mathfrak{G} = \langle V, E \rangle$ be an undirected graph.
(a) A *flow* of $\mathfrak{G}$ is a function $f : V \times V \to \mathbb{Z}$ such that, for all $u, v \in V$,
   - $f(u, v) = -f(v, u)$ and
   - $f(u, v) \neq 0$ implies $(u, v) \in E$.
(b) A flow f is *acyclic* if there is no cycle $u_0, \ldots, u_m$ of $\mathfrak{G}$ such that $f(u_m, u_0) > 0$ and $f(u_i, u_{i+1}) > 0$, for all $i < m$.
(c) The *defect* of a flow f is the distribution
$$d_f(v) := \sum_{u \in V} f(v, u) \,.$$
(d) A flow f is a *δ-flow* if, for every $v \in V$, either
$$d_f(v) = \delta(v) - 1 \quad \text{or} \quad \delta(v) = 0 \text{ and } d_f(v) = 0 \,.$$

(e) A flow f is *edge-bounded* by k if $|f(u,v)| \leq k$, for all $u,v \in V$. We call f *vertex-bounded* by k if

$$\sum_{u \in V} |f(u,v)| \leq k, \quad \text{for all } v \in V.$$

Our aim is to show that, for every k-sparse distribution δ there is a bounded δ-flow f and a function $g : V \to V$ inducing δ. Furthermore, if δ is definable then g should also be definable.

Definition 4.3. Let L be a logic.

(a) A distribution δ is *L-definable* if there exist formulae $\varphi_i(x) \in L$, $i < k$, such that

$$\mathfrak{G} \models \varphi_i(v) \quad \text{iff} \quad \delta(v) = i.$$

(b) Similarly, a flow f is *L-definable* if there exist formulae $\varphi_i(x,y) \in L$ such that

$$\mathfrak{G} \models \varphi_i(u,v) \quad \text{iff} \quad f(u,v) = i.$$

Remark 4.4. Note that every edge-bounded flow can be encoded with the help of the GSO-parameters

$$S_i := \{(u,v) \in E \mid f(u,v) = i\}.$$

For trees the problem of encoding sets by vertices has been solved by Colcombet and Löding [1]. In the general case proved below the function g is only definable with the help of GSO-parameters, but for trees we can do without them.

Theorem 4.5 (Colcombet and Löding [1]). *Let $\mathfrak{T} = \langle T, E \rangle$ be an infinite directed tree and δ a WMSO-definable k-sparse distribution of $\mathfrak{T}$. There exists a WMSO-definable flow f that is edge-bounded by $7k$ and satisfies $d_f(v) \geq \delta(v) - 1$, for all v.*

Theorem 4.6 (Colcombet and Löding [1]). *Let $\mathfrak{T} = \langle T, E \rangle$ be a directed tree and δ a WMSO-definable k-sparse distribution of $\mathfrak{T}$ such that $\delta(T) \leq |T|$. There exists WMSO-definable function $g : T \to T$ such that δ is the distribution induced by g.*

To prove our generalisation of these results we start with a few lemmas about bounded flows. The first two follow immediately from the definitions.

Lemma 4.7. *Every flow that is vertex-bounded by k is also edge-bounded by k.*

Lemma 4.8. *Suppose that $\mathfrak{G}$ is a graph with maximal degree d. Every flow of $\mathfrak{G}$ that is edge-bounded by k is vertex-bounded by dk.*

Lemma 4.9. *For every δ-flow f there exists an acyclic δ-flow f' such that, if f is edge-bounded by k or vertex-bounded by k then so is f'.*

Proof. We repeat the following construction until the flow is acyclic. Select a cycle $u_0, \ldots, u_m$ such that

$$c := \min \{ f(u_i, u_{i+1}) \mid i \leq m \} > 0.$$

We define f' by

$$f'(x,y) := \begin{cases} f(x,y) - c & \text{if } x = u_i \text{ and } y = u_{i+1}, \text{ for some } i, \\ f(x,y) + c & \text{if } x = u_{i+1} \text{ and } y = u_i, \text{ for some } i, \\ f(x,y) & \text{otherwise.} \end{cases} \qquad \square$$

Proposition 4.10. *Let $\mathfrak{G} = \langle V, E \rangle$ be an undirected graph and δ a k-sparse distribution. Then $\mathfrak{G}$ has a δ-flow f that is edge-bounded by k.*

Proof. First, we assume that $\mathfrak{G}$ is finite. In this case we can reduce the task to a network flow problem. Let $\mathfrak{H}$ be the graph obtained from $\mathfrak{G}$ by adding two new vertices s and t that are connected to every vertex of $\mathfrak{G}$. We define the capacity $c(e)$ of edges e of $\mathfrak{H}$ as follows. For edges e of $\mathfrak{G}$ we set $c(e) := k$. If $e = (s, v)$ with $v \in V$ we set $c(e) := \max\{0, \ \delta(v) - 1\}$. Finally, if $e = (v, t)$ with $v \in V$ we define

$$c(e) := \begin{cases} 0 & \text{if } \delta(v) > 0, \\ 1 & \text{otherwise}. \end{cases}$$

Let f be a maximal flow from s to t with respect to c. We claim that its restriction to the edges of $\mathfrak{G}$ is the desired flow.

According to the Max-Flow Min-Cut Theorem, there is a set X of vertices containing s but not t such that the maximal flow m from s to t equals

$$m = \sum_{e \in B_{\mathfrak{H}}(X)} c(e).$$

Let $X_0 := X \setminus \{s\} \subseteq V$ and $Y := \delta^{-1}(0)$. Since

$$B_{\mathfrak{H}}(X) = B_{\mathfrak{G}}(X_0) \cup \{ (v, t) \mid v \in X_0 \} \cup \{ (s, v) \mid v \in V \setminus X_0 \},$$

we have

$$\begin{aligned} m &= \sum_{e \in B_{\mathfrak{H}}(X)} c(e) \\ &= k \cdot |B_{\mathfrak{G}}(X_0)| + |X_0 \cap Y| + \delta(V \setminus X_0) - |(V \setminus X_0) \setminus Y| \\ &= k \cdot |B_{\mathfrak{G}}(X_0)| + |X_0| + \delta(V \setminus X_0) - |(V \setminus X_0) \setminus Y| - |X_0 \setminus Y| \\ &\geq \delta(X_0) + \delta(V \setminus X_0) - |V \setminus Y| \\ &= \delta(V) - |V \setminus Y|. \end{aligned}$$

On the other hand, for the set $X = \{s\}$, we have

$$m \leq \sum_{e \in B_{\mathfrak{H}}(X)} c(e) = \sum_{v \in V} \max\{0, \ \delta(v) - 1\} = \delta(V) - |V \setminus Y|.$$

Consequently, the maximal flow m from s to t equals

$$m = \delta(V) - |V \setminus Y|.$$

This implies that

$$f(s, v) = \max\{0, \ \delta(v) - 1\}, \quad \text{for every } v \in V.$$

For each $v \in V$, we therefore have

$$0 = \sum_{u \in V \cup \{s,t\}} f(u, v) = \max\{0, \delta(v) - 1\} + f(t, v) + \sum_{u \in V} f(u, v).$$

If $\delta(v) > 0$ this implies

$$\delta(v) - 1 - \sum_{u \in V} f(v, u) = 0, \quad \text{that is} \quad d_f(v) = \delta(v) - 1,$$

while, for $\delta(v) = 0$, we have

$$-f(v,t) - \sum_{u \in V} f(v,u) = 0\,.$$

Hence, either $d_f(v) = -1 = \delta(v) - 1$ or $d_f(v) = 0$.

It remains to prove the lemma for infinite graphs. Let $\Phi(\mathfrak{G})$ consist of the elementary diagram of $\mathfrak{G}$ together with first-order formulae stating that f is a δ-flow on $\mathfrak{G}$ that is edge-bounded by k. We will use the compactness theorem to show that $\Phi(\mathfrak{G})$ is satisfiable.

Let $\Phi_0 \subseteq \Phi(\mathfrak{G})$ be finite. There exists a finite induced subgraph $\mathfrak{G}_0 = \langle V_0, E_0 \rangle$ of $\mathfrak{G}$ such that $\Phi_0 \subseteq \Phi(\mathfrak{G}_0)$. Let $\langle u_0, v_0 \rangle, \dots, \langle u_{m-1}, v_{m-1} \rangle$ be an enumeration (without repetitions) of all edges $\langle u, v \rangle$ with $u \in V_0$ and $v \in V \setminus V_0$. We construct a new graph $\mathfrak{G}'_0 = \langle V'_0, E'_0 \rangle$ by attaching to each vertex u_i a path P_i of length k. Let δ' be the distribution on $\mathfrak{G}'_0$ with $\delta'(v) = \delta(v)$, for $v \in V_0$, and $\delta'(v) = 0$, for $v \in V'_0 \setminus V_0$. In order to show that Φ_0 is satisfiable it is sufficient to prove that $\mathfrak{G}'_0$ has a flow of the desired form. Consider an arbitrary set $X \subseteq V'_0$ of vertices. Let

$$I := \{\, i \mid u_i \in X \,\} \quad \text{and} \quad J := \{\, i \mid u_i \in X \text{ and } P_i \subseteq X \,\}\,.$$

It follows that

$$\begin{aligned}
\delta'(X) = \delta(X \cap V_0) &\leq |X \cap V_0| + k \cdot |B_{\mathfrak{G}}(X \cap V_0)| \\
&\leq |X| - k \cdot |J| + k \cdot |B_{\mathfrak{G}'_0}(X \cap V_0)| \\
&\leq |X| - k \cdot |J| + k \cdot \big(|B_{\mathfrak{G}'_0}(X)| + |J|\big) = |X| + k \cdot |B_{\mathfrak{G}'_0}(X)|\,.
\end{aligned}$$

By the first part of the proof it follows that $\mathfrak{G}'_0$ has a flow of the desired form. $\square$

It remains to show how we can use the δ-flow f we have just constructed to define the desired function $g : V \to V$. We start by selecting a certain family of definable paths. Note that we allow paths of length 0. Such paths are uniquely determined by the vertex they start (and end) at.

Lemma 4.11. *Let $\mathfrak{G}$ be a countable undirected graph and f an acyclic δ-flow of $\mathfrak{G}$. There exists a set $\mathcal{P}$ of finite paths through $\mathfrak{G}$ satisfying the following conditions:*

(i) *For every $v \in V$, there are exactly $\delta(v)$ paths in $\mathcal{P}$ starting at v.*
(ii) *For every $v \in V$ there is at most one path in $\mathcal{P}$ ending at v.*
(iii) *For every pair $u, v \in V$ of vertices there are at most $f(u,v)$ paths in $\mathcal{P}$ containing the edge (u,v) (in this direction).*

Proof. Fix an enumeration $(v_n, k_n)_{n < \omega}$ of the set

$$\{\, \langle v, k \rangle \mid v \in V,\ 0 \leq k < \delta(v) \,\}\,.$$

For $n < \omega$, we construct paths π_n with the following properties:

- π_n starts at v_n.
- If $m \neq n$ then the endpoints of π_m and π_n are different.
- For every edge (u,v) there are at most $f(u,v)$ paths π_n containing the edge (u,v).

By induction, suppose that we have already defined π_i, for $i < n$. Let

(1) $\alpha(v)$ be the number of paths π_i, $i < n$, starting at v,
(2) $\beta(v)$ the number of paths π_i, $i < n$, ending at v, and
(3) $\mu(u,v)$ the number of paths π_i, $i < n$, containing the edge (u,v).

We construct a path $u_0 \ldots u_m$ inductively starting with $u_0 := v_n$. For the induction step, suppose that we have already defined $u_0, \ldots, u_i$. If $\beta(u_i) = 0$ then we stop and set $\pi_n := u_0 \ldots u_i$. Otherwise, we claim that there is some neighbour w of u_i with $f(u_i, w) > \mu(u_i, w)$. Hence, we can set $u_{i+1} := w$.

To prove the claim, we distinguish two cases. If $i = 0$ then $\alpha(u_0) < \delta(u_0)$ implies that

$$
\begin{aligned}
\sum_{x \in V} \mu(u_0, x) &= \alpha(u_0) - \beta(u_0) + \sum_{x \in V} \mu(x, u_0) \\
&\leq \alpha(u_0) - 1 + \sum \{\, f(x, u_0) \mid f(x, u_0) \geq 0 \,\} \\
&= \alpha(u_0) - 1 + \sum \{\, f(u_0, x) \mid f(u_0, x) \geq 0 \,\} - (\delta(u_0) - 1) \\
&< \sum \{\, f(u_0, x) \mid f(u_0, x) \geq 0 \,\},
\end{aligned}
$$

as desired. Similarly, if $i > 0$ then $\mu(u_{i-1}, u_i) < f(u_{i-1}, u_i)$ implies that

$$
\begin{aligned}
\sum_{x \in V} \mu(u_i, x) &= \alpha(u_i) - \beta(u_i) + \sum_{x \in V} \mu(x, u_i) \\
&< \alpha(u_i) - 1 + \sum \{\, f(x, u_i) \mid f(x, u_i) \geq 0 \,\} \\
&= \alpha(u_i) - 1 + \sum \{\, f(u_i, x) \mid f(u_i, x) \geq 0 \,\} - (\delta(u_i) - 1) \\
&\leq \sum \{\, f(u_i, x) \mid f(u_i, x) \geq 0 \,\}.
\end{aligned}
$$

Note that the construction of π_n must terminate after at most $n + 1$ steps since the flow f is acyclic and there are only n vertices u with $\beta(u) = 1$. $\qquad\square$

Lemma 4.12. *There exist* GSO-*formulae* $\varphi_m(X; \bar{Z})$, *for* $m < \omega$, *such that, for every graph* $\mathfrak{G}$ *and each set* $\mathcal{P}$ *of finite paths such that every vertex and every edge of* $\mathfrak{G}$ *is contained in at most* m *paths of* $\mathcal{P}$, *there exists a tuple* $\bar{S}$ *of* GSO-*parameters such that*

$$\mathfrak{G} \models \varphi_m(P; \bar{S}) \quad \text{iff} \quad P \text{ is (the set of edges of) a nonempty path in } \mathcal{P}.$$

Proof. For every edge (u, v) of $\mathfrak{G}$ we fix a bijection $\mu(u, v) : [n] \to \mathcal{P}_e$ where $\mathcal{P}_e \subseteq \mathcal{P}$ is the set of all paths containing the edge (u, v) (in either direction) and $n := |\mathcal{P}_e|$. We assume that $\mu(u, v) = \mu(v, u)$.

Let S be the set of all edges of $\mathfrak{G}$ contained in some path in $\mathcal{P}$. By Lemma 2.17 there exists an MSO-formula $\chi(x, y, z; \bar{S}')$ with parameters $\bar{S}'$ such that, for every $v \in V$, the formula $\chi(x, y, v; \bar{S}')$ linearly orders the set of all vertices that are connected to v via an edge in S.

Finally, we define unary predicates Q^{ik}_{jl} containing all vertices v such that there exists a path $\pi \in \mathcal{P}$ containing edges (u, v), (v, w) where

- $\mu(u, v)(k) = \pi$, $\mu(v, w)(l) = \pi$,
- u is the i-th neighbour of v (in the order defined by χ),
- w is the j-th neighbour of v.

It follows that a nonempty set $P \subseteq E$ of edges is a path in $\mathcal{P}$ if and only if P is a minimal nonempty subset of E satisfying the following condition:

P can be written as a union $P = P_0 \cup \cdots \cup P_{m-1}$ such that, for all vertices u, v, w such that $v \in Q^{ik}_{jl}$ and u and w are, respectively, the i-th and j-th neighbour of v, we have $(u, v) \in P_k \Leftrightarrow (v, w) \in P_l$.

This condition can be expressed in GSO. $\qquad\square$

Remark 4.13. Note that the set of empty paths in $\mathcal{P}$ is trivially definable with the help of the parameter

$$Q := \{\, v \in V \mid \mathcal{P} \text{ contains an empty path from } v \text{ to } v \,\}\,.$$

Using the family $\mathcal{P}$ we can construct a formula φ defining the function g.

Proposition 4.14. *There exist GSO-formulae $\varphi_m(x, y; \bar{Z})$, for $m < \omega$, with the following property: for every graph $\mathfrak{G} = \langle V, E \rangle$ and each acyclic δ-flow f of $\mathfrak{G}$ that is vertex-bounded by m, there exist GSO-parameters $\bar{S}$ such that $\varphi_m(x, y; \bar{S})$ defines on $\mathfrak{G}$ a partial function $g : V \to V$ with*

$$|g^{-1}(v)| = \delta(v)\,, \quad \text{for all } v \in V\,.$$

Proof. Let $\mathfrak{G}'$ be the graph obtained from $\mathfrak{G}$ by removing every edge (u, v) with $f(u, v) = 0$. Note that f is also a δ-flow of $\mathfrak{G}'$. Since f is vertex-bounded by m it follows that every vertex of $\mathfrak{G}'$ has degree at most $m < \omega$. Consequently, each connected component $\mathfrak{G}_0$ of $\mathfrak{G}$ is countable. Let $\mathcal{P}_0$ be the set of paths obtained by applying Lemma 4.11 to the restriction of f to $\mathfrak{G}_0$, and let $\mathcal{P}$ be the union of all these sets $\mathcal{P}_0$ corresponding to the connected components of $\mathfrak{G}'$. By Lemma 4.12, there exists a formula $\psi(X; \bar{Z})$ and a set $\bar{S}$ of guarded relations such that

$$\mathfrak{G} \models \psi(P; \bar{S}) \quad \text{iff} \quad P \text{ is a nonempty path in } \mathcal{P}\,.$$

With the help of ψ we can define a partial function $g : V \to V$ such that

$$g(v) = u \quad :\text{iff} \quad \mathcal{P} \text{ contains a path from } u \text{ to } v\,.$$

By construction of $\mathcal{P}$ we have $|g^{-1}(v)| = \delta(v)$, for every $v \in V$. $\qquad\square$

Lemma 4.15. *Let $\mathfrak{G} = \langle V, E \rangle$ be a graph of finite degree and $\varphi(X, y)$ a GSO-formula that defines a partial function $h : \mathscr{P}(V) \to V$ such that the distribution δ induced by h is k-sparse. Suppose that there exists a GSO-formula $\chi(X, Y, z)$ such that, for every vertex $v \in V$, $\chi(X, Y, v)$ linearly orders the set $h^{-1}(v)$. Then there exist MSO-definable partial functions $h_0 : \mathscr{P}(V) \to V$ and $g : V \to V$ such that $h = g \circ h_0$ and h_0 is injective.*

Proof. By Proposition 4.10 there exists a δ-flow f that is edge-bounded by k. Since $\mathfrak{G}$ has finite degree it follows that f is vertex-bounded by some constant $m < \omega$. Hence, we can use Proposition 4.14 to find a definable function $g : V \to V$ with $|g^{-1}(v)| = \delta(v) = |h^{-1}(v)|$. Choose unary predicates $P_0, \ldots, P_{k-1}$ such that we have $i \neq l$ whenever $u \in P_i$ and $v \in P_l$ are distinct vertices with $g(u) = g(v)$. Using these predicate we can define partial functions $g_0, \ldots, g_{k-1} : V \to V$ such that $g_i(v)$ is the unique element of $g^{-1}(v) \cap P_i$. We define $h_0 : \mathscr{P}(V) \to V$ by $h_0(X) := (g_i \circ h)(X)$ where the index i is chosen such that X is the i-th element of $h^{-1}(h(X))$ (in the order defined by χ). It follows that $h(X) = g(h_0(X))$ and h_0 is injective. Furthermore, the function h_0 is clearly GSO-definable. Since the graph $\mathfrak{G}$ has degree at most k it is k-sparse. Hence, every GSO-definable function is already MSO-definable. $\qquad\square$

Recall that $\mathscr{P}_{\text{fin}}(V)$ denotes the set of all finite subsets of V. Combining the preceding lemmas we obtain the main result of this section.

Theorem 4.16. *Let $\mathfrak{G} = \langle V, E \rangle$ be a graph of finite degree and $\varphi(X, y)$ a GSO-formula that defines a partial function $h : \mathscr{P}_{\text{fin}}(V) \to V$ such that the distribution δ induced by h is k-sparse. Then there exist MSO-definable partial functions $h_0 : \mathscr{P}_{\text{fin}}(V) \to V$ and $g : V \to V$ such that $h = g \circ h_0$ and h_0 is injective.*

Proof. By the preceding lemma it is sufficient to construct a formua $\chi(X, Y, z)$ (with GSO-parameters) such that $\chi(X, Y, v)$ linearly orders $h^{-1}(v)$, for every $v \in V$. Let $T_0 \subseteq E$ be a spanning forest of $\mathfrak{G}$ and let $P \subseteq V$ be a set containing exactly one element of each connected component. Using the parameters P and T_0 we can define the tree ordering on V by

$$u \leq v \quad :\text{iff} \quad \text{the unique path in } T_0 \text{ from some element of } P \text{ to } v \text{ contains } u.$$

Let $T \subseteq V \times V$ be the set obtained from T_0 by orienting the edges according to this ordering. Then T is a directed forest. Furthermore, since the degree of $\mathfrak{G}$ is bounded we can use Lemma 2.17 to linearly order the successors of every vertex in T. We use these two orderings to define the lexicographic ordering $\leq_{\text{lex}}$ on T. Finally, we obtain the desired ordering on $\mathscr{P}_{\text{fin}}(V)$ by setting

$$X < Y \quad :\text{iff} \quad \text{the } \leq_{\text{lex}}\text{-minimal element of } (X \setminus Y) \cup (Y \setminus X) \text{ belongs to } Y.$$

Each of these definitions can be expressed in GSO. $\qquad\square$

5. CONCLUSION

We have presented several methods to encode sets of finite vertices as single vertices. In the first part, we used depth-first spanning trees to encode edges by vertices. As an application we were able to extend Courcelle's result on the collapse of GSO to MSO on sparse hypergraphs from countable hypergraphs to hypergraphs of arbitrary cardinality. In the second part we used network flows to encode arbitrary finite sets by vertices.

Let us mention some open questions. Considering the first part it would be interesting to find out whether sparse classes are the only examples where GSO collapses to MSO.

Problem 5.1. Is there a class C that is not k-sparse, for any k, such that over C every GSO-sentence is equivalent to an MSO-sentence?

The results of the second part are much less complete. It is unlikely that they are the best possible.

Problem 5.2. Improve Theorem 4.16 by allowing
(a) more general classes of graphs or hypergraphs;
(b) more general classes of partial functions $h : \mathscr{P}(V) \to V$.

Our results were inspired by work of Colcombet and Löding [1]. The question arises of whether we can also generalise the remaining results of that article.

Problem 5.3. Can we prove Corollary 4.4 of [1] for other graphs than trees?

ACKNOWLEDGEMENT

REFERENCES

[1] T. Colcombet and C. Löding, *Transforming Structures by Set Interpretations*, Logical Methods in Computer Science, 3 (2007).

[2] B. Courcelle, *The monadic second-order logic of graphs XIV: Uniformly sparse graphs and edge set quantifications*, Theoretical Computer Science, 299 (2003), pp. 1–36.

[3] R. Diestel, *Graph Theory*, Springer, 3rd ed., 2006.

[4] E. Grädel, C. Hirsch, and M. Otto, *Back and Forth Between Guarded and Modal Logics*, ACM Transactions on Computational Logics, (2002), pp. 418–463.

[5] W. Hodges, *Model Theory*, Cambridge University Press, 1993.

[6] J. Nešetřil, E. Sopena, and L. Vignal, *T-preserving homomorphisms of oriented graphs*, Comment. Math. Univ. Carolinae, 38 (1997), pp. 125–136.

[7] D. Seese, *The structure of the models of decidable monadic theories of graphs*, Annals of Pure and Applied Logic, 53 (1991), pp. 169–195.