

Is hyper-extensionality preservable under deletions of graph elements?¹

Alberto Casagrande²

*Dept. Mathematics and Geosciences
University of Trieste
Trieste, Italy*

Carla Piazza³

*Dept. Mathematics and Computer Science
University of Udine
Udine, Italy*

Alberto Policriti⁴

*Dept. Mathematics and Computer Science
University of Udine
Udine, Italy*

Abstract

Any hereditarily finite set S can be represented as a finite pointed graph –dubbed membership graph– whose nodes denote elements of the transitive closure of $\{S\}$ and whose edges model the membership relation. Membership graphs must be hyper-extensional –nodes are pairwise not bisimilar– and bisimilar nodes represent the same hereditarily finite set. It is worth to notice that the removal of even a single node or edge from a membership graph can cause “collapses” of different nodes and, therefore, the loss of hyper-extensionality of the graph itself.

With the intent of gaining a deeper understanding of the class of hereditarily finite sets, this paper investigates whether pointed hyper-extensional graphs always contain either a node or an edge whose removal does not disrupt the hyper-extensionality property.

Keywords: Set theory, Hereditarily finite sets, Non-well-foundedness, Hyper-extensionality.

1 Introduction

A set is hereditarily finite if it is finite and all its elements are hereditarily finite. Moreover, it is well-founded if any chain of membership relations starting from it is

¹ This work has been partially supported by Istituto Nazionale di Alta Matematica (INdAM) and by University of Trieste FRA Project “*Learning specifications and robustness in signal analysis (with a case study related to health care)*”

² Email: acasagrande@units.it

³ Email: carla.piazza@uniud.it

⁴ Email: alberto.policriti@uniud.it

finite. In standard Set Theory the Extensionality axiom, establishing that two sets are equal if and only if they have the same elements, guarantees that hereditarily finite well-founded sets can be inductively constructed starting from the empty set.

When also cyclic chains of memberships are allowed sets are called non-well-founded and one of the possible principles for establishing equality is Aczel’s Anti-Foundation axiom based on the notion of bisimulation [2].

A hereditarily finite set S can be canonically represented through a pointed finite graph G in which each node represents a different element of the transitive closure of $\{S\}$ and the edges of G model the membership relation. Since the notion of bisimulation can be naturally defined also on graphs, this means that in the canonical representation of S there are not two different bisimilar nodes. Well-founded sets are represented by acyclic graphs, while non-well-founded sets are represented by cyclic ones (e.g., see [2] for more details).

Now a quite general question arises: is there a natural way to inductively reason on both well-founded and non-well-founded hereditarily finite sets represented through graphs? In other terms, is there a way to inductively construct/deconstruct graphs representing hereditarily finite sets? Such question has been previously formalized and studied in [14] where the authors ask whether given the canonical representation of a set, it is always possible to find a node which can be removed producing the canonical representation of another set, i.e., without causing any bisimulation collapse. A definitive answer is not provided in [14]. In this paper we further investigate in that direction proving that there are cases in which it is not possible to remove any node without causing collapses. On the other hand, we provide positive evidence on the fact that there always exists an edge which can be safely removed. This result is achieved by introducing the notion of *n-well-founded part* of a non-well-founded graph and by applying Ackermann code on it.

The paper is organized as follows: Section 2 formalizes hereditarily finite sets. Section 3 relates hereditarily finite sets and pointed hyper-extensional graphs and defines keystones –elements whose removal disrupts the graph hyper-extensionality. Section 4 presents a pipeline to enumerate pointed hyper-extensional graphs. This pipeline is used in Section 5 to prove that there exist pointed hyper-extensional graphs whose nodes (edges) are all keystones. Section 6 introduces the notion of disposable element –an element whose removal does not produce collapses between nodes of the same connected components– and shows a pointed hyper-extensional graph that do not contain disposable nodes. In Section 7, we prove that pointed hyper-extensional graphs always have a disposable edge. Finally, in Section 8, we draw conclusions and suggest future works.

2 Hereditarily Finite Sets

Hereditarily finite sets are finite sets whose elements are hereditarily finite sets.

We write $\mathcal{P}(S)$ to denote the powerset of S i.e. $\mathcal{P}(S) = \{S' \mid S' \subseteq S\}$

Definition 2.1 [Well-founded Hereditarily Finite Sets] *Well-founded hereditarily finite sets* are the elements of $\text{HF} \stackrel{\text{def}}{=} \bigcup_{i \in \mathbb{N}} \text{HF}_i$ where the HF_i ’s are defined as

follows:

$$\text{HF}_i \stackrel{\text{def}}{=} \begin{cases} \emptyset & \text{if } i = 0 \\ \mathcal{P}(\text{HF}_{i-1}) & \text{otherwise} \end{cases} \quad (1)$$

Definition 2.2 [Non-Well-Founded Hereditarily Finite Sets] *Non-well-founded hereditarily finite sets* are finite sets specified by finite systems of equations of the form:

$$\begin{cases} Y_0 = \{X_{0,0}, \dots, X_{0,m_0}\} \\ \vdots \\ Y_n = \{X_{n,0}, \dots, X_{n,m_n}\} \end{cases} \quad (2)$$

where $\{Y_0, \dots, Y_n\} \supseteq \{X_{0,0}, \dots, X_{n,m_n}\}$.

We denote the set of all the non-well-founded hereditarily finite sets by $\text{HF}^{1/2}$.

Let us notice that $\text{HF} \subseteq \text{HF}^{1/2}$.

3 From hereditarily finite sets to graphs

Definition 3.1 [Graph] A (*directed*) *graph* is a tuple (V, E) where V is a finite set of nodes and $E \subseteq V \times V$ is a set of edges.

If $G = (V, E)$ is a graph, we write $G \setminus e$ to indicate the graph G deprived of the edge e (i.e., $G \setminus e \stackrel{\text{def}}{=} (V, E \setminus \{e\})$) and $G \setminus v$ to denote the graphs G deprived of the node v and of all its incident edges (i.e., $G \setminus v \stackrel{\text{def}}{=} (V \setminus \{v\}, E \setminus (\{v\} \times V \cup V \times \{v\}))$). If $(v, w) \in E$, then we say that v is a *predecessor* of w and w is a *successor* of v . A *path* from v_0 to v_n is a sequence of nodes $v_0 \dots v_n$ such that $(v_{i-1}, v_i) \in E$ for all $i \in [1, n]$. If $v_0 \dots v_n$ is a path, then its *length* is n . If there exists a path from v_0 to v_n , then we say that v_n is *reachable* from v_0 .

Definition 3.2 [Pointed Graph] A graph is *pointed* if all its nodes are reachable from one of its nodes, which is called *root* of the graph.

If we interpret the edges of a pointed graph $G = (V, E)$ as a membership relation, i.e., $(v, w) \in E$ as $w \in v$, G depicts an element of $\text{HF}^{1/2}$. Whenever G is clear from the context, we may write $v \ni w$ in place of $(v, w) \in E$. By writing $v \ni^n w$ we mean that there exists a path, whose length is n , from v to w (i.e., $v = w$, if $n = 0$, or there exists a v' such that $v \ni v'$ and $v' \ni^{n-1} w$, if $n > 0$). If w is reachable from v (i.e., there exists a $n \in \mathbb{N}$ such that $v \ni^n w$), we can also write $v \ni^* w$.

A *cycle* is a path, whose length is greater than 0, from a node v to v itself. A graph is *acyclic* if it does not contains cycles. If it is not acyclic, then it is *cyclic*.

Definition 3.3 [Well-founded node] Let G be a graph. A *well-founded node* v of G is a node whose transitive closure –the subgraph of the nodes w such that $v \ni^* w$ – is acyclic. A *non-well-founded node* is a node that is not well-founded.

Two distinct pointed graphs can represent the same set. In order to map each hereditarily finite set in a single pointed graph, we need the notion of *bisimulation*.

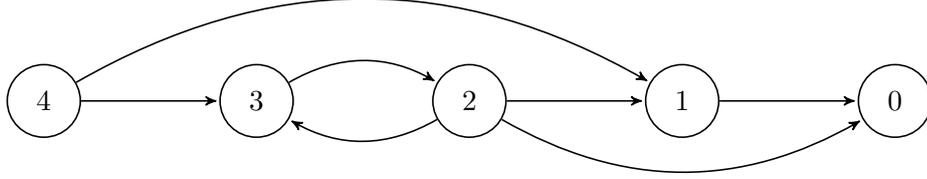


Fig. 1. This graph represents the hereditarily finite set $X_4 = \{X_3, X_1\}$ where $X_3 = \{X_2\}$, $X_2 = \{X_3, X_1, X_0\}$, $X_1 = \{X_0\}$, and $X_0 = \emptyset$.

Definition 3.4 [Bisimulation] Let $G = (V, E)$ and $G' = (V', E')$ be two graphs. A *bisimulation from G to G'* is a relation $R \subseteq V \times V'$ such that $(s, s') \in R$ if and only if:

- for all $s \ni p$ there exists a $s' \ni p'$ such that $(p, p') \in R$;
- for all $s' \ni p'$ there exists a $s \ni p$ such that $(p, p') \in R$.

If there exists a bisimulation R from G to G' such that $(v, v') \in R$, then we say that v and v' are bisimilar and we write $v \sim_{G \sim G'} v'$. If G and G' are the same graph, we may use the notation $\overset{G}{\sim}$ in place of $G \sim G'$. Whenever the graph G is clear from the context, we may also omit G from the notation $v \overset{G}{\sim} p$ by writing $v \sim p$. The relation \sim is a bisimulation and also an equivalence relation.

We say that two graphs G and G' are bisimilar whenever there exists a bisimulation R such that for all nodes v of G there exists a node v' in G' such that $(v, v') \in R$ and vice-versa. As we assumed the equality over hyper-set to be defined by bisimulation (Anti-Foundation Axiom [2]), two pointed graphs are bisimilar if and only if they represent the same hereditarily finite set.

Definition 3.5 [Collapsed Graph] Let $G = (V, E)$ be a graph and let $[v]_{\sim}$ be the set of nodes bisimilar to v in G , i.e., $[v]_{\sim} \stackrel{\text{def}}{=} \{w \in V \mid w \overset{G}{\sim} v\}$. The *collapsed graph* of G is the graph $G_{\sim} \stackrel{\text{def}}{=} (V_{\sim}, E_{\sim})$ where:

- $V_{\sim} \stackrel{\text{def}}{=} \{[v]_{\sim} \mid v \in V\}$;
- $E_{\sim} \stackrel{\text{def}}{=} \{([v]_{\sim}, [w]_{\sim}) \mid (v, w) \in E\}$.

If two graphs are bisimilar, then they share the same collapsed graph. Moreover, any graph is bisimilar to its collapsed graph. Thus, pointed collapsed graphs are a canonical form to represent hereditarily finite sets.

Definition 3.6 [Hyper-Extensional] A graph G is *hyper-extensional*, or *HE*, if the only bisimulation over it is the identity, i.e., $v \sim v'$ implies $v = v'$.

Since collapsed graphs are hyper-extensional, any hereditarily finite set corresponds to one pointed hyper-extensional graph (e.g., see [14]). Because of that, we sometime refer to pointed hyper-extensional graphs as *membership graphs*.

It is worth to underline that a membership graph may have many different roots and, thus, represent different sets. For instance, both the nodes 1 and 2 are valid roots for the graph depicted in Fig. 2.

It is easy to see that all the possible roots of a membership graph belong to the

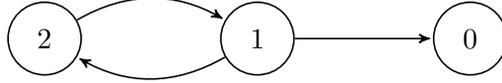


Fig. 2. Since both the nodes 1 and 2 are valid roots of the graph, this graph can represent both the hereditarily finite sets $X_2 = \{X_1\}$, where $X_1 = \{X_2, X_0\}$ and $X_0 = \emptyset$, and $X_4 = \{X_3, X_5\}$, where $X_5 = \{X_4\}$ and $X_3 = \emptyset$.

same strongly connected component.

Let us notice that nodes that share the same successors are bisimilar. Hence, if $G = (V, E)$ is a hyper-extensional graph, two nodes are the same if and only if they have the same successors. Because of that we may denote the set of the successors of $v \in V$ as v itself, i.e., $v = \{w \mid (v, w) \in E\}$; this is consistent with the notation \ni . Under the same conditions, if $v = \{v\}$ and $w = \{w\}$, then v and w are bisimilar. It follows that any hyper-extensional graph has, at most, one node v such that $v = \{v\}$ and we denote it by Ω . Analogously, any hyper-extensional graph has, at most, one node without successors and we write \emptyset to indicate it.

Proposition 3.7 *Let v be a node. It holds that $v \ni^* \emptyset$ if and only if $v \not\sim \Omega$.*

Proof. Let v be such that $v \ni^* \emptyset$ and $v \sim \Omega$. By definition of \ni^* , there should exist a path $v_0 \dots v_m$ such that $v_0 = v$ and $v_m = \emptyset$. Since Ω is the only successor of Ω and $v \sim \Omega$, $v_i \sim \Omega$ for all $i \in [0, m]$. It follows that $v_m = \emptyset \sim \Omega$. However, \emptyset has not successors and, thus, it is not bisimilar to Ω which has one successor. Thus, our assumptions were absurd and we proved the claim. \square

The following claim directly follows from Proposition 3.7.

Proposition 3.8 ([2]) *Let G be a hyper-extensional pointed graph. Either G contains the node \emptyset or G exclusively contains Ω .*

Nodes or edges whose deletion causes a collapse (i.e., reduce the number of nodes of the collapsed graph) are called *keystone*.

Definition 3.9 [Keystone] Let G be a pointed hyper-extensional graph. A node n (or an edge e , respectively) of G is a *keystone* for G , if the graph $G \setminus n$ ($G \setminus e$, respectively) is not hyper-extensional.

We are interested in establishing whether there are pointed hyper-extensional graphs in which all nodes (edges) are keystones or not. In the former case, we may identify some of them by enumerating pointed hyper-extensional graphs and testing whether all nodes (edges) are keystones.

In the following section we describe a pipeline for the enumeration of pointed hyper-extensional graphs. We used it to prove the results reported in the remaining parts of this paper.

4 Enumerating hereditarily finite sets

The enumeration of all the pointed hyper-extensional graphs up to a given order n –having n nodes– is inherently exponential with respect to $n^2/2$. As a matter of

fact, as n grows, the number of acyclic graphs having order n tends to $2^{\binom{n}{2}}/M\sigma^n$, where $M \approx 0.57436$ and $\sigma \approx 1.48807$ [3,4]. Moreover, roughly 32.6% of these graphs are (hyper-)extensional [21]. It follows that enumerating acyclic (hyper-)extensional graphs having order n lays in the time complexity class $\Omega(2^{n^2/2})$.

In order to produce all the pointed hyper-extensional graphs, we could both generate all the directed graphs and retain only those that are hyper-extensional and pointed⁵. Unfortunately, there are 2^{n^2} directed graphs of order n –having n nodes– and the large part of them are not even connected.

A significative improvement for this strategy was obtained by observing that both the properties of being hyper-extensional and pointed are preserved under isomorphism. Thus, either all graphs in an isomorphic class –the class of all the graphs that are pairwise isomorphic– are hyper-extensional or they are all not hyper-extensional. This also holds for the property of being pointed.

The isomorphic classes of directed graphs has been extensively studied (e.g., see [17,5,10,7]). They still are super-exponential in number with respect to the order n of the investigated graphs, (in particular, they are at least $2^{n^2}/n!$), but their abundance grows significantly slower than 2^{n^2} . For instance, for $n = 6, 32,$ and 64 each class of isomorphic graphs contains in average more than $44595, 10^{43},$ and 10^{108} elements, respectively (see [18]).

We implement a pipeline to enumerate all the pointed hyper-extensional graphs in SAGE [20]. A representative for each of the isomorphic classes is produced by using the SAGE command `canaug_traverse_edge(...)`. The pipeline should retain a graph only if it is hyper-extensional and pointed. In order to reduce the average time required to test these properties, two preliminary heuristics are applied. Since pointed graphs have at most one source –node that has no incoming edges–, the first heuristic filters graphs that have more than one source. We also noticed that all hyper-extensional pointed graphs, but the one representing Ω , must include exactly one sink node –node with no outgoing edges–, i.e., \emptyset . Thus, among the graphs that have survived the first filter, the pipeline considers exclusively the ones that either have one sink or that have no sinks and one node; the latter case correspond to Ω .

The next step is to identify pointed graphs. Given a graph G , the SAGE command `strongly_connected_components_digraph(G)` produces a new graph G' , analogous to it, in which each strong connected component of G has been collapsed to a distinct node. The resulting graph is pointed if and only if the original one is pointed too. Moreover, as G' does not contain non-trivial strongly connected components, if it is pointed, then it must have exactly one source. Hence, it is possible to decide whether G is pointed or not by both testing the existence of one single source in G' and, if this is the case, by performing a reachability computation from it.

Finally, the pipeline verifies hyper-extensionality of each of the remaining graphs by computing its maximum bisimulation [9,15]. If no pairs of nodes are bisimilar, the considered graph is pointed and hyper-extensional and, thus, it is kept.

Since all valid roots of a membership graph belong to the same strongly connected component, our pipeline is also able to compute the number of hereditarily

⁵ Let us notice that this is not an enumeration for pointed hyper-extensional graphs since two bisimilar graphs which are not isomorphic can be retain.

finite sets that are represented by graphs of a given order. In particular, the hereditarily finite sets that have each of the issued graphs \mathcal{G} as membership graph are in number as many as the nodes of the strongly connected components that contains a root for \mathcal{G} itself.

Table 1 lists, for each order up to 5, the number of isomorphic classes of directed graphs (with self-loops), the number of pointed hyper-extensional graphs, and the number of hereditarily finite sets as they are computed by our pipeline. The same table also details the number of well-founded hereditarily finite sets that is reported in [19].

5 Do non-keystone always exist?

In [14], it has been proved that if G is pointed hyper-extensional and acyclic (i.e., it represents a well-founded set), then not all the nodes of G are keystones. The more challenging case of cyclic graphs was left open.

Our pipeline, described in Section 4, can be used to produce all the pointed hyper-extensional graphs having up to 5 nodes. We test the existence of a non-keystone node in them by removing each of the nodes and by testing hyper-extensionality of the resulting graph. None of the considered graphs contains exclusively keystone nodes. Computing all the pointed hyper-extensional graphs of order 6 is too time consuming. However, by using the above method, we have discovered a graph in which all nodes are keystones and, as a consequence, we prove the following theorem.

Theorem 5.1 *There exists a non-empty pointed hyper-extensional graph such that all of its nodes are keystones.*

Proof. Let us consider the graph G_0 depicted in Fig. 3. All of its nodes, but 0,

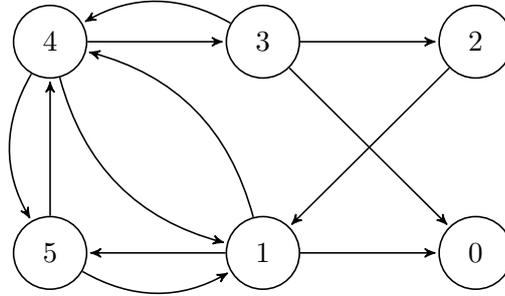


Fig. 3. A pointed hyper-extensional graph whose nodes are all keystones.

belong to the same strongly connected component (i.e. there is a path from v to w for all $v \neq 0$ and $w \neq 0$) and 0 is reachable from both 1 and 3. It follows that G_0 is pointed. Moreover, G_0 is hyper-extensional. However, $1 \stackrel{G_0 \setminus 0}{\sim} 2$, $0 \stackrel{G_0 \setminus 1}{\sim} 2$, $4 \stackrel{G_0 \setminus 2}{\sim} 5$, $4 \stackrel{G_0 \setminus 3}{\sim} 5$, $1 \stackrel{G_0 \setminus 4}{\sim} 3$, and $1 \stackrel{G_0 \setminus 5}{\sim} 3$. This proves the claim. \square

As far as keystone edges are concerned, we easily prove the following result.

Theorem 5.2 *There exists a pointed hyper-extensional graph such that it has at least one edge and all of its edges are keystones.*

Proof. Let us consider the graph G_1 depicted in Fig. 4. It is possible to reach 0 from 1, thus, it is pointed. Moreover, since 0 does not reach any node, G_1 is hyper-extensional. However, $G_1 \setminus (1 \ni 0)$ contains no edges and, hence, $0 \stackrel{G_1 \setminus (1 \ni 0)}{\sim} 1$. This proves the claim. \square

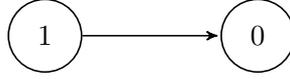


Fig. 4. A pointed hyper-extensional graph whose edges are all keystones.

Our pipeline highlights that, up to order 5, chains –connected graphs whose nodes, but \emptyset , have one single successor– are the only pointed hyper-extensional graphs whose edges are all keystones (see Table 1). This leads us to consider separately each of connected components produced by the elimination of a graph element. In the following section, we investigate whether there always exists a graph element whose elimination generates connected components that, individually, are hyper-extensional.

6 Hyper-extensionality and connected components

As a first step, we need to formalize the notion of connectivity over directed graphs. Such a notion, called *weak connectivity*, coincides with the connectivity over the corresponding undirected graph.

Definition 6.1 [Weak connectivity] A graph (V, E) is *weakly connected* if, for any $V' \subsetneq V$, there exists an edge $e \in E$ such that either $e \in V' \times (V \setminus V')$ or $e \in (V \setminus V') \times V'$.

Definition 6.2 [Disposable] Let G be a pointed hyper-extensional graph. A node n (or an edge e , respectively) of G is *disposable* whenever the weakly connected components of $G \setminus n$ (or $G \setminus e$, respectively) are hyper-extensional graphs.

Proposition 6.3 *If v is not disposable for G , then it is a keystone for it.*

Proof. If v is not disposable, then there exists a weakly connected component of $G \setminus v$ that is not hyper-extensional and there exists a bisimulation that is not the identity for it. By extending this bisimulation with the identity over the other weakly connected components, we obtain a bisimulation for $G \setminus v$ that is not the identity. Thus, $G \setminus v$ is not hyper-extensional and v is a keystone for G . \square

The graph depicted in Fig. 3 proves the following result.

Theorem 6.4 *There exists a non-empty pointed hyper-extensional graph such that none of its nodes is disposable.*

Proof. Let us consider the graph $G_0 = (V_0, E_0)$ depicted in Fig. 3. As already observed above it is hyper-extensional and pointed. Moreover, all of its nodes are keystones by Theorem 5.1 and $G_0 \setminus v$ is weakly connected for all $v \in V_0$. It follows that none of its nodes is disposable. \square

Order	IC	HP	HF ^{1/2}	Only keystone		No disposable	
				nodes	edges	nodes	edges
1	2	2	2(1)	0(0)	0(0)	0(0)	1(1)
2	10	2	2(1)	0(0)	1(1)	0(0)	0(0)
3	104	12	16(2)	0(0)	1(1)	0(0)	0(0)
4	3044	252	504(9)	0(0)	1(1)	0(0)	0(0)
5	291968	18439	52944(88)	0(0)	1(1)	0(0)	0(0)

Table 1

Number of pointed hyper-extensional graphs whose nodes/edges are all keystones and whose nodes/edges are all disposable. The columns labelled as IC, HP, and HF^{1/2} report the number of isomorphic classes of directed graphs (with self-loops), of pointed hyper-extensional graphs, and of hereditarily finite sets (roots of the pointed hyper-extensional graphs), respectively. All the data concerning the well-founded domain are reported in brackets. The number of well-founded hereditarily finite sets (pointed hyper-extensional acyclic graphs) is taken from [19]

In the next section, we prove that, whenever a pointed hyper-extensional graph contains at least one edge, it also has disposable edges.

7 Do disposable edges always exist?

At the begin of this section, we focus on the well-founded part of graphs and then we provide all the ingredients for the general case. First of all, we need to introduce the notion of *rank*.

Definition 7.1 [Rank [6]] Let $G = (V, E)$ be an graph and let v be one of its well-founded nodes. The *rank* of v in G , $rank(G, v)$, is defined as follows:

$$rank(G, v) \stackrel{\text{def}}{=} \begin{cases} 0 & \text{if } v \text{ is a sink} \\ 1 + \max_{(v,u) \in E} (rank(G, u)) & \text{otherwise} \end{cases}$$

Notice that, whenever G is acyclic, all the paths from any node v to a sink are finite in length and $rank(G, v)$ is well-defined.

It is easy to see that, if G is acyclic and pointed, then it has one single root and its rank is greater than those of the other nodes of G .

Lemma 7.2 *If $G = (V, E)$ is a pointed acyclic graph, then it has one single root p and $rank(G, p) > rank(G, q)$ for all $q \in V \setminus \{p\}$.*

Proof. If G had two roots, p_1 and p_2 , then, by definition of root, both p_1 is reachable from p_2 and p_2 is reachable from p_1 . It follows that G is cyclic and this contradicts our hypothesis. Hence, G must have one root p .

Let us assume that there exists $q \neq p$ such that $rank(G, p) \leq rank(G, q)$. Since q is reachable from p , there exists a finite path p_0, p_1, \dots, p_h in G such that $p_0 = q$ and $p_h = p$. Since $p \neq q$, h should be greater than 0. Moreover, by induction on i ,

we can prove that $\text{rank}(G, p_i) \geq \text{rank}(G, q) + i$. It follows that $\text{rank}(G, p)$ should be greater or equal to $\text{rank}(G, q) + h$ where $h \in \mathbb{N} \setminus \{0\}$. However, this contradicts our hypothesis and, thus, it proves that all the nodes in $V \setminus \{p\}$ must have a rank smaller than that of p . \square

The following lemmas show how bisimulations are affected by edge removals and relate the ranks of bisimilar nodes.

Lemma 7.3 *Let G be a graph and let v be a well-founded node of G . Moreover, let G' be the graph $G \setminus (v \ni u)$. For every well-founded node w of G such that $\text{rank}(G, w) < \text{rank}(G, v)$, $w \sim_{G'} w$.*

Proof. We prove the thesis by induction on the rank of w .

rank(G, w) = 0 If $\text{rank}(G, w) = 0$, then w is a sink in G . Hence, it is a sink also in G' and $w \sim_{G'} w$.

rank(G, w) > 0 Let us assume that $q \sim_{G'} q$ for all q in G such that $\text{rank}(G, q) < \text{rank}(G, w)$. From the definition of rank, it follows that $q \sim_{G'} q$ for all $w \ni q$ in G . Since $\text{rank}(G, w) < \text{rank}(G, v)$, $v \neq w$ and $v \ni u \neq w \ni q$ for all edge $w \ni q$ in G . Hence, by definition of bisimulation, $w \sim_{G'} w$.

This concludes the proof. \square

Lemma 7.4 *Let G and G' be two graphs (possibly the same). Moreover, let v and v' be two well-founded nodes in G and G' , respectively. If $v \sim_{G'} v'$, then $\text{rank}(G, v) = \text{rank}(G', v')$.*

Proof. We prove the thesis by induction on the rank of v .

rank(G, v) = 0 If $\text{rank}(G, v) = 0$, then $\emptyset \sim v \sim_{G'} v'$ and $\text{rank}(G', v') = 0$.

rank(G, v) > 0 Let us assume that, for all node w in G such that $\text{rank}(G, w) < \text{rank}(G, v)$, if $w \sim_{G'} w'$, then $\text{rank}(G, w) = \text{rank}(G', w')$. From the definition of bisimulation, if $v \sim_{G'} v'$ then for all $v \ni w$ in G there exists $v' \ni w'$ in G' s.t. $w \sim_{G'} w'$ and vice versa. Furthermore, from the definition of rank, we know that $\text{rank}(G, v) = 1 + \max_{v \ni w} \text{rank}(G, w)$ and $\text{rank}(G', v') = 1 + \max_{v' \ni w'} \text{rank}(G', w')$ in G' . Thus, $\text{rank}(G, v) = 1 + \max_{v \ni w} \text{rank}(G, w) = 1 + \max_{v' \ni w'} \text{rank}(G', w') = \text{rank}(G', v')$.

This concludes the proof. \square

Lemma 7.3 implies that edge removal can collapse only nodes whose ranks are greater or equal to that of the edge source, while from Lemma 7.4 it follows that two nodes that have different ranks are not bisimilar. Since Lemma 7.2 proves that roots have maximum rank in pointed acyclic graphs, removing an edge outgoing from the roots avoids collapses in other nodes.

Definition 7.5 [n -bisimulation [13]] Let $G = (V, E)$ be a graph. We define $\overset{n}{\sim}$ by induction on n as follows:

$$\overset{n}{\sim} \stackrel{\text{def}}{=} \begin{cases} V \times V & \text{if } n = 0 \\ \{(v, w) \mid \forall v \ni v' \exists w \ni w' \text{ s.t. } v' \overset{n-1}{\sim} w' \text{ and vice versa}\} & \text{otherwise} \end{cases}$$

n	$\text{wf}^n(0)$	$\text{wf}^n(1)$	$\text{wf}^n(2)$	$\text{wf}^n(3)$	$\text{wf}^n(4)$
0	\emptyset	$\{0\}$	$\{0, 1\}$	\emptyset	$\{1\}$
1	\emptyset	\emptyset	$\{\emptyset\}$	$\{\{0, 1\}\}$	$\{\emptyset\}$
2	\emptyset	\emptyset	$\{\{\{0, 1\}\}\}$	$\{\{\emptyset\}\}$	$\{\{\{0, 1\}\}\}$
3	\emptyset	\emptyset	$\{\{\{\emptyset\}\}\}$	$\{\{\{\{0, 1\}\}\}\}$	$\{\{\{\emptyset\}\}\}$
4	\emptyset	\emptyset	$\{\{\{\{\{0, 1\}\}\}\}\}$	$\{\{\{\{\emptyset\}\}\}\}$	$\{\{\{\{\{0, 1\}\}\}\}\}$

Table 2
The n -well-founded parts for n upto 4 of the graph of Fig. 1

In the following part of the paper, we write $v \stackrel{n}{\sim} w$ meaning $(v, w) \in \stackrel{n}{\sim}$.

Theorem 7.6 ([8]) *Let G be a finite graph and let x, y be nodes of G . We have $v \sim w$ if and only if for each $n \in \mathbb{N}$ it holds that $v \stackrel{n}{\sim} w$.*

Definition 7.7 [n -well-founded part] *Let G be a graph, v be one of its nodes, and $n \in \mathbb{N}$. The n -well-founded part of v , $\text{wf}^n(v)$, is defined as follows:*

$$\text{wf}^n(v) \stackrel{\text{def}}{=} \begin{cases} \{w \mid v \ni w \text{ and } w \text{ is well-founded}\} & \text{if } n = 0 \\ \{\text{wf}^{n-1}(w) \mid v \ni w \text{ and } w \text{ is non-well-founded}\} & \text{otherwise} \end{cases}$$

Notice that $\text{wf}^n(v)$ can be represented as a graph. In particular, this can be computed taking the unravelling of G of order n starting from v and using only the non-well-founded nodes reachable from x . Then the well-founded trees are attached to the leaves.

Notice also that if v is well-founded, then $\text{wf}^0(v) \sim v$, while for each $n > 0$ it holds $\text{wf}^n(v) = \emptyset$.

Example 7.8 Let us consider the graph depicted in Fig. 1. The succession of the n -well-founded parts of its nodes for n up to 4 are reported in Table 2.

Lemma 7.9 *Let G be a graph. If $\text{wf}^n(v) \sim \text{wf}^n(w)$ for each $n \in \mathbb{N}$, then $v \stackrel{m}{\sim} w$ for each $m \in \mathbb{N}$.*

Proof. By counter nominal we prove that if there exists m such that $v \not\stackrel{m}{\sim} w$, then there exists n such that $\text{wf}^n(v) \not\sim \text{wf}^n(w)$. Let m be the minimum such that $v \not\stackrel{m}{\sim} w$. We proceed by induction on m . Since $v \stackrel{0}{\sim} w$ holds by definition of $\stackrel{0}{\sim}$, m should be greater or equal to 1.

Base: $m = 1$. If there existed $v \ni v'$ and $w \ni w'$, then $v \stackrel{1}{\sim} w$ should hold because $v' \stackrel{0}{\sim} w'$ by definition of $\stackrel{0}{\sim}$. Hence, one between v and w should have no successors. We assume with no restriction that $v \ni v'$ and w has no successors. If v' is well-founded, then $\text{wf}^0(v) \neq \emptyset$, $\text{wf}^0(w) = \emptyset$ and $\text{wf}^0(v) \not\sim \text{wf}^0(w)$ hold. If v' is non-well-founded, then $\text{wf}^0(v') \in \text{wf}^1(v)$. However, w has no successors and, hence, $\text{wf}^n(w) = \emptyset$ for all $n \in \mathbb{N}$. It follows that $\text{wf}^1(v) \not\sim \text{wf}^1(w)$.

Inductive Step: $m > 1$. By hypothesis, $v \not\sim^m w$. Hence, we can safely assume that there exists $v \ni v'$ such that $v' \not\sim^{m-1} w'$ for each $w \ni w'$. If v' is well-founded, then $v' \in \text{wf}^0(v)$ and for each $w' \in \text{wf}^0(w)$ $v' \not\sim w'$. Hence $\text{wf}^0(v) \not\sim \text{wf}^0(w)$. If v' is non-well-founded, then, by inductive hypothesis on $m - 1$, we have that there exists n such that $\text{wf}^n(v') \not\sim \text{wf}^n(w')$ for each $w \ni w'$. Since $\text{wf}^n(v') \in \text{wf}^{n+1}(v)$, we conclude that $\text{wf}^{n+1}(v) \not\sim \text{wf}^{n+1}(w)$. \square

Theorem 7.10 *Let G be a graph and v and w be two of its nodes. Then $\text{wf}^n(v) \sim \text{wf}^n(w)$ for all $n \in \mathbb{N}$ if and only if $v \sim w$.*

Proof. We separately prove that if $\text{wf}^n(v) \sim \text{wf}^n(w)$ for all $n \in \mathbb{N}$, then $v \sim w$ (\Rightarrow) and that if $v \sim w$, then $\text{wf}^n(v) \sim \text{wf}^n(w)$ for all $n \in \mathbb{N}$ (\Leftarrow).

\Rightarrow) By Lemma 7.9 we have that $v \stackrel{m}{\sim} w$ holds for each $m \in \mathbb{N}$. Hence, the thesis follows from Theorem 7.6.

\Leftarrow) Let $\mathcal{S}_n = \{(\text{wf}^i(x), \text{wf}^i(y)) \mid i \leq n \text{ and } x, y \in V \text{ are bisimilar}\}$. By induction on n we prove that \mathcal{S}_n is a bisimulation up-to \sim .

Base: $n = 0$. Let $(\text{wf}^0(x), \text{wf}^0(y)) \in \mathcal{S}_0$. If $\text{wf}^0(x) \ni x'$, then x' is well-founded and $x \ni x'$ is in G . Since x and y are bisimilar, there exists an edge $y \ni y'$ such that $x' \sim y'$. This proves the claim for $n = 0$.

Inductive Step: $n > 0$. Let $(\text{wf}^j(x), \text{wf}^j(y)) \in \mathcal{S}_n$. If $j < n$, then \mathcal{S}_j is a bisimulation-up-to \sim by inductive hypothesis. Moreover, \mathcal{S}_j is included in \mathcal{S}_n by definition of \mathcal{S}_n . This proves the thesis for $j < n$. If, otherwise, $j = n$ and $x' \in \text{wf}^n(x)$, there exist $x \ni x''$ in G such that $x' = \text{wf}^{n-1}(x'')$ and x'' non-well-founded. From $x \sim y$, it follows that there exists $y \ni y'$ in G and $x'' \sim y'$. Since x'' is non-well-founded, y' has to be non-well-founded. Hence, $\text{wf}^{n-1}(y') \in \text{wf}^n(y)$ and $(\text{wf}^{n-1}(x''), \text{wf}^{n-1}(y')) \in \mathcal{S}_n$.

This concludes the proof. \square

The Ackermann code is a bijective function that maps well-founded hereditarily finite sets into Natural numbers [1]. Classically, it is defined as:

$$\text{Ack}(S) \stackrel{\text{def}}{=} \sum_{p \in S} 2^{\text{Ack}(p)}.$$

The following definition extends Ackermann code and provides an encoding for the well-founded nodes of any –possibly, neither hyper-extensional nor non-well-founded– graph.

Definition 7.11 [Ackermann code] Let G be a graph. The *Ackermann code* $\mathcal{A}(G, x)$ of any well-founded node x of G is defined as:

$$\mathcal{A}(G, x) \stackrel{\text{def}}{=} \sum_{k \in \mathcal{S}_{\mathcal{A}}(G, x)} 2^k \quad (3)$$

where $\mathcal{S}_{\mathcal{A}}(G, x) = \{\mathcal{A}(G, x') \mid x \ni x' \text{ in } G\}$

It is easy to see that, if S is a well-founded hereditarily finite sets, p_S is the corresponding node in the membership graph \mathcal{G}_S , and p is a node of a graph G such that p and p_S are bisimilar, then $\mathcal{A}(\mathcal{G}_S, p_S) = \mathcal{A}(G, p) = \text{Ack}(S)$.

Corollary 7.12 *Let G be a graph. There exists $n \in \mathbb{N}$ such that $\mathcal{A}(G, \text{wf}^n(x)) \neq \mathcal{A}(G, \text{wf}^n(y))$ if and only if $x \not\sim y$.*

From now on we use the notation $\text{Ack}^n(x)$ to denote $\text{Ack}(\text{wf}^n(x))$.

Lemma 7.13 *Let G be a graph and $x \in G$. It holds that*

$$\mathcal{A}^0(G, x) = \sum_{k \in \mathcal{S}_{\mathcal{A}}^0(G, x)} 2^k \quad (4)$$

where $\mathcal{S}_{\mathcal{A}}^0(G, x) = \{\mathcal{A}(G, x') \mid x \ni x' \text{ in } G \text{ and } x' \text{ is well-founded}\}$. Moreover,

$$\mathcal{A}^n(G, x) = \sum_{k \in \mathcal{S}_{\mathcal{A}}^n(G, x)} 2^k \quad (5)$$

where $\mathcal{S}_{\mathcal{A}}^n(G, x) = \{\mathcal{A}^{n-1}(G, x') \mid x \ni x' \text{ in } G \text{ and } x' \text{ is non-well-founded}\}$.

Proof. This is an immediate consequence of the definitions. \square

Lemma 7.14 *Let G be a graph, let x, x' be two non-well-founded nodes such that $x \ni x'$ in G , and let G' be $G \setminus (x \ni x')$. If x is non-well-founded even in G' , then:*

- (a) $\mathcal{A}^n(G', y) \leq \mathcal{A}^n(G, y)$ for each node y of G and for all $n \in \mathbb{N}$;
- (b) either $\mathcal{A}^n(G', y) = \mathcal{A}^n(G, y)$ for each node y of G and for all $n \in \mathbb{N}$ or, let $m \in \mathbb{N}$ be the minimum such that there exists z with $\mathcal{A}^m(G', z) < \mathcal{A}^m(G, z)$, both $z = x$ and $\mathcal{A}^m(G', y) = \mathcal{A}^m(G, y)$ for all $y \neq x$ hold.

Proof. Item (a) can be proved by induction on n .

As far as item (b) is concerned, we first notice that, since x is non-well-founded in G' , all the non-well-founded nodes in G remain non-well-founded in G' . Hence, since both x and x' are non-well-founded in G , $\mathcal{A}^0(G', z) = \mathcal{A}^0(G, z)$ for all nodes z of G by the definition of $\mathcal{A}^0(G, \cdot)$. Let $m \in \mathbb{N}$ be the minimum such that there exists z with $\mathcal{A}^m(G', z) < \mathcal{A}^m(G, z)$. By above considerations, $m > 0$.

By Lemma 7.13, we have that

$$\mathcal{A}^m(G', y) = \sum_{k \in \mathcal{S}_{\mathcal{A}}^m(G', y)} 2^k$$

where $\mathcal{S}_{\mathcal{A}}^m(G', y) = \{\mathcal{S}_{\mathcal{A}}^{m-1}(G', y') \mid y \ni y' \text{ and } y' \text{ is non-well-founded}\}$ because $m > 0$. Since G' contains all the edges of G , but $x \ni x'$, every edge $y \ni y'$ in G is also an edge for G' if $y \neq x$. Thus, if $y \neq x$ and m is the minimum such that $\mathcal{A}^m(G', z) < \mathcal{A}^m(G, z)$, we get that $\mathcal{A}^{m-1}(G', y') = \mathcal{A}^{m-1}(G, y')$ for all y' in G and $\mathcal{S}_{\mathcal{A}}^m(G', y) = \{\mathcal{A}^{m-1}(G, y') \mid y \ni y' \text{ and } y' \text{ is non-well-founded}\} = \mathcal{S}_{\mathcal{A}}^m(G, y)$. Hence, if $y \neq x$, then,

$$\mathcal{A}^m(G', y) = \sum_{k \in \mathcal{S}_{\mathcal{A}}^m(G, y)} 2^k = \mathcal{A}^m(G, y),$$

and x is the only node of G such that $\mathcal{A}^m(G', x) < \mathcal{A}^m(G, x)$. \square

Lemma 7.15 *Let G be a hyper-extensional graph and let G' be the graph $G \setminus (x \ni x')$. Either $x \ni x'$ is a disposable edge or there exists y in G' such that $x \stackrel{G'}{\sim} y$.*

Proof. Let us assume that $x \ni x'$ is not disposable. We prove that if the maximum bisimulation \mathcal{S} over G' is such that $Id \subset \mathcal{S}$ (meaning that \mathcal{S} and identity do not coincide) and the only pair including x in \mathcal{S} is (x, x) , then \mathcal{S} is a bisimulation over G . Let $(y, z) \in \mathcal{S}$. If $(y, z) = (x, x)$, then it is trivial to prove that the conditions required to a bisimulation hold. If $(y, z) \neq (x, x)$, then let $y \ni y'$ in G . Since, $y \neq x$ we have $y \ni y'$ in G' . Hence, there exists z' such that $z \ni z'$ in G' and $(y', z') \in \mathcal{S}$. Since $z \neq x$, we get that $z \ni z'$ belongs to G and $(y', z') \in \mathcal{S}$. \square

Thanks to Lemmas 7.14 and 7.15, we are now ready to prove the following theorem.

Theorem 7.16 *If G is a pointed hyper-extensional graph that has at least one edge, then G has a disposable edge $v \ni w$.*

Proof. Sketch Let $H = \{x_1, \dots, x_k\}$ be the set of nodes of G such that $\mathcal{A}^0(G, x_i)$ is maximum in G for each $i = 1, \dots, k$.

If there exists i such that x_i is well-founded, then $k = 1$ and x_1 is the root of G . In this case, it is easy to prove that there exists an edge $x_1 \ni y$ that is disposable.

Let us consider the case in which all the elements of H are non-well-founded. We can subsequently refine H into H_1, H_2, \dots by minimizing $\mathcal{A}^1(G, \cdot), \mathcal{A}^2(G, \cdot), \dots$. By Corollary 7.12, there exists a minimum j such that $H_j = \{x\}$. Let $x \ni y$ be an edge of G with y non-well-founded. By Lemmas 7.14 and 7.15 we can prove that $x \ni y$ is disposable. \square

8 Conclusions and Future Work

In this paper we considered the problem of removing parts from an hereditarily finite set without causing bisimulation collapses. We exploited a SAGE pipeline that we implemented to prove that the problem has a negative answer if one is interested in removing a node. On the other hand, we gave a positive answer in the case of removal of one edge, provided that only nodes in the same weakly connected components are compared.

The results in [16] are based on a Markov chain Monte Carlo-based algorithm, initially proposed for generating acyclic digraphs [11,12]. The key fact needed in order to show that the Markov chain converges to the uniform distribution were the irreducibility, aperiodicity, and symmetry of the chain. The idea exploited in the construction of the Markov chain was to show that a pair of elementary operations on graphs (implemented as basic transition rules of the Markov chain, akin to the elimination of a node) could be used to transform any graph G into another graph G_0 within the same family. A positive answer to the one-node elimination problem would have allowed to immediately extend such results to non-well-founded sets, getting a random generator for $\text{HF}^{1/2}$.

The positive result we obtained on edges cannot be easily exploited in this direction and further investigations are required.

References

- [1] Ackermann, W., *Die widerspruchsfreiheit der allgemeinen mengenlehre*, *Mathematische Annalen* **114** (1937), pp. 305–315.
URL <http://dx.doi.org/10.1007/BF01594179>
- [2] Aczel, P., “Non-Well-Founded Sets,” Number 14 in *Lecture Notes*, Center for the Study of Language and Information, Stanford University, 1988.
- [3] Bender, E., L. Richmond, R. Robinson and N. Wormald, *The asymptotic number of acyclic digraphs, I*, *Combinatorica* **6** (1986), pp. 15–22.
- [4] Bender, E. A. and R. W. Robinson, *The asymptotic number of acyclic digraphs, II*, *Journal of Combinatorial Theory, Series B* **44** (1988), pp. 363 – 369.
- [5] David, R. L., *The number of structures of finite relations*, *Proc. Amer. Math. Soc.* **4** (1953), pp. 486–495.
- [6] Dovier, A., C. Piazza and A. Policriti, *An efficient algorithm for computing bisimulation equivalence*, *Theor. Comput. Sci.* **311** (2004), pp. 221–256.
- [7] Harary, F. and E. M. Palmer, “Graphical enumeration,” Addison-Wesley, 1973.
- [8] Hennessy, M. and R. Milner, *Algebraic laws for nondeterminism and concurrency*, *J. ACM* **32** (1985), pp. 137–161.
URL <http://doi.acm.org/10.1145/2455.2460>
- [9] Kanellakis, P. C. and S. A. Smolka, *CCS expressions, finite state processes, and three problems of equivalence*, in: *Proceedings of the Second Annual ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing, Montreal, Quebec, Canada, August 17-19, 1983* (1983), pp. 228–240.
- [10] McIlroy, M. D., *Calculation of numbers of structures of relations on finite sets*, in: *Quarterly Progress Reports No. 17*, MIT Research Laboratory of Electronics, 1955 .
- [11] Melançon, G., I. Dutour and M. Bousquet-Mélou, *Random generation of directed acyclic graphs*, in: J. Neseřtil, M. Noy and O. Serra, editors, *Comb01, Euroconference on Combinatorics, Graph Theory and Applications*, *Electronic Notes in Discrete Mathematics* **10**, 2001, pp. 202–207.
- [12] Melançon, G. and F. Philippe, *Generating connected acyclic digraphs uniformly at random*, *Information Processing Letters* (2004), pp. 209–213.
- [13] Milner, R., “A Calculus of Communicating Systems,” *Lecture Notes in Computer Science* **92**, Springer, 1980.
URL <http://dx.doi.org/10.1007/3-540-10235-3>
- [14] Omodeo, E. G., C. Piazza, A. Policriti and A. I. Tomescu, *Hyper-extensionality and one-node elimination on membership graphs*, in: *Proceedings of the 29th Italian Conference on Computational Logic (CILC 2014)*, *CEUR Workshop Proceedings* **1195** (2014), pp. 341–346.
- [15] Paige, R. and R. E. Tarjan, *Three Partition Refinement Algorithms*, *SIAM J. Comput.* **16** (1987), pp. 973–989.
- [16] Policriti, A. and A. I. Tomescu, *Markov chain algorithms for generating sets uniformly at random*, *Ars Mathematica Contemporanea* **6** (2013), pp. 57–68.
- [17] Redfield, J., *The theory of group-reduced distributions*, *American Journal of Mathematics* **49** (1927), pp. 433–455.
- [18] Sloane, N., *The on-line encyclopedia of integer sequences, published electronically at <http://oeis.org>, 2010, Sequence A000273.*
- [19] Sloane, N., *The on-line encyclopedia of integer sequences, published electronically at <http://oeis.org>, 2010, Sequence A001192.*
- [20] Stein, W. et al., “Sage Mathematics Software (Version 6.5),” The Sage Development Team (2015), <http://www.sagemath.org>.
- [21] Wagner, S., *Asymptotic enumeration of extensional acyclic digraphs*, *Algorithmica* **66** (2013), pp. 829–847.
URL <http://dx.doi.org/10.1007/s00453-012-9725-4>