Strictness of the Collapsible Pushdown Hierarchy*

Alexander Kartzow[1] and Paweł Parys[2]

[1] Universität Leipzig
Johannisgasse 26, 04103 Leipzig, Germany
kartzow@informatik.uni-leipzig.de
[2] University of Warsaw
ul. Banacha 2, 02-097 Warszawa, Poland
parys@mimuw.edu.pl

Abstract. We present a pumping lemma for each level of the collapsible pushdown graph hierarchy in analogy to the second author's pumping lemma for higher-order pushdown graphs (without collapse). Using this lemma, we give the first known examples that separate the levels of the collapsible pushdown graph hierarchy and of the collapsible pushdown tree hierarchy, i.e., the hierarchy of trees generated by higher-order recursion schemes. This confirms the open conjecture that higher orders allow to generate more graphs and more trees.

1 Introduction

Already in the 70's, Maslov ([11,12]) generalised the concept of a pushdown systems to higher-order pushdown systems and studied such devices as acceptors of string languages. In the last decade, renewed interest in these systems has arisen. They are now studied as generators of graphs and trees. Knapik et al. [10] showed that the class of trees generated by deterministic level n pushdown systems coincides with the class of trees generated by *safe* level n recursion schemes,[3] and Caucal [5] gave another characterisation: trees on level $n+1$ are obtained from trees on level n by an MSO-interpretation followed by an unfolding. Carayol and Wöhrle [4] studied the ε-contractions of configuration graphs of level n pushdown systems and proved that these are exactly the graphs in the n-th level of the Caucal hierarchy.

Driven by the question whether safety implies a semantical restriction to recursion schemes, Hague et al. [7] extended the model of higher-order pushdown systems by introducing a new stack operation called collapse. They showed that the trees generated by the resulting collapsible pushdown systems coincide exactly with the class of trees generated by all higher-order recursion schemes and

* The first author is supported by the DFG project "GELO". The second author is partially supported by the Polish Ministry of Science grant nr N N206 567840. The collaboration of the authors is supported by the ESF project "Games for Design and Verification".

[3] Safety is a syntactic restriction on the recursion scheme.

this correspondence is level-by-level. Recently, Parys ([13,14]) proved the safety conjecture, i.e., he showed that higher-order recursion schemes generate more trees than safe higher-order recursion schemes, which implies that the class of collapsible pushdown trees is a proper extension of the class of higher-order pushdown trees. Similarly, due to their different behaviour with respect to MSO model checking, we know that the class of collapsible pushdown graphs forms a proper extension of the class of higher-order pushdown graphs.

Several questions concerning the relationship of these classes have been left open so far. Up to now it was not known whether collapsible pushdown graphs form a strict hierarchy in the sense that for each $n \in \mathbb{N}$ the class of level n collapsible pushdown graphs is strictly contained in the class of level $(n+1)$ collapsible pushdown graphs. The same question was open for the hierarchy of trees generated by collapsible pushdown systems (i.e. by recursion schemes). Extending the pumping arguments of Parys for higher-order pushdown systems [15] to the collapsible pushdown setting, we answer both questions affirmative.

Our main technical contribution is the following *pumping lemma* for collapsible pushdown systems. It subsumes the known pumping lemmas for level 2 collapsible pushdown systems [9] and for higher-order pushdown systems [15]. Set $\exp_0(i) = i$ and $\exp_{k+1}(i) = 2^{\exp_k(i)}$.

Theorem 1.1. *Let $\mathcal{S}$ be a collapsible pushdown system of level n. Let $\mathcal{G}$ be the ε-contraction of the configuration graph of $\mathcal{S}$. Assume that it is finitely branching and that there is a path of length m from the initial configuration to some configuration c. For $C_{\mathcal{S}}$ a constant only depending on $\mathcal{S}$, if there is a path p in $\mathcal{G}$ of length at least $\exp_{n-1}((m+1) \cdot C_{\mathcal{S}})$ which starts in c, then there are infinitely many paths in $\mathcal{G}$ which start in c and end in configurations having the same state as the last configuration of p.*

Corollary 1.2. *Let $\mathcal{G}$ be the successor tree induced by $\{1^i 0^{\exp_n(i)} \mid i \in \mathbb{N}\}$.*

$\mathcal{G}$ is the ε-contraction of the configuration graph of a pushdown system of level $n+1$ but not the ε-contraction of the configuration graph of any collapsible pushdown system of level n. Moreover, $\mathcal{G}$ is generated by a safe level $(n+1)$ recursion scheme but not by any level n recursion scheme.

$\mathcal{G}$ is not in the n-th level because application of the pumping lemma to the node $1^{2 \cdot C_{\mathcal{S}}}0$ yields a contradiction. The proof that $\mathcal{G}$ is in level $n+1$ follows from [2]. Beside this main result, our techniques allow us to decide the following problems.[4]

Lemma 1.3. *Given a collapsible pushdown system, it is decidable*

1. *whether the ε-contraction of its configuration graph is finitely branching,*
2. *whether the ε-contraction of its configuration graph is finite, and*
3. *whether the unfolding of the ε-contraction of its configuration graph is finite.*

[4] We thank several anonymous referees of our LICS submissions for pointing our interest towards these problems.

1.1 Related Work

Hayashi [8] and Gilman [6] proved a pumping and a shrinking lemma for indexed languages. It is shown in [1] that indexed languages are exactly the string languages accepted by level 2 collapsible pushdown systems. For higher levels, no shrinking techniques are known so far. Since our pumping lemma can be used only for finitely branching systems, it cannot be used to show that certain string languages do not occur on certain levels of the (collapsible) higher-order pushdown hierarchy. Note that we do not know whether the string languages accepted by nondeterministic level n pushdown systems and by nondeterministic level n collapsible pushdown systems coincide for $n > 2$. Thus, it is an interesting open question whether there is a stronger pumping lemma for runs of higher-order systems that could be used to separate these classes of string languages.

2 Collapsible Pushdown Graphs

Collapsible pushdown systems of level n (from now on $n \in \mathbb{N}$ is fixed) are an extension of pushdown systems where we replace the stack by an n-fold nested stack structure. This higher-order stack is manipulated using a a push, a pop and a collapse operation for each stack level $1 \leq i \leq n$. When a new symbol is pushed onto the stack, we attach a copy of a certain level k substack of the current stack to this symbol (for some $1 \leq k \leq n$) and at some later point the collapse operation may replace the topmost level k stack with the level k stack stored in the topmost symbol of the stack (we also talk about the linked k-stack of the topmost symbol). In some weak sense the collapse operation allows to jump back to the (level k) stack where the current topmost symbol was created for the first time.

Definition 2.1. Given a number n (the level of the system) and stack alphabet Γ, we define the set of stacks as the smallest set satisfying the following.

- If $s_1, s_2, \ldots, s_m$ are $(k-1)$-stacks, where $1 \leq k \leq n$, then the sequence $[s_1, s_2, \ldots, s_m]$ is a k-stack. This includes the empty sequence ($m = 0$).
- If s^k is a k-stack, where $1 \leq k \leq n$, and $\gamma \in \Gamma$, then (γ, k, s^k) is a 0-stack.

For a 0-stack $s^0 = (\gamma, k, t^k)$ we call γ the *symbol* of s^0 and for some k-stack t^k the *topmost symbol* is the symbol of its topmost 0-stack.

For a k-stack s^k and a $(k-1)$-stack s^{k-1} we write $s^k : s^{k-1}$ to denote the k-stack obtained by appending s^{k-1} on top of s^k. We write $s^2 : s^1 : s^0$ for $s^2 : (s^1 : s^0)$.

Let us remark that in the original definition stacks are defined differently: they are not nested, a 0-stack does not store the linked k-stack but the number of pop-operations a collapse is equivalent to. This is only a syntactical difference as discussed in Appendix A. Independently, Broadbent et al. recently also introduced our definition of stack under the name *annotated stacks* in [3].

Definition 2.2. We define the set of stack operations OP as follows. We decompose a stack s of level n into its topmost stacks $s^n : s^{n-1} : \cdots : s^0$. We have $\mathsf{pop}^i(s) := s^n : \cdots : s^{i+1} : s^i$ for all $1 \leq i \leq n$. The result is undefined if s^i is empty. For $2 \leq i \leq n$ we have $\mathsf{push}^i(s) := s^n : \cdots : s^{i+1} : (s^i : \cdots : s^0) : s^{i-1} : \cdots : s^0$. The level 1 push is $\mathsf{push}^1_{\gamma,k}$ for $\gamma \in \Gamma$, $1 \leq k \leq n$ which is defined by $\mathsf{push}^1_{\gamma,k}(s) := s^n : \cdots : s^2 : (s^1 : s^0) : (\gamma, k, s^k)$.[5] The collapse operation col^i (where $1 \leq i \leq n$) is defined if the topmost 0-stack is (γ, i, t^i), and t^i is not empty. Then it is $\mathsf{col}^i(s) := s^n : \cdots : s^{i+1} : t^i$. Otherwise the collapse operation is undefined.

Definition 2.3. The *initial* 0*-stack* $\perp_0$ is $(\perp, n, [])$ for a special symbol $\perp \in \Gamma$, i.e., a 0-stack only containing the symbol $\perp$ with link to the empty stack. The *initial* $(k+1)$*-stack* is $[\perp_k]$. Some n-stack s is a *pushdown store* (or *pds*), if there is a finite sequence of stack operations that create s from $\perp_n$.

Remark 2.4. If s is a pds and if $\mathsf{col}^j(s)$ is defined, then there is a $k \geq 1$ such that $\mathsf{col}^j(s)$ is the stack obtained from s by applying pop^j k times.

Definition 2.5. A *collapsible pushdown system of level* n (an n-CPS) is a tuple $\mathcal{S} = (\Gamma, A, Q, q_I, \perp, \Delta)$ where Γ is a finite stack alphabet containing the special symbol $\perp$, A is a finite input alphabet, Q is a finite set of states, $q_I \in Q$ is an initial state, and $\Delta \subseteq Q \times \Gamma \times (A \cup \{\varepsilon\}) \times Q \times OP$ is a transition relation.
A *configuration* is a pair (q, s) with $q \in Q$ and s a pds. The *initial configuration of* $\mathcal{S}$ is $(q_I, \perp_n)$.

Definition 2.6. We define a *run* of a CPS $\mathcal{S}$. For $0 \leq i \leq m$, let $c_i = (q_i, s_i)$ be a configuration of $\mathcal{S}$ and let γ_i denote the topmost stack symbol of s_i. A run R of length m from c_0 to c_m is a sequence $c_0 \vdash^{a_1} c_1 \vdash^{a_2} \cdots \vdash^{a_m} c_m$ such that, for $1 \leq i \leq m$, there is a transition $(q_{i-1}, \gamma_{i-1}, a_i, q_i, op)$ where $s_i = op(s_{i-1})$. We set $R(i) := c_i$ and call $|R| := m$ the *length of* R. The *subrun* $R{\restriction}_{i,j}$ is $c_i \vdash^{a_{i+1}} c_{i+1} \vdash^{a_{i+2}} \cdots \vdash^{a_j} c_j$. For runs R, S with $R(|R|) = S(0)$, we write $R \circ S$ for the *composition* of R and S which is defined as expected.

Definition 2.7. Let $\mathcal{S}$ be a collapsible pushdown system. The *(collapsible pushdown) graph*[6] of $\mathcal{S} = (\Gamma, A, Q, q_I, \perp, \Delta)$ is $\mathcal{G} := (G, (E_a)_{a \in A \cup \{\varepsilon\}})$ where G consists of all configurations reachable from $(q_0, \perp_n)$ and there is an a-labelled edge from a configuration c to a configuration d if there is a run $c \vdash^a d$. The ε*-contraction* of $\mathcal{G}$ is the graph $(G', (E'_a)_{a \in A})$ where $G' := \{c \in G : \exists d \in G\ d \vdash^a c$ for some $a \in A\}$ and two configurations c, d are connected by E'_a if there is a run $c \vdash^\varepsilon c_1 \vdash^\varepsilon \cdots \vdash^\varepsilon c_n \vdash^a d$ for some $n \in \mathbb{N}$.

3 Proof Structure

The proof of the pumping lemma consists of three parts. In the first part we introduce a special kind of context free grammars (called well-formed grammars)

[5] In the following, we write push^1 whenever we mean some $\mathsf{push}^1_{\gamma,k}$ operation where the values of γ and k do not matter for the argument.

[6] In fact it is an edge-labelled graph; sets E_a need not to be disjoint.

for runs of a collapsible pushdown system $\mathcal{S}$. In such a grammar, each nonterminal represents a set of runs and each terminal is one of the transitions of $\mathcal{S}$. Let X and $X_1, \dots, X_m$ be sets of runs and δ some transition. A rule $X \supseteq \delta X_1 X_2 \dots X_m$ describes a run R if $R = S \circ T_1 \circ T_2 \circ \dots \circ T_n$ such that S performs only the transition δ and $T_i \in X_i$. A grammar describes a family $\mathcal{X}$ if the rules for each $X \in \mathcal{X}$ describe exactly the runs in X. Well-formed grammars are syntactically restricted in order to obtain the following result. If $\mathcal{X}$ is a finite family described by a well-formed grammar, we can define

1. a function $\mathsf{ctype}_{\mathcal{X}}$ from configurations of $\mathcal{S}$ to a finite partial order $(\mathcal{T}_{\mathcal{S}}, \sqsubseteq)$ (of *types of configurations*), and
2. for each $X \in \mathcal{X}$ a level $\mathsf{lev}(X) \in \{0, 1, \dots, n\}$

such that the following transfer property of runs holds.

Theorem 3.1. *Let $\mathcal{X}$ be a family of sets of runs described by a well-formed grammar, $R \in X \in \mathcal{X}$, and c be a configuration with $\mathsf{ctype}_{\mathcal{X}}(R(0)) \sqsubseteq \mathsf{ctype}_{\mathcal{X}}(c)$.*

1. *There is a run $S \in X$ starting in c which has the same final state as R and*
2. *if $\mathsf{lev}(X) = 0$, then $\mathsf{ctype}_{\mathcal{X}}(R(|R|)) \sqsubseteq \mathsf{ctype}_{\mathcal{X}}(S(|S|))$.*

The idea behind the definition of $\mathsf{ctype}_{\mathcal{X}}$ is that we assign a type not only to the whole configuration, but also to every k-stack (for every k). This type summarises possible behaviours of the k-stack in dependence on the type of the n-stack below this k-stack. This makes types compositive: the type of a stack $s^{k+1} : s^k$ is determined by the type of s^{k+1} and of s^k. The above theorem generalises results of [15] in two ways: first, it works for collapsible systems; second, it works for arbitrary well-formed grammars instead of a fixed family of set of runs. The corresponding part of the proof from [15] is not transferable to collapsible systems at all. For collapsible systems we even need a new definition of types (see Appendix B). We stress that new definition of types rely on the different form of representing links in stacks: our k-stack already contains all linked stacks, so we can summarise it using a type from a finite set. On the other hand the original k-stack has arbitrarily many numbers pointing to stacks "outside", and we could not define a type from a finite set because the behaviour of a k-stack would depend on this unbounded context "outside".

In the second part of the proof (cf. Section 5), we introduce a well-formed grammar for a certain family $\mathcal{X}$. As a main feature, $\mathcal{X}$ contains the sets of so-called *pumping runs* $\mathcal{P}$. In the grammar describing $\mathcal{X}$, the level of $\mathcal{P}$ is 0 whence the strong version of Theorem 3.1 applies. If a pumping run R starts and ends in configurations of the same type, this theorem then allows to pump this run, i.e., basically we can append a copy of this run to its end and iterating this process we obtain infinitely many pumping runs.

The last part of the proof uses Theorem 3.1 for the above family $\mathcal{X}$ to deduce the pumping lemma. This part follows closely the analogous proof for the non-collapsible pushdown systems in [15] (see Appendices F–H): we prove that a long run contains a pumping run such that the application of Theorem 3.1 yields a configuration c on this path such that either the graph is infinitely branching at c or the pumped runs yield longer and longer paths in the ε-contraction of the pushdown graph.

4 Run Grammars

Let $\mathcal{X}$ be a finite family whose elements are sets of runs of $\mathcal{S}$. We want to describe this family using a kind of context free grammar. In this grammar the members of $\mathcal{X}$ appear as nonterminals and the transitions of $\mathcal{S}$ play the role of terminals.

We assume that there is a partition $\mathcal{X} = \bigcup_{i=0}^{n} \mathcal{X}_i$ into pairwise distinct families of sets of runs. For each set $X \in \mathcal{X}$, we define its level to be $\mathsf{lev}(X) := i$ if $X \in \mathcal{X}_i$. We only consider *well-formed grammars* that satisfy the restriction that all rules of the grammar have to be *well-formed*.

Definition 4.1. A *well-formed rule over* $\mathcal{X}$ (*wf-rule* for short) is of the form

1. $X \supseteq$ where $X \in \mathcal{X}$, or
2. $X \supseteq \delta$ where $\delta \in \Delta$, $X \in \mathcal{X}$ and if the operation in δ is pop^k or col^k then $k \leq \mathsf{lev}(X)$, or
3. $X \supseteq \delta Y$ where $\delta \in \Delta$, $X, Y \in \mathcal{X}$, $\mathsf{lev}(Y) \leq \mathsf{lev}(X)$ and if the operation in δ is pop^k or col^k then $k \leq \mathsf{lev}(Y)$, or
4. $X \supseteq \delta Y Z$ where $\delta \in \Delta$, $X, Y, Z \in \mathcal{X}$, $\mathsf{lev}(Z) \leq \mathsf{lev}(X)$, if the operation in δ is pop^k or col^k then $k \leq \mathsf{lev}(Y)$, and whenever R is a composition of a one-step run performing transition δ with a run from Y, then the topmost $\mathsf{lev}(Y)$-stacks of $R(0)$ and $R(|R|)$ coincide.

Definition 4.2. We say that a run R is *described* by a wf-rule $X \supseteq \delta X_1 \dots X_m$, $m \in \{0, 1, 2\}$ if there is a decomposition $R = R_0 \circ R_1 \circ \dots \circ R_m$ such that R_0 has length 1 and performs δ and $R_i \in X_i$ for each $1 \leq i \leq m$; a run R is described by $X \supseteq$ if $|R| = 0$. We say that a family $\mathcal{X}$ is *described* by a well-formed grammar $\mathcal{R}_{\mathcal{X}}$ if for each $X \in \mathcal{X}$, a run R is in X if and only if it is described by some rule $X \supseteq \delta X_1 \dots X_m \in \mathcal{R}_{\mathcal{X}}$.

Example 4.3. Let $\mathcal{Q}$ be the set of all runs. Setting $\mathsf{lev}(\mathcal{Q}) = n$, the one-element family $\{\mathcal{Q}\}$ is described by the wf-rules $\mathcal{Q} \supseteq \delta \mathcal{Q}$ for each transition δ, and $\mathcal{Q} \supseteq$.

Indeed, for every run R either $|R| = 0$ or R consists of a first transition followed by some run. Note that we cannot choose $\mathsf{lev}(\mathcal{Q})$ different from n whenever $\mathcal{S}$ contains a transition δ_0 performing col^n or pop^n. If we set $\mathsf{lev}(\mathcal{Q}) < n$, then $\mathcal{Q} \supseteq \delta_0 \mathcal{Q}$ would not be a wf-rule.

Next we prove that the class of families described by well-formed grammars is closed under addition of unions and compositions. This is crucial for the decidability results mentioned in Lemma 1.3. If X and Y are sets of runs, we set $X \circ Y := \{R \circ S : R \in X, S \in Y\}$.

Lemma 4.4. *Let $\mathcal{X}$ be a family described by a well-formed grammar. For $X, Y \in \mathcal{X}$ the family $\mathcal{X} \cup \{X \cup Y\}$ is described by a well-formed grammar. Moreover, there is a family $\mathcal{Y} \supseteq \mathcal{X} \cup \{X \circ Y\}$ that is described by a well-formed grammar. In these grammars, we have* $\mathsf{lev}(X \cup Y) = \mathsf{lev}(X \circ Y) = \max(\mathsf{lev}(X), \mathsf{lev}(Y))$.

Proof. For each rule $Z \supseteq \delta Z_1 \dots Z_m$ with $Z \in \{X, Y\}$ adding the rule $(X \cup Y) \supseteq \delta Z_1 \dots Z_m$ settles the case of unions.

For the composition, we add a set $Z \circ Y$ for each $Z \in \mathcal{X}$, and a new set Y^i for $0 \leq i \leq n$. Y^i contains exactly the same runs as Y, but we set $\mathsf{lev}(Y^i) := \max(\mathsf{lev}(Y), i)$. Wf-rules describing Y^i are clearly obtained from the rules for Y by replacing the left-hand side by Y^i. Note that increasing the level of the left-hand size turns well-formed rules into well-formed rules. Rules for each of the $Z \circ Y$ are easily obtained from rules for Z as follows.

- If there is a rule $Z \supseteq$, for each rule having Y on the left side we add the same rule with $Z \circ Y$ on the left side,
- for each rule $Z \supseteq \delta$ we add a rule $(Z \circ Y) \supseteq \delta Y^{\mathsf{lev}(Z)}$,
- for each rule $Z \supseteq \delta X_1$ we add a rule $(Z \circ Y) \supseteq \delta(X_1 \circ Y)$,
- for each rule $Z \supseteq \delta X_1 X_2$ we add a rule $(Z \circ Y) \supseteq \delta X_1(X_2 \circ Y)$.

It is straightforward to check that this is a well-formed grammar describing the family $\mathcal{Y} := \mathcal{X} \cup \{Z \circ Y : Z \in \mathcal{X}\} \cup \{Y^i : 0 \leq i \leq n\}$. $\square$

5 A Family of Runs

We now define a family $\mathcal{X}$ described by a well-formed grammar. We first name the sets of runs that we define in the following. Some of our classes have subscripts from ε, $\not\varepsilon$, $=$, and $<$. Subscript ε marks a set if all runs in the set only perform ε-transitions, while $\not\varepsilon$ marks a set if each run in the set performs at least one non-ε-transitions. Subscript $<$ marks sets (of pumping runs) where each run starts in a smaller stack than it ends, while $=$ marks sets where no run starts in a smaller stack than it ends (it follows that each such pumping run ends in the same stack as it starts). $\mathcal{X}$ consists of the following sets.

- $\mathcal{Q}$ of all runs,
- $\mathcal{N}_k$ of top^k-non-erasing runs,
- $\mathcal{P}$ of pumping runs which is the disjoint union of the sets $\mathcal{P}_{x,y}$ for $x \in \{<, =\}$, $y \in \{\varepsilon, \not\varepsilon\}$. Additionally, we set $\mathcal{P}_{\not\varepsilon} = \mathcal{P}_{<,\not\varepsilon} \cup \mathcal{P}_{=,\not\varepsilon}$ and $\mathcal{P}_{\varepsilon} = \mathcal{P}_{<,\varepsilon} \cup \mathcal{P}_{=,\varepsilon}$.
- $\mathcal{R}_{k,j}$ of k-returns of change level $j \geq k$ which is the disjoint union of the sets $\mathcal{R}_{k,j,y}$ for $y \in \{\varepsilon, \not\varepsilon\}$, and
- $\mathcal{C}_{k,j}$ of k-colreturns of change level $j \geq k$ which is the disjoint union of the sets $\mathcal{C}_{k,j,y}$ for $y \in \{\varepsilon, \not\varepsilon\}$.

In order to easily distinguish between ε-runs and $\not\varepsilon$-runs in the rules, we partition the transition relation $\Delta = \Delta_\varepsilon \cup \Delta_{\not\varepsilon}$ such that Δ_ε contains exactly the ε-labelled transitions. Before we can give rules for the family we need to define the levels of its sets. We set $\mathsf{lev}(\mathcal{Q}) = n, \mathsf{lev}(\mathcal{R}_{k,j,y}) = k, \mathsf{lev}(\mathcal{C}_{k,j,y}) = k, \mathsf{lev}(\mathcal{N}_k) = n$ and $\mathsf{lev}(\mathcal{P}_{x,y}) = 0$.

Now we give rules for these sets and we describe the main properties of runs in each of the sets. In Appendix E we prove that these descriptions are correct.

Recall that we have described $\mathcal{Q}$ by well-formed rules in Example 4.3. The sets of returns and colreturns are auxiliary sets. Returns occur in the wf-rules for $\mathcal{N}_k$ and $\mathcal{P}_{x,y}$ while colreturns are necessary to give wf-rules for returns.

$\mathcal{N}_k$ contains all runs R where the topmost k-stack of $R(0)$ is never removed during the run. First, we give an idea how the set $\mathcal{N}_0$ plays an important role in our pumping lemma. Recall that we want to apply Theorem 3.1 to pumping runs in order to obtain arbitrary many runs starting in a given configuration. Our final goal is to construct infinitely many different paths in the ε-contraction of the graph of a given collapsible pushdown system that all end in a specific state q. But in general, the pumping runs we construct end in a different state. Thus, the type of the stack reached by each of the pumping runs should determine that we can reach a configuration with state q from this position. This could be done using the set $\mathcal{Q}$ but it is not enough: if the pumping runs induce ε-labelled paths then we could append runs from $\mathcal{Q}$ that all lead to the same configuration. In this case, we construct longer and longer runs but all these runs encode the same edge in the ε-contraction. This is prohibited by the use of runs from $\mathcal{N}_0$: we can prove that the longer pumping runs we construct end in larger stacks. Appending a run from $\mathcal{N}_0$ to such a run ensures that the resulting run also ends in a large stack. From this observation we will obtain infinitely many runs that end in different configurations with state q. Thus, they induce infinitely many paths in the ε-contraction.

The rules for $\mathcal{N}_k$ are

- $\mathcal{N}_k \supseteq$,
- $\mathcal{N}_k \supseteq \delta\mathcal{N}_k$ for each $\delta \in \Delta$ performing an operation of level at most k,
- $\mathcal{N}_k \supseteq \delta^j\mathcal{N}_{j-1}$ for each $\delta^j \in \Delta$ performing a push^j and $j \geq k+1$,
- $\mathcal{N}_k \supseteq \delta^j\mathcal{R}_{j,j}\mathcal{N}_k$ for each $\delta^j \in \Delta$ performing a push^j.

Our analysis of returns reveals that δ^j followed by a run from $\mathcal{R}_{j,j}$ starts and ends in the same stack. Thus, the last rule satisfies the requirement that the topmost j-stacks of these stacks coincide. Moreover, such a run never changes the topmost j-stack of the initial configuration. Using this fact it is straightforward to see that every run described by these rules does not remove the topmost k-stack. The other direction, i.e., the proof that every run preserving the existence of the topmost k-stack is described by one of these rules, can be found in the appendix.

Some run R is a pumping run, i.e., $R \in \mathcal{P}$, if its final stack is created completely on top of its initial stack in the following sense: the topmost 1-stack of $R(|R|)$ is obtained as a (possibly modified) copy of the topmost 1-stack of $R(0)$, and in this copy the topmost 0-stack of $R(0)$ was never removed. Another view on this definition is as follows: for each k, the run R may look into a copy of the topmost k-stack of $R(0)$ only if this copy is not directly involved in the creation of the topmost k-stack of $R(|R|)$. In Appendix D we define a history function that makes the notion of being involved in the creation of some stack precise: for each $i < |R|$ and for each k-stack s^k of $R(|R|)$ we can identify a k-stack t^k in $R(i)$ which is the maximal k-stack involved in the creation of this stack.

In the rest of this section y, y_0, y_1, y_2 are variables in $\{\varepsilon, \not\varepsilon\}$ where we assume that either all are ε or $y = \not\varepsilon$ and one of the y_i occurring in the rule is $\not\varepsilon$. The rules for $\mathcal{P}$ are

- $\mathcal{P}_{=,\varepsilon} \supseteq$,

- $\mathcal{P}_{<,y} \supseteq \delta_{y_0} \mathcal{P}_{x,y_1}$ for each $\delta_{y_0} \in \Delta_{y_0}$ performing push^j and $x \in \{=,<\}$,
- $\mathcal{P}_{x,y} \supseteq \delta^j_{y_0} \mathcal{R}_{j,j,y_1} \mathcal{P}_{x,y}$ for each $\delta^j_{y_0} \in \Delta_{y_0}$ performing push^j and $x \in \{=,<\}$,
- $\mathcal{P}_{<,y} \supseteq \delta^j_{y_0} \mathcal{R}_{j,j',y_1} \mathcal{P}_{x,y_2}$ for each $\delta^j_{y_0} \in \Delta_{y_0}$ performing push^j, $j' > j$ and $x \in \{=,<\}$.

Proving the correctness of this set of rules with respect to our intended meaning of the sets $\mathcal{P}_{x,y}$ requires a very detailed study of returns which is done in Appendix E. In order to see that the rules of the last two kinds are well-formed we need the property that for every run R which first performs a push^j operation followed by a j-return, the topmost j-stack of $R(0)$ and $R(|R|)$ is the same.

Example 5.1. A run of length 1 performing a push^1 operation is a pumping run. Also a run of length 2 performing a push^1 operation followed by a pop^1 operation is a pumping run. However a run of length 2 performing first a pop^1 operation and then a push^1 operation is not a pumping run. This shows that in the definition of a pumping run we do not only care about the initial and final configuration, but about the way the final configuration is created by the run: a pumping run R may never remove the topmost 0-stack of $R(0)$. Next consider a run R of length 3 performing the sequence of operations

$$\mathsf{push}^2,\ \mathsf{pop}^1,\ \mathsf{pop}^2.$$

It is also a pumping run. Notice that this run "looks" into a copy of the topmost 1-stack of $R(0)$, i.e., it removes its topmost 0-stack whence it depends on symbols of $R(0)$ other than the topmost one. One can see that in any 2-CPS, whenever a pumping run R looks into a copy of the topmost 1-stack of $R(0)$, then this copy is completely removed from the stack at some later point in the run. However this is not true for higher levels, as shows the next pumping run R which performs the sequence

$$\mathsf{push}^2,\ \mathsf{pop}^1,\ \mathsf{push}^3,\ \mathsf{pop}^2.$$

Next we define returns. A run R is a k-return (where $1 \leq k \leq n$) if

- the topmost $(k-1)$-stack of $R(|R|)$ is obtained as a copy of the second topmost $(k-1)$-stack of $R(0)$ (in particular we require that there are at least two $(k-1)$-stacks in the topmost k-stack of $R(0)$), and
- while tracing this copy of the second topmost $(k-1)$-stack of $R(0)$ which finally becomes the topmost $(k-1)$-stack of $R(|R|)$, it is never the topmost $(k-1)$-stack of $R(i)$ for all $i < |R|$.

Additionally, for a k-return R its change level is the maximal j such that the topmost j-stack of the initial and of the final stack of R differ in size (i.e. in the number of $(j-1)$-stacks they contain).[7] One can see that the topmost k-stack of $R(0)$ is always greater by one than the topmost k-stack of $R(|R|)$, so we have $j \geq k$. Recall that $\mathcal{R}_{k,j}$ is the set of k-returns of change level j.

[7] One can see that it is the same as saying that the initial and of the final stack of R differ. However a definition using size is more convenient.

Let us just give some intuition about returns before we state their exact characterisation using wf-rules. The easiest sets of returns are those where $k = j$. A run $R \in \mathcal{R}_{k,k}$ starts in some stack s, ends in the stack $\mathsf{pop}^k(s)$, and never visits $\mathsf{pop}^k(s)$ (or any smaller stack) before the final configuration. Notice also that there is a minor restriction on the use of collapse operations: R is not allowed to use links of level k stored in s in order to reach $\mathsf{pop}^k(s)$. Indeed, if such link would be used, then the topmost $(k-1)$-stack of $R(|R|)$ would not be a copy of the second topmost $(k-1)$-stack of $R(0)$, but a copy of the $(k-1)$-stack stored in the link used. Note that this distinction is due to our special representation of links, yet it is useful for the understanding of the definitions.

In the case that $j > k$ things are more complicated but similar. This time $R \in \mathcal{R}_{k,j}$ makes a number of copies of the (possibly modified) topmost $(j-1)$-stack of the initial stack s whence the topmost j-stack of the final stack s' is of bigger size than the topmost j-stack of s. But again the topmost k-stack of s' is the same as the topmost k-stack of $\mathsf{pop}^k(s)$, and is in fact created as a modified copy of the topmost k-stack of s. Furthermore, while tracing the history of this copy along the configurations of the run, the size of this copy is always greater than its size in $R(|R|)$. Notice however that we may also create some other copies of the topmost k-stack of s, in which we can remove arbitrarily many $(k-1)$-stacks. Finally, there is again a minor restriction on the use of collapse links stored in the initial stack s. This restriction implies that the stack obtained via application of the stack operations of the return to s is independent of the linked stacks, i.e., if we replace one of the links of the stack s such that the stack operations of R are still applicable to the resulting stack s', then this sequence of stack operations applied to s' results in the same stack (as when applied to s). In Example 5.4 we discuss the conditions under which we may use a link stored in the initial stack of some return.

Example 5.2. Consider a run R of length 6 (of a collapsible pushdown system of level 2) which performs the following sequence of operations:

$$\mathsf{push}^2,\ \mathsf{pop}^1,\ \mathsf{pop}^2,\ \mathsf{pop}^1,\ \mathsf{push}^1,\ \mathsf{pop}^1.$$

Below we use the notation that symbols taken in square brackets are in one 1-stack (we omit the collapse links). Assume we start from a stack $[aa][aa]$. The stacks of the following configurations of R are:

$$[aa][aa][aa],\ [aa][aa][a],\ [aa][aa],\ [aa][a],\ [aa][aa],\ [aa][a].$$

We have $R{\restriction}_{0,2} \in \mathcal{R}_{1,2}$, $R{\restriction}_{0,4}, R{\restriction}_{1,2}, R{\restriction}_{3,4}, R{\restriction}_{5,6} \in \mathcal{R}_{1,1}$ and $R{\restriction}_{1,3}, R{\restriction}_{2,3} \in \mathcal{R}_{2,2}$. These are the only subruns of R being returns, in particular R is not a 1-return because it visits its final stack before its final configuration.

Example 5.3. The run R of length 5 performing the following sequence of operations

$$\mathsf{push}^2,\ \mathsf{pop}^1,\ \mathsf{push}^3,\ \mathsf{pop}^2,\ \mathsf{pop}^1$$

is a 1-return of change level 3. Notice that the final stack contains a copy of the topmost 1-stack of $R(0)$ with its topmost 0-stack removed.

The rules for returns are as follows:

- $\mathcal{R}_{k,k,y} \supseteq \delta_{y_0}$ for each $\delta_{y_0} \in \Delta_{y_0}$ performing pop^k,
- $\mathcal{R}_{k,j,y} \supseteq \delta_{y_0}\mathcal{R}_{k,j,y_1}$ for each $\delta_{y_0} \in \Delta_{y_0}$ performing an operation of level $< k$,
- $\mathcal{R}_{k,j,y} \supseteq \delta^{j_0}_{y_0}\mathcal{R}_{k,j_1,y_1}$ for each $\delta^{j_0}_{y_0} \in \Delta_{y_0}$ performing a push^{j_0} such that $j_0 > k$ and $\max\{j_0, j_1\} = j$,
- $\mathcal{R}_{k,j,y} \supseteq \delta^{j_0}_{y_0}\mathcal{R}_{j_0,j_0,y_1}\mathcal{R}_{k,j,y_2}$ for each $\delta^{j_0}_{y_0} \in \Delta_{y_0}$ performing a push of level j_0,
- $\mathcal{R}_{k,j,y} \supseteq \delta^{j_0}_{y_0}\mathcal{R}_{j_0,j_1,y_1}\mathcal{R}_{k,j_2,y_2}$ for each $\delta^{j_0}_{y_0} \in \Delta_{y_0}$ performing a push^{j_0} such that $j_1 > j_0$ and $\max\{j_1, j_2\} = j$, and
- $\mathcal{R}_{k,j,y} \supseteq \delta_{y_0}\mathcal{C}_{k,j,y_1}$ for each $\delta_{y_0} \in \Delta_{y_0}$ performing a $\mathsf{push}^1_{a,k}$.

A k-colreturn is a run R that performs in the last step a col^k on a copy of the topmost symbol of its initial stack. The change level of k-colreturns is (again) defined as the maximal j such that the topmost j-stack of the initial and of the final stack of the colreturn R differ in size.

Note that k-colreturns appear in the rules for returns after a push of level 1. The simplest example of a return described by the third rule is a run R starting in a stack s and performing $\mathsf{push}^1_{a,k}$ and then col^k. Note that such a sequence has the same effect as applying pop^k to s. Note that $R{\restriction}_{1,2}$ in this example is a run from the stack $s' := \mathsf{push}^1_{a,k}(s)$ to $\mathsf{pop}^k(s')$ (for $k \geq 2$). Nevertheless we exclude it from the definition of a k-return of change level k because this effect is not transferable to arbitrary other stacks: of course, we can apply the transition of $R{\restriction}_{1,2}$ to the stack $\mathsf{push}^k(s')$ and obtain a run R' from $\mathsf{push}^k(s')$ to $\mathsf{pop}^k(s')$. But apparently this is not a run from some stack s'' to a stack $\mathsf{pop}^k(s'')$, so it is not a k-return. For this reason our definition of returns disallows the application of certain stored collapse links. The colreturns take care of such situations where we use the links stored in the stack. Notice that k-colreturns occur in the rules defining the other sets of runs only at those points where we performed a push of level 1 whence we can be sure that the effect of the collapse operation coincides with the application of exactly one pop^k operation to the initial stack.

Example 5.4. Consider a run R of length 4 performing

$$\mathsf{push}^2, \mathsf{col}^1, \mathsf{pop}^2, \mathsf{pop}^1.$$

It is a 1-return of change level 1. Notice that it performs a collapse operation using a (copy of a) link already stored in $R(0)$. But $R{\restriction}_{1,3}$ is a 2-return (of change level 2) which covers this collapse operation, i.e., whenever this sequence is applicable to some stack s it ends in the stack $\mathsf{pop}^1(s)$. As a general rule, we allow the use of a col^k from a (copy) of a link stored in the initial stack of some return R if it occurs within some subrun R' that is a k'-return or k'-colreturn of higher level(i.e., $k' > k$). In such cases the resulting stack does not depend on the stack stored in the link (as long as the whole sequence of operations of the return is applicable). Hence, the following sequence of operations also induces a 1-return of change level 1: $\mathsf{push}^3,\ \mathsf{col}^2,\ \mathsf{pop}^3,\ \mathsf{pop}^1$.

Finally, let us state the rules for k-colreturns.

- $\mathcal{C}_{k,k,y} \supseteq \delta_{y_0}$ for each $\delta_{y_0} \in \Delta_{y_0}$ performing a col^k,
- $\mathcal{C}_{k,j,y} \supseteq \delta_{y_0}^{j_0}\mathcal{C}_{k,j_1,y_1}$ for each $\delta_{y_0}^{j_0} \in \Delta_{y_0}$ performing a push^{j_0} such that $j_0 \geq 2$ and $\max\{j_0, j_1\} = j$,
- $\mathcal{C}_{k,j,y} \supseteq \delta_{y_0}^{j_0}\mathcal{R}_{j_0,j_0,y_1}\mathcal{C}_{k,j,y_2}$ for each $\delta_{y_0}^{j_0} \in \Delta_{y_0}$ performing a push^{j_0}, and
- $\mathcal{C}_{k,j,y} \supseteq \delta_{y_0}^{j_0}\mathcal{R}_{j_0,j_1,y_1}\mathcal{C}_{k,j_2,y_2}$ for each $\delta_{y_0}^{j_0} \in \Delta_{y_0}$ performing a push^{j_0} such that $j_1 > j_0$ and $\max\{j_1, j_2\} = j$.

This completes the presentation of the well-formed grammar describing $\mathcal{X}$.

References

1. K. Aehlig, J. G. de Miranda, and C.-H. L. Ong. Safety is not a restriction at level 2 for string languages. In V. Sassone, editor, *FoSSaCS*, volume 3441 of *Lecture Notes in Computer Science*, pages 490–504. Springer, 2005.
2. A. Blumensath. On the structure of graphs in the caucal hierarchy. *Theor. Comput. Sci.*, 400(1-3):19–45, 2008.
3. C. Broadbent, A. Carayol, M. Hague, and O. Serre. A Saturation Method for Collapsible Pushdown Systems. In *to appear in ICALP*, 2012.
4. A. Carayol and S. Wöhrle. The Caucal hierarchy of infinite graphs in terms of logic and higher-order pushdown automata. In P. K. Pandya and J. Radhakrishnan, editors, *FSTTCS*, volume 2914 of *Lecture Notes in Computer Science*, pages 112–123. Springer, 2003.
5. D. Caucal. On infinite terms having a decidable monadic theory. In K. Diks and W. Rytter, editors, *MFCS*, volume 2420 of *Lecture Notes in Computer Science*, pages 165–176. Springer, 2002.
6. R. H. Gilman. A shrinking lemma for indexed languages. *Theor. Comput. Sci.*, 163(1&2):277–281, 1996.
7. M. Hague, A. S. Murawski, C.-H. L. Ong, and O. Serre. Collapsible pushdown automata and recursion schemes. In *LICS*, pages 452–461. IEEE Computer Society, 2008.
8. T. Hayashi. On derivation trees of indexed grammars. *Publ. RIMS, Kyoto Univ.*, 9:61–92, 1973.
9. A. Kartzow. A pumping lemma for collapsible pushdown graphs of level 2. In M. Bezem, editor, *CSL*, volume 12 of *LIPIcs*, pages 322–336. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2011.
10. T. Knapik, D. Niwinski, and P. Urzyczyn. Higher-order pushdown trees are easy. In M. Nielsen and U. Engberg, editors, *FoSSaCS*, volume 2303 of *Lecture Notes in Computer Science*, pages 205–222. Springer, 2002.
11. A. N. Maslov. The hierarchy of indexed languages of an arbitrary level. *Soviet Math. Dokl.*, 15:1170–1174, 1974.
12. A. N. Maslov. Multilevel stack automata. *Problems of Information Transmission*, 12:38–43, 1976.
13. P. Parys. Collapse operation increases expressive power of deterministic higher order pushdown automata. In T. Schwentick and C. Dürr, editors, *STACS*, volume 9 of *LIPIcs*, pages 603–614. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2011.
14. P. Parys. On the significance of the collapse operation. Accepted to LICS, 2012.
15. P. Parys. A pumping lemma for pushdown graphs of any level. In C. Dürr and T. Wilke, editors, *STACS*, volume 14 of *LIPIcs*, pages 54–65. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2012.

A Deviations from Standard Definitions

Our definition of a collapsible pushdown system of level n deviates from the original one from [7] in several respects.

1. Instead of storing links which control the collapse operation, we store the content of the stack where the link points to. Note that this is only a syntactical difference.
2. Instead of having one collapse operation whose level is controlled by the topmost element of the stack, we have one collapse operation for each stack level. Note that we can simulate a collapse transition in the original sense by using n collapse transitions (one for each level).
3. Finally, for reasons of uniformity, the definition of the $\mathsf{push}^1_{a,1}$ operation differs from the original one. If we apply this operation to a stack s we do not create a link to the topmost 1-stack s^1 of s but to $\mathsf{pop}^1(s^1)$. Thus, the effect of using a collapse of level 1 in the original definition is always equal to the effect of applying pop^1 while in our definition it is always equal to the effect of applying pop^1 *twice*. Nevertheless, we can simulate every system of the original definition by simulating collapse of level one with a pop^1 operation directly.[8]

Due to these observations, it should be clear that every "original" collapsible pushdown system of level n and size s is simulated by a collapsible pushdown system of level n and size at most $n \cdot s$. Moreover, with respect to ε-contractions of the configuration graphs both definitions are equal (from our systems to the original ones, we need one ε-transition in order to simulate each col^1 operation by two pop^1 operations; all other transitions are translated one-to-one).

B Types of Stacks and Configurations—Definitions

Before we start defining types, let us introduce one more restriction for a set of wf-rules.

Definition B.1. A set $\mathcal{R}_\mathcal{X}$ of well-formed rules over $\mathcal{X}$ is called *well-formed* if for each rule $(X \supseteq \delta YZ) \in \mathcal{R}_\mathcal{X}$ there is also the rule $(X_{\delta Y} \supseteq \delta Y) \in \mathcal{R}_\mathcal{X}$ for a new nonterminal $X_{\delta Y}$ such that $\mathsf{lev}(X_{\delta Y}) = \mathsf{lev}(Y)$, and for each rule $(X \supseteq \delta X) \in \mathcal{R}_\mathcal{X}$ or $(X \supseteq \delta XY) \in \mathcal{R}_\mathcal{X}$ there is also the rule $(X_\delta \supseteq \delta) \in \mathcal{R}_\mathcal{X}$ for a new nonterminal X_δ.

Remark B.2. Note that each set of well-formed rules over some family $\mathcal{X}$ can be turned into a well-formed set of well-formed rules by adding the necessary symbols X_δ and $X_{\delta Y}$ to $\mathcal{X}$, the corresponding rules to the set of rules, and by setting $\mathsf{lev}(X_\delta) = n$ and $\mathsf{lev}(X_{\delta Y}) = \mathsf{lev}(Y)$.

[8] Notice however that col^1 can be performed only if the topmost 0-stack stores a 1-stack. The same should be true for the new transition performing pop^1. To ensure this, it is enough to extend the stack alphabet so that the stack symbol stores also the level of the stack stored in the 0-stack.

For the rest of this section we fix some finite family $\mathcal{X}$ described by a well-formed set $\mathcal{R}_{\mathcal{X}}$ of wf-rules. The aim of this section is to assign to any k-stack s^k a *type* $\mathsf{type}_{\mathcal{X}}(s^k)$ that determines the possible runs from any of the sets $X \in \mathcal{X}$ starting in a stack with topmost k-stack s^k. The type of s^k is a set of *run descriptors* which come from a set $\mathcal{T}^k$ that are defined inductively from $k = n$ to $k = 0$. A typical element of $\mathcal{T}^k$ has the form

$$\sigma = (\Sigma^n, \Sigma^{n-1}, \dots, \Sigma^{k+1}, p, \widehat{\sigma}) \text{ with } \widehat{\sigma} = (X, \Omega^n, \Omega^{n-1}, \dots, \Omega^{\mathsf{lev}(X)+1}, q),$$

where $X \in \mathcal{X}$ is one of the sets of runs we are interested in, Σ^i and Ω^i are types of i-stacks, and $p, q \in Q$ are states of the CPS $\mathcal{S}$. Let us explain the intended meaning of such a tuple. We want to have $\sigma \in \mathsf{type}_{\mathcal{X}}(s^k)$ if and only if for all stacks $t^n, t^{n-1}, \dots, t^{k+1}$ where $\Sigma^i \subseteq \mathsf{type}_{\mathcal{X}}(t^i)$ there is a run $R \in X$ such that $R(0) = (p, t^n : t^{n-1} : \dots : t^{k+1} : s^k)$ and $R(|R|) = (q, u^n : u^{n-1} : \dots : u^0)$ such that $\Omega^j \subseteq \mathsf{type}_{\mathcal{X}}(u^j)$ for all $\mathsf{lev}(X) + 1 \le j \le n$. In other words, if we put σ into $\mathsf{type}_{\mathcal{X}}(s^k)$ we claim the following. If for each $k + 1 \le i \le n$ we take an i-stack t^i that satisfies the claims of Σ^i, then there is a run in X that starts in state p and the stack obtained by putting s^k on top of the n-stack $t^n : \dots : t^{k+1}$, ends in state q, and the final stack decomposes into a sequence of stacks such that the j-th element satisfies all claims of Ω^j.

Recall that a 0-stack contains the whole stack to which it links. Thus the type of a 0-stack depends not only on its symbol, but also on the whole stack it contains in the link. In order to deal with this fact we define the type of a stack by induction on its level and by induction on the nesting depth of its links.

We first introduce the set $\mathcal{T}^k$ of possible run descriptors of level k (the possible types of k-stacks are the elements of $\mathcal{P}(\mathcal{T}^k)$).[9]

Definition B.3. Let $0 \le k \le n$. Assume we have already defined sets $\mathcal{T}^i$ for $k + 1 \le i \le n$. We take

$$\mathcal{T}^k = \{\mathsf{ne}\} \cup \big(\mathcal{P}(\mathcal{T}^n) \times \mathcal{P}(\mathcal{T}^{n-1}) \times \dots \times \mathcal{P}(\mathcal{T}^{k+1}) \times Q \times \mathcal{D}^k\big),$$

$$\mathcal{D}^k = \bigcup \{\mathcal{D}_X \colon X \in \mathcal{X}, \mathsf{lev}(X) \ge k\}, \quad \text{where}$$

$$\mathcal{D}_X = \{X\} \times \mathcal{P}(\mathcal{T}^n) \times \mathcal{P}(\mathcal{T}^{n-1}) \times \dots \times \mathcal{P}(\mathcal{T}^{\mathsf{lev}(X)+1}) \times Q.$$

Note that beside the run descriptors of the typical form, we also have $\mathsf{ne} \in \mathcal{T}^k$: ne will appear in the type of some stack if and only if this stack is non-empty. In order to easily talk about the intended meaning of types and run descriptors we introduce the following definition.

Definition B.4. Let $\widehat{\sigma} = (X, \Omega^n, \Omega^{n-1}, \dots, \Omega^{\mathsf{lev}(X)+1}, q) \in \mathcal{D}_X$ for some $X \in \mathcal{X}$. We say that a run R *agrees with* $\widehat{\sigma}$ if $R \in X$, and R ends in a configuration $(q, t^n : t^{n-1} : \dots : t^0)$ such that $\Omega^i \subseteq \mathsf{type}(t^i)$ for $\mathsf{lev}(X) + 1 \le i \le n$.

Now we can reformulate the intended meaning of a run descriptors and types.

[9] We write $\mathcal{P}(X)$ for the power set of X.

Lemma B.5. *Let $\widehat{\sigma} \in \mathcal{D}_X$ for some $X \in \mathcal{X}$, and let $0 \leq l \leq \mathsf{lev}(X)$. Let $c = (p, s^n : s^{n-1} : \cdots : s^l)$ be a configuration. Then there is a run from c which agrees with $\widehat{\sigma}$ if and only if there is a tuple $(\Psi^n, \Psi^{n-1}, \dots, \Psi^{l+1}, p, \widehat{\sigma}) \in \mathsf{type}(s^l)$ such that $\Psi^i \subseteq \mathsf{type}(s^i)$ for $l+1 \leq i \leq n$.*

We postpone the definition of types of stacks to the end of this section. The proof of Lemma B.5 can be found in Appendix C. Assuming that we already knew what the type of a stack is, it is easy to define $\mathsf{ctype}_{\mathcal{X}}$, the function mapping configurations to their types. After giving the definition, we then already can prove part 1 of Theorem 3.1.

Definition B.6. Let $c = (q, s)$ be a configuration with $s = s^n : \cdots : s^0$. Set

$$\mathsf{ctype}_{\mathcal{X}}(c) := (\mathsf{type}_{\mathcal{X}}(s^n), \dots, \mathsf{type}_{\mathcal{X}}(s^0), q).$$

We define a partial order on the types of configurations as follows: $(\Phi^n, \dots, \Phi^0, p) \sqsubseteq (\Psi^n, \dots, \Psi^0, q)$ if and only if $p = q$ and $\Phi^i \subseteq \Psi^i$ for $0 \leq i \leq n$.

Proof (of part 1 of Theorem 3.1). Let R start in state p and end in state q. Let the pds of $R(0)$ be $s^n : s^{n-1} : \cdots : s^0$, let the pds of $R(|R|)$ be $t^n : t^{n-1} : \cdots : t^0$, and let the pds of c be $u^n : u^{n-1} : \cdots : u^0$. The assumptions say that R agrees with

$$\widehat{\sigma} = (X, \mathsf{type}(t^n), \mathsf{type}(t^{n-1}), \dots, \mathsf{type}(t^{\mathsf{lev}(X)+1}), q).$$

Due to Lemma B.5, there are $\Psi^i \subseteq \mathsf{type}(s^i)$ for each $1 \leq i \leq n$ such that run descriptor $\sigma = (\Psi^n, \Psi^{n-1}, \dots, \Psi^1, p, \widehat{\sigma}) \in \mathsf{type}(s^0)$. Since the types of s^i are included in the types of u^i, we also have $\sigma \in \mathsf{type}(u^0)$, and $\Psi^i \subseteq \mathsf{type}(u^i)$ for $1 \leq i \leq n$. Using again Lemma B.5 there is a run S from c which agrees with $\widehat{\sigma}$. By definition this implies that $S \in X$ and S ends in state q.

Moreover, decomposing the final stack of S as $v^n : v^{n-1} : \cdots : v^0$, we obtain

$$\mathsf{type}(t^i) \subseteq \mathsf{type}(v^i) \text{ for each } \mathsf{lev}(X) + 1 \leq i \leq n. \quad (1)$$

□

The proof of part 2 requires a more detailed knowledge about the types. Thus, we postpone it to the end of this section.

Next we prepare the definition of types. We first define composers. The intention is that a composer gives us the type of a k-stack s^k from the types of its decomposition as $s^k = t^k : t^{k-1} : \cdots : t^l$.

Definition B.7. Let $0 \leq l \leq k \leq n$, and let $\Psi^i \subseteq \mathcal{T}^i$ for each $l \leq i \leq k$. Their *composer* $\mathsf{comp}(\Psi^k, \Psi^{k-1}, \dots, \Psi^l)$ is the a subset of $\mathcal{T}^k$ containing all tuples $(\Sigma^n, \Sigma^{n-1}, \dots, \Sigma^{k+1}, q, \widehat{\sigma})$ such that there is a $(\Sigma^n, \Sigma^{n-1}, \dots, \Sigma^{l+1}, q, \widehat{\sigma}) \in \Psi^l$ such that $\Sigma^i \subseteq \Psi^i$ for $l+1 \leq i \leq k$ and $\widehat{\sigma} \in \mathcal{D}^k$; additionally the composer contains ne if $l < k$ or if $\mathsf{ne} \in \Psi^k$.

Remark B.8. Note that $\mathsf{comp}(\Psi^k) = \Psi^k$. Furthermore, note that the definitions concerning ne are compatible with our intended meaning: a k-stack s^k is nonempty if it is composed as $s^k = t^k : t^{k-1} : \cdots : t^l$ for $l < k$. All t^i may be empty stacks, but the resulting k-stack contains a list of $(k-1)$-stacks whose topmost element is the possibly empty $(k-1)$-stack $t^{k-1} : \cdots : t^l$. Even if all elements of a list are empty stacks, it is not empty itself as long as it contains at least one element.

Note that the following properties of comp follow directly from the definition.

Lemma B.9. *Let $1 \leq k \leq n$, and for each $0 \leq i \leq k$ let $\Psi^i \subseteq \Phi^i \in \mathcal{P}(\mathcal{T}^i)$. Then $\mathsf{comp}(\Psi^k, \Psi^{k-1}, \dots, \Psi^0) \subseteq \mathsf{comp}(\Phi^k, \Phi^{k-1}, \dots, \Phi^0)$.*

Lemma B.10. *Let $0 \leq l < j < k \leq n$, and let $\Psi^i \in \mathcal{P}(\mathcal{T}^i)$ for all $l \leq i \leq k$. Then $\mathsf{comp}(\Psi^k, \Psi^{k-1}, \dots, \Psi^l) = \mathsf{comp}(\Psi^k, \Psi^{k-1}, \dots, \Psi^{j+1}, \mathsf{comp}(\Psi^j, \Psi^{j-1}, \dots, \Psi^l))$.*

In the set $\mathcal{R}_\mathcal{X}$ of rules we distinguish a subset $\mathcal{R}_\mathcal{X}^{>0}$ of those rules which are not of the form $X \supseteq$ (i.e., they describe runs of positive lengths). In the next step towards the definition of types, to each rule $r = (X \supseteq \delta X_1 \dots X_m) \in \mathcal{R}_\mathcal{X}^{>0}$, where the operation in δ is op, we assign two numbers $L(op)$ and $M(r)$ as follows.

$$L(op) = \begin{cases} k & \text{if } op = \mathsf{pop}^k \text{ or } op = \mathsf{col}^k, \\ 0 & \text{otherwise,} \end{cases}$$

$$M(r) = \begin{cases} \mathsf{lev}(X) & \text{if } m = 0, \\ \mathsf{lev}(X_1) & \text{otherwise.} \end{cases}$$

Note that the inequality $0 \leq L(op) \leq M(r) \leq n$ follows from the definition of wf-rules. Then, to each rule $r = (X \supseteq \delta X_1 \dots X_m) \in \mathcal{R}_\mathcal{X}^{>0}$ we assign a set

$$\begin{aligned} T(r) \subseteq & \big(\mathcal{P}(\mathcal{T}^n) \times \mathcal{P}(\mathcal{T}^{n-1}) \times \cdots \times \mathcal{P}(\mathcal{T}^0)\big) \times \\ & \times \big(\mathcal{P}(\mathcal{T}^n) \times \mathcal{P}(\mathcal{T}^{n-1}) \times \cdots \times \mathcal{P}(\mathcal{T}^{M(r)})\big) \times \mathcal{D}_X. \end{aligned}$$

The intuitive meaning of this set is as follows. Consider a run R described by a wf-rule r. The first part of a tuple describes the types of the stack of $R(0)$ (decomposed as $s^n : \cdots : s^0$), the second part describes the types of the stack of $R(1)$ (decomposed as $t^n : \cdots : t^{M(r)}$) and the last part is an element with which this run agrees. In fact, in $T(r)$ we do not care whether δ can connect two stacks of the described types for $R(0)$ and $R(1)$. We define it in such a way that under the assumption that δ may connect two stacks of the corresponding type, the third part of the tuple in $T(r)$ in fact permits a run that agrees with this description and starts in the stack whose types are described by the first part of the tuple. The question whether δ can transform a stack of a certain type into a stack of another type is later dealt with when defining another function U.

Definition B.11. Let $r = (X \supseteq \delta X_1 \dots X_m) \in \mathcal{R}_\mathcal{X}^{>0}$ and $\delta = (q_0, a, l, q_1, op)$. We distinguish the following cases.

1. Assume that $r = (X \supseteq \delta)$. The set $T(r)$ contains all tuples
$$((\Psi^n, \Psi^{n-1}, \dots, \Psi^0), (\Phi^n, \Phi^{n-1}, \dots, \Phi^{\mathsf{lev}(X)}), \widehat{\sigma}) \\ \text{for } \widehat{\sigma} = (X, \Omega^n, \Omega^{n-1}, \dots, \Omega^{\mathsf{lev}(X)+1}, q_1)$$
such that $\Omega^i \subseteq \Phi^i$ for $\mathsf{lev}(X) + 1 \leq i \leq n$ (recall that q_1 is the state reached after application of δ).
2. Assume that $r = (X \supseteq \delta X_1)$. The set $T(r)$ contains all tuples
$$((\Psi^n, \Psi^{n-1}, \dots, \Psi^0), (\Phi^n, \Phi^{n-1}, \dots, \Phi^{\mathsf{lev}(X_1)}), \widehat{\sigma}) \\ \text{for } \widehat{\sigma} = (X, \Omega^n, \Omega^{n-1}, \dots, \Omega^{\mathsf{lev}(X)+1}, q')$$
such that in $\Phi^{\mathsf{lev}(X_1)}$ we have a tuple
$$(\Phi^n, \Phi^{n-1}, \dots, \Phi^{\mathsf{lev}(X_1)+1}, q_1, \widehat{\tau}) \\ \text{where } \widehat{\tau} = (X_1, \Omega^n, \Omega^{n-1}, \dots, \Omega^{\mathsf{lev}(X_1)+1}, q').$$
3. Assume that $r = (X \supseteq \delta X_1 X_2)$. The set $T(r)$ contains all tuples
$$((\Psi^n, \Psi^{n-1}, \dots, \Psi^0), (\Phi^n, \Phi^{n-1}, \dots, \Phi^{\mathsf{lev}(X_1)}), \widehat{\sigma}) \\ \text{for } \widehat{\sigma} = (X, \Omega^n, \Omega^{n-1}, \dots, \Omega^{\mathsf{lev}(X)+1}, q')$$
such that in $\Phi^{\mathsf{lev}(X_1)}$ we have a tuple
$$(\Phi^n, \Phi^{n-1}, \dots, \Phi^{\mathsf{lev}(X_1)+1}, q_1, \widehat{\tau}) \\ \text{for } \widehat{\tau} = (X_1, \Sigma^n, \Sigma^{n-1}, \dots, \Sigma^{\mathsf{lev}(X_1)+1}, q_2)$$
and in Ψ^0 we have a tuple
$$(\Sigma^n, \Sigma^{n-1}, \dots, \Sigma^{\mathsf{lev}(X_1)+1}, \Psi^{\mathsf{lev}(X_1)}, \dots, \Psi^1, q_2, \widehat{\rho}) \\ \text{where } \widehat{\rho} = (X_2, \Omega^n, \Omega^{n-1}, \dots, \Omega^{\mathsf{lev}(X_2)+1}, q').$$

Remark B.12. Recall that for each rule of the form $X \supseteq \delta X_1 X_2$ the definition of well-formed rules requires that any run of the form δX_1 starts and ends with the same topmost level k stack for $k = \mathsf{lev}(X_1)$. Thus, for each run performing δX_1 such that the X_1-part agrees with $\widehat{\tau}$ the type of the stack is completely determined: if the final stack decomposes as $s^n : s^{n-1} : \dots : s^1 : s^0$, the type of s^k for $k > \mathsf{lev}(X_1)$ is determined by $\widehat{\tau}$ and the type of s^k for $k \leq \mathsf{lev}(X_1)$ is determined by the type of the initial stack, i.e., it is Ψ^k.

Using the function T we define a function U and a function stype which assigns types to 0-stacks. In fact, U and stype are defined as simultaneous fixpoints of sequences $(U_z)_{z \in \mathbb{N}}$ and $(\mathsf{stype}_z)_{z \in \mathbb{N}}$. For each $z \in \mathbb{N}$, each operation op, each number $1 \leq K \leq n$ and each $\Sigma^K \subseteq \mathcal{T}^K$ we define the set
$$U_z(op, K, \Sigma^K) \subseteq \left(\mathcal{P}(\mathcal{T}^n) \times \mathcal{P}(\mathcal{T}^{n-1}) \times \dots \times \mathcal{P}(\mathcal{T}^0)\right) \times \\ \times \Big(\mathcal{P}(\mathcal{T}^n) \times \mathcal{P}(\mathcal{T}^{n-1}) \times \dots \times \mathcal{P}(\mathcal{T}^{L(op)})\Big).$$

The intention is that $U_z(op, K, \Sigma^K)$ contains a tuple

$$((\Psi^n, \Psi^{n-1}, \dots, \Psi^0), (\Pi^n, \Pi^{n-1}, \dots, \Pi^{L(op)}))$$

if for a stack $s = s^n : s^{n-1} : \dots : s^0$ such that Ψ^i is contained in the type of s^i and s^0 carries a link of level K to a stack of type Σ^K, we can decompose $op(s) = t^n : t^{n-1} : \dots : t^{L(op)}$ and $\Pi^j \subseteq \mathsf{type}_{\mathcal{X}}(t^j)$ for all $L(op) \leq j \leq n$. When we enter the fixpoint $U(op, K, \Sigma^K)$ we are able to replace the "if" by an "if and only if". In the definition of U_z we use stype_{z-1} in order to approximate the type of the topmost 0-stack if $op = \mathsf{push}^1$ (from below). In this case, the "and only if" part requires to consider the complete type of the 0-stack pushed onto the stack. The fixpoint stype of the functions stype_z yields this complete type information. For the definition of U_0 and stype_0, we assume that stype_{-1} maps any input to $\emptyset$.

Definition B.13. Let op be a stack operation, let $1 \leq K \leq n$, let $\Sigma^K \subseteq \mathcal{T}^K$, and let $z \in \mathbb{N}$. Assume that stype_{z-1} is already defined. We have four cases according to the operation used in δ.

1. Assume that $op = \mathsf{pop}^k$. Then the set $U_z(op, K, \Sigma^K)$ contains all tuples
$$((\Psi^n, \Psi^{n-1}, \dots, \Psi^0), (\Psi^n, \Psi^{n-1}, \dots, \Psi^k))$$
where $\mathsf{ne} \in \Psi^k$.
2. Assume that $op = \mathsf{col}^k$. If $k \neq K$ or $\mathsf{ne} \notin \Sigma^K$, the set $U_z(op, K, \Sigma^K)$ is empty. If $k = K$ and $\mathsf{ne} \in \Sigma^K$, the set $U_z(op, K, \Sigma^K)$ contains all tuples
$$((\Psi^n, \Psi^{n-1}, \dots, \Psi^0), (\Psi^n, \Psi^{n-1}, \dots, \Psi^{k+1}, \Sigma'^k))$$
such that $\Sigma'^k \subseteq \Sigma^K$.
3. Assume that $op = \mathsf{push}^1_{b,k}$. The set $U_z(op, K, \Sigma^K)$ contains all tuples
$$((\Psi^n, \Psi^{n-1}, \dots, \Psi^0), (\Pi^n, \Pi^{n-1}, \dots, \Pi^0))$$
which satisfy $\Pi^i = \Psi^i$ for $2 \leq i \leq k$, $\Pi^1 = \mathsf{comp}(\Psi^1, \Psi^0)$ and $\Pi^0 \subseteq \mathsf{stype}_{z-1}(b, k, \Psi^k)$.
4. Assume that $op = \mathsf{push}^k$ with $k \geq 2$. The set $U_z(op, K, \Sigma^K)$ contains all tuples
$$((\Psi^n, \Psi^{n-1}, \dots, \Psi^0), (\Pi^n, \Pi^{n-1}, \dots, \Pi^0))$$
such that $\Pi^i = \Psi^i$ for $0 \leq i \leq n$ with $i \neq k$ and $\Pi^k = \mathsf{comp}(\Psi^k, \Psi^{k-1}, \dots, \Psi^0)$.

Definition B.14. Let $z \in \mathbb{N}$, let $a \in \Gamma$, let $1 \leq K \leq n$, and let $\Sigma^K \subseteq \mathcal{T}^K$. Assume that stype_{z-1} and U_z are already defined. We define $\mathsf{stype}_z(a, K, \Sigma^K)$ as the set containing

1. all tuples
$$(\Psi^n, \dots, \Psi^1, q_0, (X, \Omega^n, \Omega^{n-1}, \dots, \Omega^{\mathsf{lev}(X)+1}, q_0))$$
such that $\mathcal{R}_{\mathcal{X}}$ contains the rule $X \supseteq$, and $\Omega^i \subseteq \Psi^i$ for $\mathsf{lev}(X) + 1 \leq i \leq n$ (and q_0 is an arbitrary state), and

2. all tuples $(\Psi^n, \Psi^{n-1}, \dots, \Psi^1, q_0, \widehat{\sigma})$ such that for $0 \leq m \leq 2$ and some rule $r = (X \supseteq \delta X_1 \dots X_m) \in \mathcal{R}_{\mathcal{X}}^{\geq 0}$ with $\delta = (q_0, a, \cdot, \cdot, op)$ we have

$$\begin{aligned}
&((\Psi^n, \dots, \Psi^0), (\Pi^n, \Pi^{n-1}, \dots, \Pi^{L(op)})) \in U_z(op, K, \Sigma^K),\\
&((\Psi^n, \dots, \Psi^0), (\Pi^n, \Pi^{n-1}, \dots, \Pi^{M(r)+1}, \Phi^{M(r)}), \widehat{\sigma}) \in T(r),\\
&\Psi^0 \subseteq \mathsf{stype}_{z-1}(a, K, \Sigma^K), \quad \text{and}\\
&\Phi^{M(r)} = \mathsf{comp}(\Pi^{M(r)}, \Pi^{M(r-1)}, \dots, \Pi^{L(op)}).
\end{aligned}$$

Notice that the sequence stype_z is monotone with respect to both z and Σ^K: for $\Sigma^K \subseteq \Sigma'^K$ and each $z \in \mathbb{N}$ we have $\mathsf{stype}_z(a, K, \Sigma^K) \subseteq \mathsf{stype}_{z+1}(a, K, \Sigma'^K)$. Independent of $z \in \mathbb{N}$, the domain and range of stype_z are fixed finite sets whence there is some $z \in \mathbb{N}$ such that $\mathsf{stype}_z = \mathsf{stype}_{z+1}$. This fixpoint is denoted as $\mathsf{stype}_{\mathcal{X}}$ (formally, stype_z also depends on $\mathcal{X}$).

Definition B.15. We define $\mathsf{type}_{\mathcal{X}}(s^k)$ for each k-stack s^k (for $0 \leq k \leq n$) by induction on the structure of s^k. If s^k is empty,

$$\mathsf{type}_{\mathcal{X}}(s^k) := \emptyset.$$

Otherwise, assume that $k = 0$ and $s^k = (a, K, t^K)$ where $a \in \Gamma$, $1 \leq K \leq n$, and t^K is a K-stack such that $\mathsf{type}_{\mathcal{X}}(t^K)$ is already defined. In this case we set

$$\mathsf{type}_{\mathcal{X}}(s^k) = \mathsf{stype}_{\mathcal{X}}\left(a, K, \mathsf{type}_{\mathcal{X}}(t^K)\right).$$

Finally, assume that $k \geq 1$ and $s^k = t^k : t^{k-1}$ such that $\mathsf{type}_{\mathcal{X}}(t^k)$ and $\mathsf{type}_{\mathcal{X}}(t^{k-1})$ are defined. In this case set

$$\mathsf{type}_{\mathcal{X}}(s^k) = \mathsf{comp}\left(\mathsf{type}_{\mathcal{X}}(t^k), \mathsf{type}_{\mathcal{X}}(t^{k-1})\right).$$

With the help of Lemma B.5 and the properties of the composer, the proof of part 2 of Theorem 3.1 is done as follows.

Proof (of part 2 of Theorem 3.1). The proof is by induction on the length of the run. Let $X \in \mathcal{X}$ be a set of level 0. Let $R \in X$ be a run starting in a pds $s^n : s^{n-1} : \dots : s^0$ and ending in a pds $t^n : t^{n-1} : \dots : t^0$. Furthermore, let $u := u^n : u^{n-1} : \dots : u^0$ be a pds. Assume that $\mathsf{type}_{\mathcal{X}}(s^i) \subseteq \mathsf{type}_{\mathcal{X}}(u^i)$ for $0 \leq i \leq n$. We prove that there is a run $S \in X$ such that S starts in u and ends in a stack $v^n : v^{n-1} : \dots : v^0$ with $\mathsf{type}_{\mathcal{X}}(t^i) \subseteq \mathsf{type}_{\mathcal{X}}(v^i)$ for each $0 \leq i \leq n$ and such that R and S have the same initial and final states. We continue by case distinction on the wf-rule r describing R.

- If $r = (X \supseteq\ )$, the run S of length 0 with $S(0) = (q, u)$ for q the initial state of R satisfies the claim.
- Assume that $r = (X \supseteq \delta)$. Because $\mathcal{X}$ is described by a well-formed set of rules, the run is also described by the rule of the set X_δ. Using part 1 of Theorem 3.1, there is a run S starting in u which performs δ, and such that R and S have the same initial and final states. Since $\mathsf{lev}(X) = 0$, the operation in δ is a push^k of some level k. Notice that for $1 \leq i \leq k-1$ and

for $k+1 \leq i \leq n$ we have $t^i = s^i$ and $v^i = u^i$, so $\mathsf{type}_{\mathcal{X}}(t^i) \subseteq \mathsf{type}_{\mathcal{X}}(v^i)$. The same holds for $i = 0$ if $k \geq 2$. If the operation is $\mathsf{push}^1_{a,j}$, we have $t^0 = (a, j, s^j)$ and $v^0 = (a, j, u^j)$. Since $\mathsf{stype}_{\mathcal{X}}$ is monotone, $\mathsf{type}_{\mathcal{X}}(t^0) = \mathsf{stype}_{\mathcal{X}}(a, j, \mathsf{type}_{\mathcal{X}}(s^j)) \subseteq \mathsf{stype}_{\mathcal{X}}(a, j, \mathsf{type}_{\mathcal{X}}(u^j)) = \mathsf{type}_{\mathcal{X}}(v^0)$. We also have $t^k = s^k : s^{k-1} : \cdots : s^0$ and $v^k = u^k : u^{k-1} : \cdots : u^0$. Due to Lemmas B.10 and B.9, we obtain

$$\begin{aligned}\mathsf{type}_{\mathcal{X}}(t^k) = \mathsf{comp}(\mathsf{type}_{\mathcal{X}}(s^k), \mathsf{type}_{\mathcal{X}}(s^{k-1}), \ldots, \mathsf{type}_{\mathcal{X}}(s^0)) \subseteq \\ \subseteq \mathsf{comp}(\mathsf{type}_{\mathcal{X}}(u^k), \mathsf{type}_{\mathcal{X}}(u^{k-1}), \ldots, \mathsf{type}_{\mathcal{X}}(u^0)) = \mathsf{type}_{\mathcal{X}}(v^k).\end{aligned}$$

- Assume that $r = (X \supseteq \delta Y)$. By definition of a wf-rule, $\mathsf{lev}(Y) = 0$. Decompose $R = R_1 \circ R_2$ where R_1 has length 1. As in the above case, we obtain a run S_1 from u of length 1, performing transition δ, such that the types of $R_1(1)$ and $S_1(1)$ are appropriately contained (for all levels), and that R_1 and S_1 have the same initial and final states. Then we apply the induction assumption tor $R_2 \in Y$ and obtain a run $S_2 \in Y$ from $S_1(1)$, such that the types at the end of R_2 and S_2 are contained as required (for all levels), and that the final states are the same. Thus, $S := S_1 \circ S_2$ satisfies the claim.
- Finally, assume that $r = (X \supseteq \delta Y Z)$. By definition of a wf-rule, $\mathsf{lev}(Z) = 0$. Decompose $R = R_1 \circ R_2$ where R_1 performs the transition δ followed by a run from Y, and R_2 is in Z. Since $\mathcal{X}$ is described by a well-formed set of rules, R_1 is in the set described by $X_{\delta Y}$. Using part 1 of Theorem 3.1 for R_1 and $X_{\delta Y}$, there is a run S_1 from u which performs δ followed by a run from Y such that R_1 and S_1 have the same initial and final states. Decompose the pds of $R_1(|R_1|)$ and $S_1(|S_1|)$ as $s'^n : s'^{n-1} : \cdots : s'^0$ and $u'^n : u'^{n-1} : \cdots : u'^0$. Recall from the proof of part 1 of Theorem 3.1 (see page 15) that $\mathsf{type}_{\mathcal{X}}(s'^i) \subseteq \mathsf{type}_{\mathcal{X}}(u'^i)$ for $\mathsf{lev}(Y) + 1 \leq i \leq n$ (notice that $\mathsf{lev}(X_{\delta Y}) = \mathsf{lev}(Y)$). But by definition of a wf-rule we know that the topmost $\mathsf{lev}(Y)$-stack of $R_1(0)$ and of $R_1(|R_1|)$ are the same, so $s'^i = s^i$ for $0 \leq i \leq \mathsf{lev}(Y)$; for the same reason $u'^i = u^i$ for $0 \leq i \leq \mathsf{lev}(Y)$. Thus, $\mathsf{type}_{\mathcal{X}}(s'^i) \subseteq \mathsf{type}_{\mathcal{X}}(u'^i)$ for $0 \leq i \leq n$. Then we apply the induction assumption to $R_2 \in Z$ and obtain a run $S_2 \in Z$ from $S_1(|S_1|)$ such that the types at the end of R_2 and S_2 are contained as required (for all levels) and such that the final states are the same. Thus, $S := S_1 \circ S_2$ satisfies the claim.

$\square$

Types in previous papers. A similar concept of defining types were already present in [13] and [15]. In both these papers types were used only for systems without collapse. The types in [13] are defined completely semantically: the definition is similar to our Lemma B.5. Then it is necessary to prove that the type of s^l does not depend on the choice of $s^n, s^{n-1}, \ldots, s^{l+1}$ present in the assumptions of the lemma. We were unable to give a proof of the analogous fact for systems with collapse.

The types in [15] are much more similar to our types: they are also defined syntactically, i.e. basing on possible transitions of the system. But these types were defined only for one class of runs, namely for k-returns for each k. The generalisation to an arbitrary family described by wf-rules required mainly the invention of a proper definition of these rules. The generalisation to systems with collapse required mainly the invention of a proper definition of stacks (i.e. that a 0-stack should keep the copy of the linked stack, instead of just a link).

C Types of Stacks—Proofs

This appendix is devoted to the proof of Lemma B.5. We assume that the family $\mathcal{X}$ is fixed, and we write type for $\mathsf{type}_{\mathcal{X}}$. We will first prove the left-to-right implication of this lemma, as restated below (in a slightly stronger version).

Lemma C.1. *Let $\widehat{\sigma} \in \mathcal{D}_X$ for some $X \in \mathcal{X}$, and let $0 \leq l \leq \mathsf{lev}(X)$. Let R be a run which agrees with $\widehat{\sigma}$, where $R(0) = (q_0, s^n : s^{n-1} : \dots : s^l)$. Then*

$$(\mathsf{type}(s^n), \mathsf{type}(s^{n-1}), \dots, \mathsf{type}(s^{l+1}), q_0, \widehat{\sigma}) \in \mathsf{type}(s^l).$$

The proof is by induction on the length of R. It is divided into several lemmas; the division follows the steps in the definition of types, i.e., we prove certain properties of the functions T and U which finally allow to prove the lemma. We start with two observations. The first follows immediately from the definitions and the second is a corollary of Lemma B.10.

Proposition C.2. *Let $1 \leq k \leq n$, and let s^k be a k-stack. Then $\mathsf{ne} \in \mathsf{type}(s^k)$ if and only if s^k is not empty.*

Proposition C.3. *Let $0 \leq l \leq k \leq n$, and let $s = s^k : s^{k-1} : \dots : s^l$ be a k-stack. The type of s, $\mathsf{type}(s)$, is $\mathsf{comp}(\mathsf{type}(s^k), \mathsf{type}(s^{k-1}), \dots, \mathsf{type}(s^l))$.*

The next lemma proves our intuition about T. This lemma uses the "big" induction assumption, i.e., Lemma C.1 for shorter runs.

Lemma C.4. *Let R be a run which agrees with some $\widehat{\sigma} \in \mathcal{D}_X$, and let $r = (X \supseteq \delta X_1 \dots X_m) \in \mathcal{R}_{\mathcal{X}}^{>0}$ be a rule which describes R. Assume that the statement of Lemma C.1 is true for all runs strictly shorter than R. Decompose the stack of $R(0)$ as $s^n : s^{n-1} : \dots : s^0$, and the stack of $R(1)$ as $t^n : t^{n-1} : \dots : t^{M(r)}$. Then $T(r)$ contains the tuple*

$$\eta = \Big(\big(\mathsf{type}(s^n), \dots, \mathsf{type}(s^0)\big), \big(\mathsf{type}(t^n), \mathsf{type}(t^{n-1}), \dots, \mathsf{type}(t^{M(r)})\big), \widehat{\sigma}\Big).$$

Proof. Let $\widehat{\sigma} = (X, \Omega^n, \Omega^{n-1}, \dots, \Omega^{\mathsf{lev}(X)+1}, q')$ and q_1 the state of $R(1)$, which is also the state reached after application of δ. We distinguish three cases depending on the form of r, i.e., on the value of $m \in \{0, 1, 2\}$.

1. Assume that r is $X \supseteq \delta$. Then $|R| = 1$ whence the state of $R(|R|)$ is q_1 and the stack of $R(|R|)$ is $t^n : t^{n-1} : \dots : t^{\mathsf{lev}(X)}$ (recall that $\mathsf{lev}(X) = M(r)$). Since R agrees with $\widehat{\sigma}$, $q' = q_1$ and $\Omega^i \subseteq \mathsf{type}(t^i)$ for $\mathsf{lev}(X) + 1 \leq i \leq n$. Due to Point 1 of Definition B.11, $\eta \in T(r)$.

2. Assume that r is $X \supseteq \delta X_1$. Let $\widehat{\tau} = (X_1, \Omega^n, \Omega^{n-1}, \dots, \Omega^{\mathsf{lev}(X_1)+1}, q')$ where $\Omega^i = \emptyset$ for $\mathsf{lev}(X_1)+1 \leq i \leq \mathsf{lev}(X)$ (and the other Ω^i are specified by $\widehat{\sigma}$). We know that $R{\restriction}_{1,|R|}$ is in X_1 whence it agrees with $\widehat{\tau}$. Application of Lemma C.1 to the shorter run $R{\restriction}_{1,|R|}$, to $\widehat{\tau}$ and to $\mathsf{lev}(X_1)$ yields

$$\big(\mathsf{type}(t^n), \mathsf{type}(t^{n-1}), \dots, \mathsf{type}(t^{\mathsf{lev}(X_1)+1}), q_1, \widehat{\tau}\big) \in \mathsf{type}\left(t^{\mathsf{lev}(X_1)}\right).$$

Recall that $M(r) = \mathsf{lev}(X_1)$. Due to Point 2 of Definition B.11, $\eta \in T(r)$.

3. Assume that r is $X \supseteq \delta X_1 X_2$. Fix some $1 \leq i \leq |R|$ such that $R{\restriction}_{1,i} \in X_1$ and $R{\restriction}_{i,|R|} \in X_2$. Let $u^n : u^{n-1} : \dots : u^0$ be the stack of $R(i)$, and q_2 its state. Let

$$\begin{aligned}\widehat{\tau} &= (X_1, \mathsf{type}(u^n), \mathsf{type}(u^{n-1}), \dots, \mathsf{type}(u^{\mathsf{lev}(X_1)+1}), q_2),\\ \widehat{\rho} &= (X_2, \Omega^n, \Omega^{n-1}, \dots, \Omega^{\mathsf{lev}(X_2)+1}, q'),\end{aligned}$$

where $\Omega^i = \emptyset$ for $\mathsf{lev}(X_2) + 1 \leq i \leq \mathsf{lev}(X)$. The subrun $R{\restriction}_{1,i}$ agrees with $\widehat{\tau}$, and the subrun $R{\restriction}_{i,|R|}$ agrees with $\widehat{\rho}$. Application of Lemma C.1 to the shorter run $R{\restriction}_{1,i}$, to $\widehat{\tau}$ and to $\mathsf{lev}(X_1)$ yields

$$(\mathsf{type}(t^n), \mathsf{type}(t^{n-1}), \dots, \mathsf{type}(t^{\mathsf{lev}(X_1)+1}), q_1, \widehat{\tau}) \in \mathsf{type}(t^{\mathsf{lev}(X_1)}).$$

Analogously, application of the lemma to $R{\restriction}_{i,|R|}$, to $\widehat{\rho}$ and to 0 yields

$$(\mathsf{type}(u^n), \mathsf{type}(u^{n-1}), \dots, \mathsf{type}(u^1), q_2, \widehat{\rho}) \in \mathsf{type}(u^0).$$

The definition of a wf-rule implies that the topmost $\mathsf{lev}(X_1)$-stacks of $R(0)$ and $R(i)$ coincide whence $\mathsf{type}(s^i) = \mathsf{type}(u^i)$ for $\mathsf{lev}(X_1) \geq i \geq 0$. Thus, Point 3 of Definition B.11 (with $\Psi^i = \mathsf{type}(s^i)$, $\Phi^i = \mathsf{type}(t^i)$ and $\Sigma^i = \mathsf{type}(u^i)$), implies that $\eta \in T(r)$.

□

Having related the sets $T(r)$ with the subruns starting after the first transition of some run described by r, we now relate the function U_z with the first operation of such a run. Recall that $\mathsf{stype}_{\mathcal{X}}$ is the fixpoint of the sequence $(\mathsf{stype}_z)_{z\in\mathbb{N}}$ which is reached at some $z' \in \mathbb{N}$, i.e., $\mathsf{stype}_{z'} = \mathsf{stype}_{z'-1} = \mathsf{stype}_{\mathcal{X}}$. For the next lemma we fix this value z'.

Lemma C.5. *Let $s = s^n : s^{n-1} : \dots : s^0$ and $t = t^n : t^{n-1} : \dots : t^{L(op)}$ be pds such that s^0 contains a link of level K to a stack u^K. Assume that $t = op(s)$ for some operation op. For all $z > z'$, $U_z(op, K, \mathsf{type}(u^K))$ contains*

$$\begin{aligned}\eta = \Big(&\big(\mathsf{type}(s^n), \mathsf{type}(s^{n-1}), \dots, \mathsf{type}(s^0)\big),\\ &\big(\mathsf{type}(t^n), \mathsf{type}(t^{n-1}), \dots, \mathsf{type}(t^{L(op)})\big)\Big).\end{aligned}$$

Proof. The proof is by case distinction on *op*. Fix $z > z'$

1. Assume that $op = \mathsf{pop}^k$. Then we have $t = s^n : s^{n-1} : \cdots : s^k$, so $t^i = s^i$ for $k \leq i \leq n$ (recall that $L(op) = k$). In particular s^k is not empty, so $\mathsf{ne} \in \mathsf{type}(s^k)$. Due to Point 1 of Definition B.13, we conclude that $\eta \in U_z(op, K, \mathsf{type}(u^K))$.
2. Assume that $op = \mathsf{col}^k$. In this case, $k = K$ and u^K is not empty (equivalently: $\mathsf{ne} \in \mathsf{type}(u^K)$) because otherwise col^k would not be applicable. We have $t = s^n : s^{n-1} : \cdots : s^{k+1} : u^k$, so $t^i = s^i$ for $k+1 \leq i \leq n$, and $t^k = u^k$ (recall that $L(op) = k$). Due to Point 2 of Definition B.13, we conclude that $\eta \in U_z(op, K, \mathsf{type}(u^K))$.
3. Assume that $op = \mathsf{push}^1_{b,k}$. Then

$$t = s^n : s^{n-1} : \cdots : s^2 : (s^1 : s^0) : (b, k, s^k)$$

whence $t^i = s^i$ for $2 \leq i \leq n$, $t^1 = s^1 : s^0$ and $t^0 = (b, k, s^k)$ (recall that $L(op) = 0$). Due to Proposition C.3, $\mathsf{type}(t^1) = \mathsf{comp}(\mathsf{type}(s^1), \mathsf{type}(s^0))$. Additionally,

$$\mathsf{type}(t^0) = \mathsf{stype}(b, k, \mathsf{type}(s^k)) = \mathsf{stype}_{z'}(b, k, \mathsf{type}(s^k)).$$

Thus, using Point 3 of Definition B.13 we conclude that $\eta \in U_z(op, K, \mathsf{type}(u^K))$.
4. Finally, assume that $op = \mathsf{push}^k$ for $k \geq 2$. Then we have

$$\begin{aligned} &t = s^n : s^{n-1} : \cdots : s^{k+1} : t^k : s^{k-1} : \cdots : s^0 \text{ where} \\ &t^k = (s^k : s^{k-1} : \cdots : s^0). \end{aligned}$$

Thus, $t^i = s^i$ for $0 \leq i \leq n$ with $i \neq k$ (recall that $L(op) = 0$). Proposition C.3 implies that

$$\mathsf{type}(t^k) = \mathsf{comp}(\mathsf{type}(s^k), \mathsf{type}(s^{k-1}), \ldots, \mathsf{type}(s^0)).$$

Thus, using Point 4 of Definition B.13, we conclude that $\eta \in U_z(op, K, \mathsf{type}(u^K))$.

□

We are now prepared to prove Lemma C.1.

Proof (of Lemma C.1). Let R be a run with $R(0) = s^n : s^{n-1} : \cdots : s^l$ that agrees with

$$\sigma = (\mathsf{type}(s^n), \mathsf{type}(s^{n-1}), \ldots, \mathsf{type}(s^{l+1}), q_0, \widehat{\sigma}).$$

We make an external induction on the length of R and an internal induction on l.

– Assume that $l = 0$ and $|R| = 0$. Let $s^0 = (a, K, u^K)$. Since R agrees with $\widehat{\sigma}$, we have $\widehat{\sigma} = (X, \Omega^n, \Omega^{n-1}, \ldots, \Omega^{\mathsf{lev}(X)+1}, q_0)$ where $\Omega^i \subseteq \mathsf{type}(s^i)$ for $\mathsf{lev}(X)+1 \leq i \leq n$ (and q_0 is the state of $R(0)$). Due to Point 1 of Definition B.14, $\sigma \in \mathsf{stype}_z(a, K, \mathsf{type}(u^K))$ for every $z \in \mathbb{N}$ whence $\sigma \in \mathsf{type}(s^0)$.

- Assume that $l = 0$ and $|R| > 0$. Then there is a rule $r = (X \supseteq \delta X_1 \dots X_m) \in \mathcal{R}_{\mathcal{X}}^{>0}$ describing R. Let $s^0 = (a, K, u^K)$. We have $\delta = (q_0, a, \cdot, \cdot, op)$. Let $t^n : t^{n-1} : \dots : t^{L(op)}$ be the stack of $R(1)$. Lemma C.5 implies that for all $z > z'$ the set $U_z(op, K, \mathsf{type}(u^K))$ contains

  $$\Big((\mathsf{type}(s^n), \mathsf{type}(s^{n-1}), \dots, \mathsf{type}(s^0)), (\mathsf{type}(t^n), \mathsf{type}(t^{n-1}), \dots, \mathsf{type}(t^{L(op)}))\Big).$$

  Setting $v^{M(r)} = t^{M(r)} : t^{M(r)-1} : \dots : t^{L(op)}$, the stack of $R(1)$ is $t^n : t^{n-1} : \dots : t^{M(r)+1} : v^{M(r)}$. Our induction assumption on shorter runs and Lemma C.4 implies that $T(r)$ contains

  $$\Big(\big(\mathsf{type}(s^n), \mathsf{type}(s^{n-1}), \dots, \mathsf{type}(s^0)\big), \big(\mathsf{type}(t^n), \mathsf{type}(t^{n-1}), \dots, \mathsf{type}(t^{M(r)+1}), \mathsf{type}(v^{M(r)})\big), \widehat{\sigma}\Big).$$

  Additionally, Proposition C.3 implies that

  $$\mathsf{type}(v^{M(r)}) = \mathsf{comp}\Big(\mathsf{type}(t^{M(r)}), \mathsf{type}(t^{M(r)-1}), \dots, \mathsf{type}(t^{L(op)})\Big).$$

  Due to Point 2 of Definition B.14, $\sigma \in \mathsf{stype}_z(a, K, \mathsf{type}(u^K))$ for $z > z'$ whence $\sigma \in \mathsf{type}(s^0)$.
- Assume that $l \geq 1$. Decompose $s^l = t^l : t^{l-1}$. The (inner) induction assumption implies that $\mathsf{type}(t^{l-1})$ contains the tuple

  $$(\mathsf{type}(s^n), \mathsf{type}(s^{n-1}), \dots, \mathsf{type}(s^{l+1}), \mathsf{type}(t^l), q_0, \widehat{\sigma}).$$

  From Definition B.7, it follows that $\sigma \in \mathsf{comp}(\mathsf{type}(t^l), \mathsf{type}(t^{l-1})) = \mathsf{type}(s^l)$.

□

The rest of this appendix deals with the right-to-left implication of Lemma B.5. In the proof we use the notion of *having a witness*. The intuition is that a stack and a run descriptor have a witness, if this right-to-left implication holds for them. Our goal is to prove that every such pair has a witness, which means that the implication is always true.

Definition C.6. Let $0 \leq k \leq n$, let s^k be a k-stack, and let $\Phi^k \subseteq \mathcal{T}^k$. We define when (s^k, Φ^k) *has a witness* by induction on k, starting with $k = n$. We say that (s^k, Φ^k) has a witness if (s^k, σ) has a witness for every $\sigma \in \Phi^k$, as defined below.

- (s^k, ne) has a witness if $\mathsf{ne} \in \mathsf{type}(s^k)$ (equivalently: if $k \geq 1$ and s^k is nonempty).
- For

  $$\sigma = (\Phi^n, \Phi^{n-1}, \dots, \Phi^{k+1}, p, \widehat{\sigma}) \text{ and}$$
  $$\widehat{\sigma} = (X, \Omega^n, \Omega^{n-1}, \dots, \Omega^{\mathsf{lev}(X)+1}, q),$$

  (s^k, σ) has a witness if

- $\sigma \in \mathsf{type}(s^k)$ and
- for each configuration $c = (p, t^n : \cdots : t^{k+1} : s^k)$ such that (t^i, Φ^i) has a witness for each $k+1 \leq i \leq n$ there is a run R_c from c to some stack $u^n : u^{n-1} : \cdots : u^0$ such that R_c agrees with $\widehat{\sigma}$ and (u^i, Ω^i) has a witness for each $\mathsf{lev}(X) + 1 \leq i \leq n$.

We first prove that composers preserve witnesses.

Proposition C.7. *Let $0 \leq l \leq k \leq n$. For each $l \leq i \leq k$ let s^i be an i-stack, and let $\Phi^i \subseteq \mathcal{T}^i$ be such that (s^i, Φ^i) has a witness. Then $(s^k : s^{k-1} : \cdots : s^l, \mathsf{comp}(\Phi^k, \Phi^{k-1}, \ldots, \Phi^l))$ has a witness.*

Proof. We have to show that for each $\sigma^k \in \mathsf{comp}(\Phi^k, \Phi^{k-1}, \ldots, \Phi^l)$, $(s^k : s^{k-1} : \cdots : s^l, \sigma^k)$ has a witness. By Proposition C.3 $\sigma^k \in \mathsf{type}(s^k : s^{k-1} : \cdots : s^l)$. If $\sigma^k = \mathsf{ne}$ we are already done. Otherwise, $\sigma^k = (\Sigma^n, \ldots, \Sigma^{k+1}, p, \widehat{\sigma})$ for some $\widehat{\sigma} = (X, \Omega^n, \ldots, \Omega^{\mathsf{lev}(X)+1}, q)$. By definition of the composer, Φ^l contains a tuple $\sigma^l = (\Sigma^n, \Sigma^{n-1} \ldots, \Sigma^{l+1}, p, \widehat{\sigma})$ such that $\Sigma^i \subseteq \Phi^i$ for $l+1 \leq i \leq k$. Let $c = (p, t^n : \cdots : t^{k+1} : s^k : \cdots : s^l)$ be a configuration for stacks t^i such that (t^i, Σ^i) has a witness for each $k+1 \leq i \leq n$. By assumption of the lemma, also (s^i, Σ^i) has a witness for $l+1 \leq i \leq k$ (since $\Sigma^i \subseteq \Phi^i$), and (s^l, σ^l) has a witness (since $\sigma^l \in \Phi^l$). Application of Definition C.6 to (s^l, σ^l) shows that there is a run R from c to some configuration $(q, u^n : \cdots : u^0)$ which agrees with $\widehat{\sigma}$ and such that (u^i, Ω^i) has a witness for $lev(X) + 1 \leq i \leq n$. Thus, R also shows that $(s^k : \cdots : s^l, \sigma^k)$ has a witness. ☐

Our next goal is to show that each 0-stack has a witness.

Lemma C.8. *Let $z \in \mathbb{N}$, $1 \leq K \leq n$, u^K a K-stack, and $\Sigma^K \subseteq \mathcal{T}^K$ such that (u^K, Σ^K) has a witness. Let $a \in \Gamma$, and let $\sigma \in \mathsf{stype}_z(a, K, \Sigma^K)$. Then $((a, K, u^K), \sigma)$ has a witness.*

Corollary C.9. *Let $1 \leq K \leq n$, u^K be a K-stack, $\Sigma^K \subseteq \mathcal{T}^K$ be such that (u^K, Σ^K) has a witness, and $a \in \Gamma$. $\big((a, K, \Sigma^K), \mathsf{type}((a, K, \Sigma^K))\big)$ has a witness.*

The proof of the lemma is by induction on the fixpoint stage z. As an auxiliary step we show how the set $T(r)$ can be used to prove that there is an appropriate run described by r.

Lemma C.10. *Let $r = (X \supseteq \delta X_1 \ldots X_m)$ be a rule from $\mathcal{R}_{\mathcal{X}}^{>0}$, and let $\widehat{\sigma} = (X, \Omega^n, \Omega^{n-1}, \ldots, \Omega^{\mathsf{lev}(X)+1}, q') \in \mathcal{D}_X$. Let R_1 be a run of length 1 from stack $s^n : s^{n-1} : \cdots : s^0$ to stack $t^n : t^{n-1} : \cdots : t^{M(r)}$ performing the transition δ. For $0 \leq i \leq n$, let Ψ^i be such that (s^i, Ψ^i) has a witness, and for $M(r) \leq i \leq n$, let Φ^i be such that (t^i, Φ^i) has a witness. Assume that*

$$((\Psi^n, \Psi^{n-1}, \ldots, \Psi^0), (\Phi^n, \Phi^{n-1}, \ldots, \Phi^{M(r)}), \widehat{\sigma}) \in T(r).$$

Then there exists a run R from $R_1(0)$ which agrees with $\widehat{\sigma}$, and ends in a stack $v^n : v^{n-1} : \cdots : v^0$ such that (v^i, Ω^i) has a witness for $\mathsf{lev}(X) + 1 \leq i \leq n$.

Proof. Let q_1 be the state of $R_1(1)$, which is also the state reached after application of δ. We distinguish three cases depending on the shape of r, i.e., on the value of $m \in \{0, 1, 2\}$.

1. Assume that r is $X \supseteq \delta$. Recall that $M(r) = \mathsf{lev}(X)$. Set $R = R_1$. By definition of the set $T(r)$, we have $q_1 = q'$, and $\Omega^i \subseteq \Phi^i$ for $\mathsf{lev}(X) + 1 \leq i \leq n$. Notice that (t^i, Ω^i) has a witness (in particular $\Omega^i \subseteq \mathsf{type}(t^i)$) for $\mathsf{lev}(X) + 1 \leq i \leq n$, because (t^i, Φ^i) has a witness, and $\Omega^i \subseteq \Phi^i$. Observe that R_1 is in X whence it agrees with $\widehat{\sigma}$.
2. Assume that r is $X \supseteq \delta X_1$. Recall that $M(r) = \mathsf{lev}(X_1)$. By definition of $T(r)$, we have
$$\tau = (\Phi^n, \Phi^{n-1}, \dots, \Phi^{\mathsf{lev}(X_1)+1}, q_1, \widehat{\tau}) \in \Phi^{\mathsf{lev}(X_1)} \text{ where}$$
$$\widehat{\tau} = (X_1, \Omega^n, \Omega^{n-1}, \dots, \Omega^{\mathsf{lev}(X_1)+1}, q').$$
Since $(t^{\mathsf{lev}(X_1)}, \tau)$ has a witness and (t^i, Φ^i) has a witness for $\mathsf{lev}(X_1)+1 \leq i \leq n$, there is a run R_2 agreeing with $\widehat{\tau}$ from $R_1(1)$ to a stack $v^n : v^{n-1} : \dots : v^0$ such that (v^i, Ω^i) has a witness for $\mathsf{lev}(X_1)+1 \leq i \leq n$. As R we take $R_1 \circ R_2$; this run is in X. By definition of a wf-rule we know that $\mathsf{lev}(X_1) \leq \mathsf{lev}(X)$, so R agrees with $\widehat{\sigma}$, and (v^i, Ω^i) has a witness for $\mathsf{lev}(X) + 1 \leq i \leq n$.
3. Assume that r is $X \supseteq \delta X_1 X_2$. Recall that $M(r) = \mathsf{lev}(X_1)$. We have
$$\tau = (\Phi^n, \Phi^{n-1}, \dots, \Phi^{\mathsf{lev}(X_1)+1}, q_1, \widehat{\tau}) \in \Phi^{\mathsf{lev}(X_1)}$$
for some $\widehat{\tau} = (X_1, \Sigma^n, \Sigma^{n-1}, \dots, \Sigma^{\mathsf{lev}(X_1)+1}, q_2)$ and
$$\rho = (\Sigma^n, \dots, \Sigma^{\mathsf{lev}(X_1)+1}, \Psi^{\mathsf{lev}(X_1)}, \dots, \Psi^1, q_2, \widehat{\rho}) \in \Psi^0$$
where $\widehat{\rho} = (X_2, \Omega^n, \Omega^{n-1}, \dots, \Omega^{\mathsf{lev}(X_2)+1}, q')$. Since $(t^{\mathsf{lev}(X_1)}, \tau)$ has a witness, and (t^i, Φ^i) has a witness for $\mathsf{lev}(X_1) + 1 \leq i \leq n$, there is a run R_2 agreeing with $\widehat{\tau}$ from $R_1(1)$ to a stack $u^n : u^{n-1} : \dots : u^0$ such that (u^i, Σ^i) has a witness for $\mathsf{lev}(X_1) + 1 \leq i \leq n$. By definition of a wf-rule we know that the topmost $\mathsf{lev}(X_1)$-stack of $R_2(|R_2|)$ is the same as of $R_1(0)$ whence $u^i = s^i$ and (u^i, Ψ^i) has a witness for $0 \leq i \leq \mathsf{lev}(X_1)$. In particular (u^0, ρ) has a witness. Hence, there is a run R_3 agreeing with $\widehat{\rho}$ from $R_2(|R_2|)$ to a stack $v^n : v^{n-1} : \dots : v^0$ such that (v^i, Ω^i) has a witness for $\mathsf{lev}(X_2) + 1 \leq i \leq n$. As R we take $R_1 \circ R_2 \circ R_3$; this run is in X. By definition of a wf-rule we know that $\mathsf{lev}(X_2) \leq \mathsf{lev}(X)$, so R agrees with $\widehat{\sigma}$, and (v^i, Ω^i) has a witness for $\mathsf{lev}(X) + 1 \leq i \leq n$.

□

The next lemma shows how the set $U_z(op, K, \Sigma^K)$ can be used to prove that an appropriate run performing operation *op* exists.

Lemma C.11. *Fix a number $z \geq 1$ and assume that Lemma C.8 holds for $z-1$. Let $s = s^n : s^{n-1} : \dots : s^0$ be a pds, where s^0 contains a link of level K to a* *stack u^K, and let op be an operation. For $0 \leq i \leq n$, let Ψ^i be such that (s^i, Ψ^i) has a witness; let also Σ^K be such that (u^K, Σ^K) has a witness. Assume that*

$$\big((\Psi^n, \Psi^{n-1}, \dots, \Psi^0), (\Pi^n, \Pi^{n-1}, \dots, \Pi^{L(op)})\big) \in U_z(r, K, \Sigma^K).$$

Then op can be applied to s, and $op(s) = t^n : t^{n-1} : \dots : t^{L(op)}$ is such that (t^i, Π^i) has a witness for $L(op) \leq i \leq n$.

Proof. We proceed by case distinction on the operation op performed by δ.

- Assume that $op = \mathsf{pop}^k$. Then $L(op) = k$, $\Pi^i = \Psi^i$ for $k \leq i \leq n$ and $\mathsf{ne} \in \Psi^k$. Thus s^k is not empty, so pop^k can be applied to s, which results in the stack $s^n : s^{n-1} : \dots : s^k$. We have $(t^i, \Pi^i) = (s^i, \Psi^i)$ for $k \leq i \leq n$ whence (t^i, Π^i) has a witness.
- Assume that $op = \mathsf{col}^k$. Then $L(op) = k$, $k = K$, $\mathsf{ne} \in \Sigma^K$, $\Pi^i = \Psi^i$ for $k+1 \leq i \leq n$ and $\Pi^k \subseteq \Sigma^K$. Thus, u^k is not empty whence col^k can be applied to s. This results in the stack $s^n : s^{n-1} : \dots : s^{k+1} : u^K$. We have $(t^i, \Pi^i) = (s^i, \Psi^i)$ for $k+1 \leq i \leq n$. Moreover, $t^k = u^K$ and $\Pi^k \subseteq \Sigma^K$ whence (t^i, Π^i) has a witness for $k \leq i \leq n$.
- Assume that $op = \mathsf{push}^1_{b,k}$. Then $L(op) = 0$, $\Pi^i = \Psi^i$ for $2 \leq i \leq n$, $\Pi^i = \mathsf{comp}(\Psi^1, \Psi^0)$, and $\Pi^0 \subseteq \mathsf{stype}_{z-1}(b, k, \Psi^k)$. Additionally,

  $$\mathsf{push}^1_{b,k}(s) = s^n : s^{n-1} : \dots : s^2 : (s^1 : s^0) : (b, k, s^k).$$

  For $2 \leq i \leq n$, we have $(t^i, \Pi^i) = (s^i, \Psi^i)$ whence (t^i, Π^i) has a witness. Due to Proposition C.7, $(t^1, \Pi^1) = (s^1 : s^0, \mathsf{comp}(\Psi^1, \Psi^0))$ has a witness. Since we assumed that Lemma C.8 holds for $z-1$, we conclude that $(t^0, \mathsf{stype}_{z-1}(b, k, \Psi^k))$ has a witness whence (t^0, Π^0) has a witness.
- Assume that $op = \mathsf{push}^k$ (for $k \geq 2$). Then $L(op) = 0$, $\Pi^i = \Psi^i$ for $0 \leq i \leq n$ with $k \neq i$, and $\Pi^k = \mathsf{comp}(\Psi^k, \Psi^{k-1}, \dots, \Psi^0)$. Additionally,

  $$\mathsf{push}^k(s) = s^n : s^{n-1} : \dots : s^{k+1} : t^k : s^{k-1} : \dots : s^0$$

  where $t^k = s^k : s^{k-1} : \dots : s^0$. For $0 \leq i \leq n$ with $i \neq k$ we have $(t^i, \Pi^i) = (s^i, \Psi^i)$ whence (t^i, Π^i) has a witness. Due to Proposition C.7,

  $$(t^k, \Pi^k) = (s^k : s^{k-1} : \dots : s^0, \mathsf{comp}(\Psi^k, \Psi^{k-1}, \dots, \Psi^0))$$

  has a witness.

□

Proof (Lemma C.8). The proof is by induction on z. Recall that we defined $\mathsf{stype}_{-1}(a, K, \Sigma^k) = \emptyset$. Let $z \geq 0$, $\sigma \in \mathsf{stype}_z(a, K, \Sigma^K)$ and $s^0 = (a, K, u^K)$. Assume that we have already proved the lemma for $z-1$. By definition, $\sigma \in \mathsf{type}(s^0)$. Let

$$\sigma = (\Psi^n, \Psi^{n-1}, \dots, \Psi^1, q_0, \widehat{\sigma}), \quad \text{and}$$
$$\widehat{\sigma} = (X, \Omega^n, \Omega^{n-1}, \dots, \Omega^{\mathsf{lev}(X)+1}, q').$$

Let $c = (q_0, s^n : \cdots : s^1 : s^0)$ be a configuration such that (s^i, Ψ^i) has a witness for each $1 \leq i \leq n$. We have construct a run R from c to a stack $w^n : w^{n-1} : \cdots : w^0$ such that R agrees with $\widehat{\sigma}$ and (w^i, Ω^i) has a witness for each $\mathsf{lev}(X) + 1 \leq i \leq n$. We distinguish two cases.

- Assume that σ is in $\mathsf{stype}_z(a, K, \Sigma^K)$ thanks to the first point of Definition B.14. Then $\Omega^i \subseteq \Psi^i$ for $\mathsf{lev}(X) + 1 \leq i \leq n$, and $q' = q_0$. It follows that (s^i, Ω^i) has a witness (whence in particular $\Omega^i \subseteq \mathsf{type}(s^i)$) for $\mathsf{lev}(X) + 1 \leq i \leq n$ and the run R of length 0 from c agrees with $\widehat{\sigma}$.
- Assume that σ is in $\mathsf{stype}_z(a, K, \Sigma^K)$ thanks to the second point of Definition B.14. Then for some rule $r = (X \supseteq \delta X_1 \dots X_m) \in \mathcal{R}^{>0}_{\mathcal{X}}$, where $\delta = (q_0, a, \cdot, \cdot, op)$, we have

$$\begin{aligned}
&\big((\Psi^n, \Psi^{n-1}, \dots, \Psi^0),\\
&\ (\Pi^n, \Pi^{n-1}, \dots, \Pi^{L(op)})\big) \in U_z(op, K, \Sigma^K),\\
&\big((\Psi^n, \Psi^{n-1}, \dots, \Psi^0),\\
&\ (\Pi^n, \Pi^{n-1}, \dots, \Pi^{M(r)+1}, \Phi^{M(r)}), \widehat{\sigma}\big) \in T(r),\\
&\Psi^0 \subseteq \mathsf{stype}_{z-1}(a, K, \Sigma^K), \text{ and}\\
&\Phi^{M(r)} = \mathsf{comp}(\Pi^{M(r)}, \Pi^{M(r-1)}, \dots, \Pi^{L(op)}).
\end{aligned}$$

  By induction assumption, (s^0, Ψ^0) has a witness. Notice that the state and the topmost symbol of c are as required by δ. Lemma C.11 implies that δ can be applied to c. Let d be the resulting configuration and $t^n : t^{n-1} : \cdots : t^{L(op)}$ its stack. Furthermore, this lemma implies that (t^i, Π^i) has a witness for $L(op) \leq i \leq n$. Setting $v^{M(r)} = t^{M(r)} : t^{M(r)-1} : \cdots : t^{L(op)}$ the stack of d is $t^n : t^{n-1} : \cdots : t^{M(r)+1} : v^{M(r)}$. Due to Proposition C.7, $(v^{M(r)}, \Phi^{M(r)})$ has a witness. Thus, Lemma C.10 can be applied (where as R_1 we take the run from c to d). We obtain a run R from c which agrees with $\widehat{\sigma}$, and ends in a stack $w^n : w^{n-1} : \cdots : w^0$ such that (w^i, Ω^i) has a witness for $\mathsf{lev}(X) + 1 \leq i \leq n$ as required.

□

Corollary C.12. *Let $0 \leq k \leq n$, let s^k be a k-stack, and let $\Phi^k \subseteq \mathsf{type}(s^k)$. Then (Φ^k, s^k) has a witness.*

Proof. It is enough to prove this corollary for $\Phi^k = \mathsf{type}(s^k)$. We just make an induction on the structure of the stack. Assume that $k = 0$ and let $s^0 = (a, K, t^K)$. From the induction assumption we know that $(t^K, \mathsf{type}(t^K))$ has a witness. Using Corollary C.9 we obtain that $(s^0, \mathsf{type}(s^0))$ has a witness. If $k > 0$ and s^k is empty, $\mathsf{type}(s^k) = \emptyset$, whence the claim is trivial. Let now $k > 0$ and let s^k be nonempty. Decompose $s^k = t^k : t^{k-1}$. By definition, $\mathsf{type}(s^k) = \mathsf{comp}(\mathsf{type}(t^k), \mathsf{type}(t^{k-1}))$. By induction assumption $(t^k, \mathsf{type}(t^k))$ and $(t^{k-1}, \mathsf{type}(t^{k-1}))$ have witnesses. Using Proposition C.7 we conclude that $(t^k, \mathsf{type}(t^k))$ has also a witness. □

With this corollary, we can prove the right-to-left implication of Lemma B.5.

Proof (of Lemma B.5). Assume that there is a $\sigma = (\Psi^n, \Psi^{n-1}, \ldots, \Psi^{l+1}, p, \widehat{\sigma}) \in \mathsf{type}(s^l)$ such that $\Psi^i \subseteq \mathsf{type}(s^i)$ for $l+1 \leq i \leq n$. Application of the corollary shows that (s^i, Ψ^i) has a witness for $l+1 \leq i \leq n$, and (s^l, σ) has a witness. Thus, there is a run from $(p, s^n : s^{n-1} : \cdots : s^l)$ which agrees with $\widehat{\sigma}$ as required. The other direction has already been proved (see Proof of Lemma C.1 on page 23). □

D Runs, Positions and the History Function

In this section we give a technical analysis of runs and introduce the history function which is useful to describe certain sets of runs. Appendix E relies on the results developed here.

D.1 Positions and Histories of Stacks

In this section we first introduce *positions* of i-stacks in a k-stack for $i \leq k$. These positions allow to access each substack contained in a stack. Afterwards we introduce the *history function.* Given a run R and a position x in the final stack of the run, this function determines the origin of this position in the first stack of R, i.e., it returns a position y such that the stack at position x in the last stack of R was created from the stack at position y in the first stack of R. For a k-stack s let us denote by $|s|$ its *size*, i.e. the number of $(k-1)$-stacks s consists of.

Definition D.1. For each stack s of level k (where $1 \leq k \leq n$) we define the set of positions in s as follows.

If $k = 1$, a *simple position* in s is a number $x^1 \in \mathbb{N}$ such that $x^1 \leq |s|$.

If $k \geq 2$, a *simple position* in s is either a tuple $(0, \ldots, 0)$ of length k, or a tuple $(x^k, \ldots, x^1)$ where $1 \leq x^k \leq |s|$ and $(x^{k-1}, \ldots, x^1)$ is a simple position in the x^k-th $(k-1)$-stack of s (counted bottom up).

We say that a simple position x *points to a k-stack* if $k \in \mathbb{N}$ is maximal such that x ends in a sequence of 0's of length k.

A *position* in s is either a simple position in s or a sequence $x := x_0 \xrightarrow{k} y$ such that x_0 is a simple position pointing to a 0-stack (a, k, t^k) in s and y is a position in t^k, but $y \neq (0, \ldots, 0)$.[10] A position x *points to a k-stack* if its rightmost simple position points to a k-stack.

For x, y positions in s we say that *y points into the stack at x* (abbreviated *y points into x*) if

[10] We forbid nonsimple positions ending in the simple position $(0, 0, \ldots, 0)$ because of the following interpretation. In a 0-stack $s^0 = (a, k, t^k)$ we actually do not consider t^k to be a k-stack but only the *content* of a k-stack. In this interpretation the application of col^k when s^0 is the topmost 0-stack does not replace the topmost k-stack by t^k but the *content* of the topmost k-stack by the *content* of t^k. This difference is only of syntactical nature but it is useful to exclude such positions when defining the history function.

1. either x points to a level 0 stack and $y = x \xrightarrow{k} z$ or
2. x points to a level $k \geq 1$ stack, x and y agree on all entries where x is nonzero, and $y \neq x$, i.e., y extends the position x where x starts to be constantly 0.

Let s be some n-stack where s^i denotes the topmost i-stack of s. The *position of the topmost k-stack of s* is $\mathsf{top}^k(s) := (|s^n|, \ldots, |s^{k+1}|, 0, \ldots, 0)$.

Finally, we define the nesting rank of a position. This rank counts the number of simple positions involved in the position. Let $\mathsf{nr}(x) := 0$ if x is simple, and $\mathsf{nr}(x \xrightarrow{k} z) := 1 + \mathsf{nr}(x) + \mathsf{nr}(z)$.

Remark D.2. We use the notation $z \xrightarrow{k} z'$ where z is a non-simple position of a 0-stack that links to a k-stack and z' points to some position inside this linked stack.

We next introduce the *history function*. This function is useful for giving semantical characterisations of the sets in the family $\mathcal{X}$ defined by a grammar in Section 5.

Definition D.3. Let R be a run from stack s to stack t and let x be a position in t. If $|R| = 0$, then $\mathsf{hist}(x, R) := x$. If $|R| = 1$, we make a case distinction on the operation performed by R, and on the form of x.

– If R performs a $\mathsf{push}^1_{a,k}$ operation and $x = \mathsf{top}^0(t) = (x^n, \ldots, x^1)$, then

$$\mathsf{hist}(x, R) := (x^n, \ldots, x^2, x^1 - 1).$$

– If R performs a $\mathsf{push}^1_{a,k}$ operation and x is of the form $\mathsf{top}^0(t) \xrightarrow{k} (y^k_1, \ldots, y^1_1)$, then we set

$$\mathsf{hist}(x, R) := (|u^n|, \ldots, |u^{k+1}|, y^k_1, \ldots, y^1_1)$$

for u^i the topmost i-stack of t.

– If R performs a $\mathsf{push}^1_{a,k}$ operation and x is of the form $\mathsf{top}^0(t) \xrightarrow{k} (y^k_1, \ldots, y^1_1) \xrightarrow{k'} z$, then

$$\mathsf{hist}(x, R) := (|u^n|, \ldots, |u^{k+1}|, y^k_1, \ldots, y^1_1) \xrightarrow{k'} z$$

where u^i is the topmost i-stack of t.

– If R performs a push^i operation for $2 \leq i \leq n$ and x is of the form $(x^n, \ldots, x^1)$ such that $(x^n, \ldots, x^1)$ is $\mathsf{top}^{i-1}(t)$ or points into $\mathsf{top}^{i-1}(t)$, then

$$\mathsf{hist}(x, R) := (x^n, \ldots, x^i, x^{i-1} - 1, x^{i-2}, \ldots, x^1)$$

– If R performs a push^i operation for $2 \leq i \leq n$ and x is of the form $(x^n, \ldots, x^1) \xrightarrow{k} z$ such that x points into $\mathsf{top}^{i-1}(t)$, then

$$\mathsf{hist}(x, R) := (x^n, \ldots, x^i, x^{i-1} - 1, x^{i-2}, \ldots, x^1) \xrightarrow{k} z.$$

- If R performs a col^k operation and $x = (x_0^n, \dots, x_0^1)$ points into[11] $\mathsf{top}^k(t)$, then

$$\mathsf{hist}(x, R) := \mathsf{top}^0(s) \xrightarrow{k} (x_0^k, \dots, x_0^1).$$

- If R performs a col^k operation and $x = (x_0^n, \dots, x_0^1) \xrightarrow{k'} y$ points into $\mathsf{top}^k(t)$, then

$$\mathsf{hist}(x, R) := \mathsf{top}^0(s) \xrightarrow{k} (x_0^k, \dots, x_0^1) \xrightarrow{k'} y.$$

- In all other cases, we set $\mathsf{hist}(x, R) := x$.

If $|R| \geq 2$, we decompose $R = S \circ T$ where $|S| = 1$, and we set $\mathsf{hist}(x, R) := \mathsf{hist}(\mathsf{hist}(x, T), S)$.

Remark D.4. Our intuition of the history function is the following. $\mathsf{hist}(x, R)$ is the (unique) position of a k-stack in s from which R created the k-stack at x in t in the sense that the stack at x in t is a (possibly modified) copy of the stack at $\mathsf{hist}(x, R)$ in s not only in terms of content but also in the way it was produced by R.

Due to the inductive definition of the history function it is compatible with decomposition of runs in the following sense.

Proposition D.5. *Let R, S, T be runs such that $R = S \circ T$. If x, y are positions such that $\mathsf{hist}(x, T) = y$, then $\mathsf{hist}(x, R) = z$ if and only if $\mathsf{hist}(y, S) = z$.*

D.2 Basic Properties of Runs

In this section we collect useful properties of runs of collapsible pushdown systems and of the history function.

A careful look at the definition of the history function shows that a k-stack can be changed only if it is the topmost one, and only by an operation of level at most k.

Proposition D.6. *Let R be a run of length 1, and x a position of some k-stack in $R(1)$. Let t^k be the stack at x in $R(1)$ and s^k the stack at $\mathsf{hist}(x, R)$ in $R(0)$. Then exactly one of the following holds.*

- *$x = \mathsf{top}^k(R(1))$ and the operation in R is of level below k. In this case we have $\mathsf{hist}(x, R) = \mathsf{top}^k(R(0))$ and $|t^k| = |s^k|$.*
- *$x = \mathsf{top}^k(R(1))$ and the operation in R is of level k. In this case we have $\mathsf{hist}(x, R) = \mathsf{top}^k(R(0))$ and*
  - *$|t^k| = |s^k| - 1$ if the operation is pop^k,*
  - *$|t^k| < |s^k|$ if the operation is col^k, and*
  - *$|t^k| = |s^k| + 1$ if the operation is push^k.*

[11] Recall that $\mathsf{top}^k(t)$ does *not* point into $\mathsf{top}^k(t)$.

– $s^k = t^k$ *and if* y *points into* x*, then* $\mathsf{hist}(y, R)$ *points to the same position in* $\mathsf{hist}(x, R)$ *as* y *in* x*.*

We have an analogous property for longer runs which follows by straightforward induction on the length of the run.

Corollary D.7. *Let* R *be a run of length* m *and* x_0 *a position of some* k*-stack in* $R(m)$ *such that*

1. $\mathsf{hist}(x_0, R{\restriction}_{i,m}) \neq \mathsf{top}^k(R(i))$ *for all* $0 \leq i < m$*, or*
2. $\mathsf{hist}(x_0, R{\restriction}_{i,m}) \neq \mathsf{top}^k(R(i))$ *for all* $0 < i < m$*, and* $m \geq 2$*.*

Then the k*-stack at* $\mathsf{hist}(x_0, R)$ *is equal to the* k*-stack at* x_0 *in* $R(m)$*. Moreover, if* x *points into* x_0*, then* $\mathsf{hist}(x, R)$ *points to the same position in* $\mathsf{hist}(x_0, R)$ *as* x *in* x_0*.*

Similarly, the relationship of k-stacks that are next to each other is preserved unless the lower one becomes the topmost stack.

Proposition D.8. *Let* R *be a run, and* x, y *positions such that* x *points to a* k*-stack that is directly below the* k*-stack to which* y *points (in the same* $(k+1)$*-stack), i.e.,* x *and* y *differ only on the last non-zero coordinate by* 1 *(these positions are not required to be simple). Assume that*

1. $\mathsf{hist}(x, R{\restriction}_{m,|R|}) \neq \mathsf{top}^k(R(m))$ *for all* $0 \leq m \leq |R|$*, or*
2. $\mathsf{hist}(y, R{\restriction}_{m,|R|}) \neq \mathsf{top}^k(R(m))$ *for all* $0 \leq m < |R|$*.*

Then $\mathsf{hist}(x, R)$ *points to a* k*-stack that is directly below the* k*-stack to which* $\mathsf{hist}(y, R)$ *points.*

Proof. First assume that $|R| = 1$. For almost every operation in R, the history function behaves in the same way for two neighbouring k-stacks. The only exception is push^{k+1} if $\mathsf{hist}(x, R) = \mathsf{hist}(y, R) = \mathsf{top}^k(R(0))$. But this case is forbidden by our assumptions. For $|R| \geq 2$, note that the claim is compatible with compositions of runs, so we conclude by induction on the length of the run. $\square$

Let x and y be positions such that y points into x. Intuitively, the history function should preserve this containment because if a stack s is a copy of some other stack t then every stack of lower level in this stack was created from some stack of lower level inside of t. The next lemma provides a formal statement of this kind.

Proposition D.9. *Let* R *be some run and* x *a position of a* k*-stack. Let* y *point into* x *such that for* $l := \mathsf{nr}(y) - \mathsf{nr}(x)$ *the last* l *links in* y *are of level at most* k*.*[12] *Then* $\mathsf{hist}(y, R)$ *points into* $\mathsf{hist}(x, R)$ *and for* $l' := \mathsf{nr}(\mathsf{hist}(y, R)) - \mathsf{nr}(\mathsf{hist}(x, R))$ *the last* l' *links in* $\mathsf{hist}(y, R)$ *are of level at most* k*.*

---

[12] In other words, if x decomposes as $x = \hat{x} \xrightarrow{k'} x'$ for a simple position x', then all decompositions of y as $y = \hat{x} \xrightarrow{k'} y' \xrightarrow{k''} y''$ satisfy $k'' \leq k$.

Proof. For $|R| = 0$ there is nothing to prove. For $|R| = 1$ the claim follows directly from a tedious but straightforward case distinction on the operation performed by R. The general case then follows by induction: if $|R| \geq 2$ we can decompose $R = R_1 \circ R_2$ such that by induction hypothesis $y' := \mathsf{hist}(y, R_2)$ points into $x' := \mathsf{hist}(x, R_2)$ and we can apply the lemma again to R_1, x' and y'. $\square$

Corollary D.10. *Let $j > k$. For every run R, $\mathsf{hist}(\mathsf{top}^k(R(|R|)), R)$ points into $\mathsf{hist}(\mathsf{top}^j(R(|R|)), R)$. Additionally, if $\mathsf{hist}(\mathsf{top}^k(R(|R|)), R) = \mathsf{top}^k(R(0))$ then $\mathsf{hist}(\mathsf{top}^j(R(|R|)), R) = \mathsf{top}^j(R(0))$.*

According to our intuition that the history function tells us the original copy from which a stack was created, history can only decrease a position (with respect to the lexicographic ordering $\preceq$). On the other hand, when a position is always present in a stack, the history should point to the same position. The next two lemmas prove this intuition.

Lemma D.11. *Let R be a run and x a position in the final stack of R such that x_0 is the simple prefix of x, i.e., x_0 is a simple position such that there is a position x' with $x = x_0 \xrightarrow{k} x'$. If $y := \mathsf{hist}(x, R)$ is a simple position, $y \preceq x_0$.*

Lemma D.12. *Let R be a run and let $x := \mathsf{top}^k(R(0))$. If x is present in all configurations of R then $\mathsf{hist}(x, R) = x$ and for all $i \leq |R|$ $\mathsf{hist}(x, R{\restriction}_{i,|R|})$ is simple if and only if $\mathsf{hist}(x, R{\restriction}_{i,|R|}) = x$.*

In the rest of this section we prove these two Lemmas. For the proofs we use functions pack_i. Let $x = x_0 \xrightarrow{k_1} x_1 \dots \xrightarrow{k_m} x_m$ with $m = \mathsf{nr}(x)$. For $0 \leq i \leq m$ we define a simple position $\mathsf{pack}_i(x)$ as follows.

- $\mathsf{pack}_0(x) := x_0$ and
- for $i \geq 1$, $\mathsf{pack}_i(x)$ is obtained from $\mathsf{pack}_{i-1}(x)$ by replacing its last k_i coordinates by x_i.

Note that pack_1 is closely related to the col^k and the $\mathsf{push}^1_{a,k}$ operations: if R is a run of length 1 performing col^k and x points into the topmost k-stack of $R(1)$, then $\mathsf{pack}_{i+1}(\mathsf{hist}(x, R)) = \mathsf{pack}_i(x)$ for all $0 \leq i \leq \mathsf{nr}(x)$. On the other hand, if R is of length 1 performing $\mathsf{push}^1_{a,k}$ and $x = \mathsf{top}^0(R(1)) \xrightarrow{k} x_1$, then $\mathsf{pack}_i(x) = \mathsf{pack}_{i-1}(\mathsf{hist}(x, R))$ for all $1 \leq i \leq \mathsf{nr}(x)$.

In the following, for a simple position $z = (z^n, z^{n-1}, \dots, z^1)$ we call z^k the *level k coordinate* of z. Furthermore, we write $x \prec_k y$ for simple positions x, y if $x \prec y$ and the first coordinate on which they differ is the level k coordinate.

Lemma D.13. *Let R be a run and $x = x_0 \xrightarrow{k_1} x_1 \dots \xrightarrow{k_m} x_m$ a position in the final stack of R. Assume that $y := \mathsf{hist}(x, R)$ is a simple position. Then*

$$y \preceq \mathsf{pack}_m(x) \prec_{k_m} \mathsf{pack}_{m-1}(x) \prec_{k_{m-1}} \dots \prec_{k_1} \mathsf{pack}_0(x).$$

Proof. The proof is by induction of the length of R. If $|R| = 0$ the claim is trivial (as $x = y$ and $\mathsf{nr}(x) = 0$). For $|R| \geq 1$, let $S = R{\upharpoonright}_{|R|-1,|R|}$, and let $z = \mathsf{hist}(x, S)$.

The induction assumption, applied for $R{\upharpoonright}_{0,|R|-1}$ and for $z = z_0 \xrightarrow{k'_1} z_1 \dots \xrightarrow{k'_{m'}} z_{m'}$, gives us that

$$y \preceq \mathsf{pack}_{m'}(z) \prec_{k_{m'}} \mathsf{pack}_{m'-1}(z) \prec_{k_{m'-1}} \dots \prec_1 \mathsf{pack}_0(z).$$

We analyse the cases of the definition of the history function, for run S.

- If S performs a $\mathsf{push}^1_{a,k}$ operation and $x = \mathsf{top}^0(S(1))$, then $m = m' = 0$ and $x = \mathsf{pack}_0(x) \succeq z = \mathsf{pack}_0(z) \succeq y$.
- If S performs a $\mathsf{push}^1_{a,k}$ operation, and x points into $\mathsf{top}^0(S(1))$ we already remarked that $\mathsf{pack}_i(x) = \mathsf{pack}_{i-1}(z)$ and $m = m' + 1$. Thus, we immediately conclude that

  $$y \preceq \mathsf{pack}_m(x) \prec_{k_m} \mathsf{pack}_{m-1}(x) \dots \prec_{k_2} \mathsf{pack}_1(x).$$

  Note that for s^k the topmost k-stack of $S(0)$, x_1 points to a position in $\mathsf{pop}^k(s^k)$ while x_0 points into $\mathsf{top}^k(s)$. Thus, the level k coordinate of $\mathsf{pack}_0(x)$ is $|s^k|$ while the corresponding coordinate in $\mathsf{pack}_1(x)$ has value at most $|s^k| - 1$. Thus, $\mathsf{pack}_1(x) \prec_k \mathsf{pack}_0(x)$.
- If S performs a push^i operation and x is $\mathsf{top}^{i-1}(S(1))$ or points into $\mathsf{top}^{i-1}(S(1))$, then every coordinate of z_0 is either the same or smaller than the same coordinate of x_0 (and the rest of x and z is the same). Thus, we conclude immediately from the properties of z that x also satisfies the claim.
- If S performs a col^k operation, and x points into the topmost k-stack of $S(1)$. Then, $\mathsf{pack}_i(x) = \mathsf{pack}_{i+1}(z) \succeq y$ for all $i \leq m$, and $m = m' - 1$. Thus, the claim follows trivially.
- If none of the previous cases applies, then $z = x$ and there is nothing to show.

□

From the previous lemma we can easily deduce Lemma D.11.

Proof (Lemma D.11). Due to Lemma D.13, $y \preceq \mathsf{pack}_{\mathsf{nr}(x)}(x) \preceq \mathsf{pack}_0(x) = x_0$. □

We also obtain the following corollary of Lemma D.13.

Corollary D.14. *Let s be some pds and $0 \leq k \leq n$. If x be a position in s such that $\mathsf{pack}_{\mathsf{nr}(x)}(x)$ points to a k-stack and $\mathsf{pack}_{\mathsf{nr}(x)}(x) \succeq \mathsf{top}^k(s)$, then $x = \mathsf{top}^k(s)$.*

Proof. Decompose $x = x_0 \xrightarrow{k_1} x_1 \dots \xrightarrow{k_m} x_m$ and $z := \mathsf{pack}_m(x)$. Consider any n-CPS such that there is a run R from the initial configuration $(q_0, \bot_n)$ to the pds s (recall that such a run exists by definition of a pds. Since $\bot_n$ only contains simple positions, $y := \mathsf{hist}(x, R)$ is simple. Application of Lemma D.13 gives us $\mathsf{top}^k(s) \preceq z \preceq x_0$, so x_0 points into the topmost k-stack of s. This implies that coordinates of levels greater than k of $\mathsf{top}^k(s)$, z, and x_0 agree whence $z = \mathsf{top}^k(s)$.

If $m = 0$, $x = z = \mathsf{top}^k(s)$ and we are done. Heading towards a contradiction assume that $m \geq 1$. Let j be the maximum of all k_i. The last k coordinates of $z = \mathsf{top}^k(s)$ are 0. Since the last k_m coordinates of z are $x_m \neq (0, \dots, 0)$ (by definition of a position), we see that $j \geq k_m > k$. Thus z points to or into $\mathsf{top}^{j-1}(s)$, whence its level j coordinate is the size of the topmost j-stack of s. Since each application of pack preserves the coordinates of level above j, x also points into $\mathsf{top}^j(s)$ whence its level j coordinate is bounded by the size of the topmost j-stack of s. Now Lemma D.13 implies that the level j coordinate in $\mathsf{pack}_{\mathsf{nr}(x)}(x)$ is smaller than that in x which is a contradiction. □

Finally, we prepare the proof of Lemma D.12 with the following lemma.

Lemma D.15. *Let R be a run, and let x be a position of $R(|R|)$, and let $y := \mathsf{hist}(x, R)$. If $\mathsf{pack}_{\mathsf{nr}(x)}(x)$ is present in all configurations of R, then $\mathsf{pack}_{\mathsf{nr}(y)}(y) = \mathsf{pack}_{\mathsf{nr}(x)}(x)$.*

Proof (Lemma D.15). If we prove the lemma for runs of length 1, the whole claim follows by a simple induction on the length of a run. Let R be a run of length 1. The proof is by case distinction on the definition of the history function.

- Assume that R performs a $\mathsf{push}^1_{a,k}$ operation, and $x = \mathsf{top}^0(R(1))$. Then $x = pack_{\mathsf{nr}(x)}$ is not present in $R(0)$ whence there is nothing to show.
- Assume that R performs a push^j operation, and x is $\mathsf{top}^{j-1}(R(1))$ or points into $\mathsf{top}^{j-1}(R(1))$. Let $x = x_0 \xrightarrow{k_1} x_1 \dots \xrightarrow{k_m} x_m$. If $k_i < j$ for all i, also $\mathsf{pack}_{\mathsf{nr}(x)}(x)$ is $\mathsf{top}^{j-1}(R(1))$ or points into $\mathsf{top}^{j-1}(R(1))$ (as then x and $\mathsf{pack}_{\mathsf{nr}(x)}(x)$ are equal on all coordinates of level at least j). But this would mean that $\mathsf{pack}_{\mathsf{nr}(x)}(x)$ was not present in $R(0)$; thus $k_i \geq j$ for some i (in particular $m \geq 1$). Notice that $y = y_0 \xrightarrow{k_1} x_1 \dots \xrightarrow{k_m} x_m$, where y_0 differs from x_0 only on the level j coordinate. This coordinate does not appear in $\mathsf{pack}_m(x)$ whence $\mathsf{pack}_{\mathsf{nr}(y)}(y) = \mathsf{pack}_{\mathsf{nr}(x)}(x)$.
- In the remaining three cases we easily see (from the definition of the history) that $\mathsf{pack}_{\mathsf{nr}(y)}(y) = \mathsf{pack}_{\mathsf{nr}(x)}(x)$.

□

Proof (Lemma D.12). For a simple position $x = \mathsf{top}^k(R(0))$ we have $\mathsf{pack}_{\mathsf{nr}(x)}(x) = x$. For arbitrary $i \leq |R|$, let $y_i := \mathsf{hist}(x, R{\restriction}_{i,|R|})$. We apply Lemma D.15 for $R{\restriction}_{i,|R|}$ and obtain $\mathsf{pack}_{\mathsf{nr}(y_i)}(y_i) = \mathsf{top}^k(R(0))$. If y_i is simple, this implies $y = x$. Corollary D.14 implies that $y_0 = x$. □

E A Family of Sets of Runs

In this appendix we prove that the sets defined in Section 5 satisfy the (informal) claims we formulated. In fact, our proof goes from intuition (which is made precise using the history function) to grammars: First, using the history function we give alternative definitions of pumping runs, top^k-non-erasing runs, k-returns and k-colreturns and show that they satisfy the intuition given before. We then show that these runs are actually described by the grammars we presented in Section 5.

E.1 Characterisation of Returns and Colreturns

We start with a definition of returns. In Lemmas E.6 and E.16 we later see that the grammar from Section 5 correctly describes the sets of returns.

Definition E.1. A run R of length m is called *k-return* (where $1 \leq k \leq n$) if

- $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R)$ points to the second topmost $(k-1)$-stack[13] in the topmost k-stack of $R(0)$, and
- $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R{\restriction}_{i,m}) \neq \mathsf{top}^{k-1}(R(i))$ for all $1 \leq i \leq m-1$.

The following propositions confirm our intuition about k-returns.

Proposition E.2. *The last operation of a k-return R is pop^k or col^k.*

Proof. Let $m := |R|$. Note that in order to satisfy

$$\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R{\restriction}_{m-1,m}) \neq \mathsf{top}^{k-1}(R(m-1))$$

the last operation of R is pop^j or col^j with $j \geq k$. Heading for a contradiction assume that $j > k$. It follows that

$$\mathsf{hist}(\mathsf{top}^{k}(R(m)), R{\restriction}_{m-1,m}) \neq \mathsf{top}^{k}(R(m-1)).$$

Let $i \leq m-2$ be maximal such that

$$x_i := \mathsf{hist}(\mathsf{top}^{k}(R(m)), R{\restriction}_{i,m}) = \mathsf{top}^{k}(R(i))$$

Such i exists because $i = 0$ is of this form (cf. Corollary D.10). Due to Corollary D.7 (variant 2), $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R{\restriction}_{i,m})$ points to the topmost $(k-1)$-stack of the k-stack to which x_i points. Thus, it points to the topmost $(k-1)$-stack of $R(i)$. But this contradicts the definition of a return. $\square$

Proposition E.3. *For every k-return R, the topmost k-stack of $R(0)$ after removing its topmost $(k-1)$-stack is equal to the topmost k-stack of $R(|R|)$. If x points into $\mathsf{top}^k(R(|R|)$ then $\mathsf{hist}(x, R)$ points to the same position in the stack at $\mathsf{top}^k(R(0))$.*

Proof. Let $m := |R|$ and let l be the size of the topmost k-stack of $R(m)$. For $1 \leq i \leq l$, let t_i^{k-1} be the i-th $(k-1)$-stack (counting from the bottom) of the topmost k-stack of $R(m)$. Let x_i be the position pointing to t_i^{k-1}. By the above proposition, R ends in pop^k or col^k. Note that the histories of $x_1, \dots, x_l$ with respect to $R{\restriction}_{m-1,m}$ point to $(k-1)$-stacks, where the history of x_1 points to the bottommost $(k-1)$-stack of some k-stack, x_i is directly below the history of x_{i+1} for each $1 \leq i < l$ and none of these histories point to the topmost $(k-1)$-stack of $R(m-1)$. Note that non-topmost $(k-1)$-stacks in the same k-stack are always

[13] Whenever we write "the second topmost $(k-1)$-stack" we assume that it is in the same k-stack as the topmost $(k-1)$-stack, i.e., we assume that the topmost k-stack has size at least 2.

treated the same way by the history function. Thus, a simple induction on the operations performed by $R{\restriction}_{0,m-1}$ shows that this property is preserved by the history function, i.e., $\mathsf{hist}(x_1, R), \mathsf{hist}(x_2, R), \ldots, \mathsf{hist}(x_l, R)$ point to the first l $(k-1)$-stacks of a k-stack. But by definition $\mathsf{hist}(x_l, R) = \mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R)$ is the second topmost $(k-1)$-stack of $\mathsf{top}^k(R(0))$. Application of Corollary D.7 (variant 1) shows that the $(k-1)$-stack at x_i in $R(m)$ is the same as the $(k-1)$-stack at $\mathsf{hist}(x_i, R)$ in $R(0)$. This proves the first part of this proposition. Similarly, Corollary D.7 implies the preservation of pointers into $\mathsf{top}^k(R(m))$ which completes the proof. □

Corollary E.4. *For every run R which starts with a push^k operation (including arbitrary $\mathsf{push}^1_{a,l}$ for $k = 1$), and continues with a k-return, the topmost k-stacks of $R(0)$ and of $R(|R|)$ coincide. Additionally, if x points into $\mathsf{top}^k(R(|R|)$ then $\mathsf{hist}(x, R)$ points to the same position in the stack at $\mathsf{top}^k(R(0))$.*

Recall that wf-rules of the form $X \supseteq \delta Y Z$ must satisfy the property that whenever R is a composition of a one-step run performing transition δ with a run from Y, then the topmost $\mathsf{lev}(Y)$-stack of $R(0)$ and $R(|R|)$ are the same. Notice that in the grammars in Section 5 such rules appear only when δ performs a push operation of some level k, and X is a set of k-returns. Since $\mathsf{lev}(X) = k$, the above corollary proves this property.

We now give a definition of k-colreturns. Lemmas E.7 and E.16 show that, for such definitions, the grammar from Section 5 correctly describes the sets of colreturns. As already mentioned, the intuition of the definition is the following. A k-colreturn is a run whose last transition is col^k from a stack where the topmost symbol is a copy of the topmost symbol of the first stack.

Definition E.5. A run R of length m is called *k-colreturn* (where $1 \leq k \leq n$) if

- $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R)$ is of the form $\mathsf{top}^0(R(0)) \xrightarrow{k} x$, where x is simple, and
- $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R{\restriction}_{i,m})$ is not simple for all $0 \leq i \leq m-1$.

We first prove a decomposition result of k-returns and k-colreturns into one transition followed by a sequence of shorter returns or colreturns. Later we deal with the change levels.

Lemma E.6. *Let R be some run. Then R is a k-return if and only if R is of one of the following forms.*

1. *$|R| = 1$ and R performs pop^k.*
2. *R starts with an operation of level at most $k-1$, and continues with a k-return.*
3. *R starts with $\mathsf{push}^1_{a,k}$ and continues with a k-colreturn.*
4. *R starts with a push^j for $j > k$ and continues with a k-return.*
5. *R starts with a push^j for $j \geq k$ (including $\mathsf{push}^1_{a,l}$ for $j = k = 1$) and decomposes as $R = S \circ T \circ U$ where S has length 1, T is a j-return and U is a k-return.*

Lemma E.7. *Let R be some run. Then R is a k-colreturn if and only if R is of one of the following forms*

1. *R has length 1 and performs col^k.*
2. *R starts with push^j for some $j \geq 2$ and continues with a k-colreturn.*
3. *R starts with a push^j (including $\mathsf{push}^1_{a,l}$ for $j = 1$) and decomposes as $R = S \circ T \circ U$ where S has length 1, T is a j-return and U is a k-colreturn.*

Before we start the proof of these two lemmas, we state some auxiliary claims. First, we observe that the history function either manipulates the simple prefix of a position, or adds a simple prefix, or removes it. This will be useful while analysing k-colreturns.

Proposition E.8. *Let R be some run of length m. Let $x \xrightarrow{k} y$ be a position in $R(m)$ such that*

$$\mathsf{nr}(\mathsf{hist}(x \xrightarrow{k} y, R{\restriction}_{i,m})) > \mathsf{nr}(\mathsf{hist}(y, R))$$

for all $i \leq m$. Let $x' = \mathsf{hist}(x, R)$. Then $\mathsf{hist}(x \xrightarrow{k} y, R) = x' \xrightarrow{k} y$ (neither x nor x' have to be simple). Additionally, the 0-stack of $R(m)$ at position x is the same as the 0-stack of $R(0)$ at position x'.

Proof. Induction on m. For $m = 1$ we just analyse all cases. For $m \geq 2$ we observe that the claim for any decomposition $R = S \circ T$ follows from the claim for S and for T. ☐

Corollary E.9. *Let R be a run of length $m \geq 2$, and x a simple position in $R(m)$ such that $\mathsf{hist}(x, R)$ is simple, but $\mathsf{hist}(x, R{\restriction}_{i,m})$ is not simple for $1 \leq i \leq m-1$. Then, for some k, R starts with $\mathsf{push}^1_{a,k}$ and ends with col^k. Additionally, $x = \mathsf{top}^{k-1}(R(m))$ if and only if $\mathsf{hist}(x, R)$ is the second topmost $(k-1)$-stack of $R(0)$.*

Proof. The last operation of R has to be a col^k for some k, because otherwise $\mathsf{hist}(x, R{\restriction}_{m-1,m})$ would be simple. Then $\mathsf{hist}(x, R{\restriction}_{m-1,m})$ is of the form $\mathsf{top}^0(R(m-1)) \xrightarrow{k} x'$. Proposition E.8, applied for $R{\restriction}_{1,m-1}$, shows that $\mathsf{hist}(x, R{\restriction}_{1,m})$ is of the form $z \xrightarrow{k} x'$, and that the topmost 0-stack of $R(m-1)$, and the 0-stack in $R(1)$ to which z points to are the same k-stack u^k, which is in fact the topmost k-stack of $R(m)$. Because $\mathsf{hist}(z \xrightarrow{k} x', R{\restriction}_{0,1})$ is simple, necessarily $z = \mathsf{top}^0(R(1))$, the first operation of $R(0)$ is $\mathsf{push}^1_{a,k}$, and the topmost k-stack of $R(0)$ after removing its topmost $(k-1)$-stack is equal to u^k. Additionally, x points to the same position in the topmost k-stack of $R(m)$, as $\mathsf{hist}(x, R)$ in the topmost k-stack of $R(0)$. Thus $x = \mathsf{top}^{k-1}(R(m))$ if and only if $\mathsf{hist}(x, R)$ is the second topmost k-stack of $R(0)$. ☐

The following lemma proves the intuition that k-colreturns make a copy of the topmost stack symbol and finally use its collapse link of level k (the proof is almost the same as that of the previous corollary).

Lemma E.10. *Let R be a colreturn. Then the topmost k-stack of $R(|R|)$ is equal to the k-stack contained in the topmost 0-stack of $R(0)$. In particular its size is smaller than the size of the topmost k-stack of $R(0)$. Additionally, the last operation of R is col^k, and $\mathsf{hist}(\mathsf{top}^k(R(|R|)), R) = \mathsf{top}^k(R(0))$.*

Proof. Let $m := |R|$, and let $x := \mathsf{top}^{k-1}(R(m))$. The last operation of R has to be a col^j for some j, because otherwise $\mathsf{hist}(x, R{\restriction}_{m-1,m})$ would be simple. Then $\mathsf{hist}(x, R{\restriction}_{m-1,m})$ is of the form $\mathsf{top}^0(R(m-1)) \xrightarrow{j} x'$. Proposition E.8, applied for $R{\restriction}_{0,m-1}$, implies that $\mathsf{hist}(x, R) = \mathsf{hist}(\mathsf{top}^0(R(m-1)), R{\restriction}_{0,m-1}) \xrightarrow{j} x'$. By definition of a k-colreturn it follows that $j = k$ and $\mathsf{hist}(\mathsf{top}^0(R(m-1)), R{\restriction}_{0,m-1}) = \mathsf{top}^0(R(0))$. From this proposition we also conclude that the topmost 0-stack of $R(0)$ and the topmost 0-stack of $R(m-1)$ store the same k-stack u^k, which is in fact the topmost k-stack of $R(m)$. Of course the size of u^k is smaller than the size of the the topmost k-stack of $R(0)$ because col^k must decrease the size of the topmost k-stack (see Remark 2.4). Corollary D.10 implies that $\mathsf{hist}(\mathsf{top}^k(R(m-1)), R{\restriction}_{0,m-1}) = \mathsf{top}^k(R(0))$. Since the last operation is col^k, $\mathsf{hist}(\mathsf{top}^k(R(m-1)), R) = \mathsf{top}^k(R(0))$. □

The next two propositions describe which operations are allowed as the first operation of a k-return and of a k-colreturn.

Proposition E.11. *Let R be a k-return. The first operation of R is neither col^j for $j \geq k$ nor pop^j for $j > k$. If the first operation of R is pop^k, then $|R| = 1$.*

Proof. Let $m := |R|$. If the first operation of a run R is col^j, $j \geq k$ then by definition of the history function $\mathsf{hist}(x, R{\restriction}_{0,1})$ does not point to any simple position inside $\mathsf{top}^j(R(0))$ for all positions x in $R(1)$. Thus, also $\mathsf{hist}(x, R)$ does not point to any simple position inside $\mathsf{top}^j(R(0))$ for all positions x in $R(m)$. But if R is a k-return, $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R)$ is a simple position and points into $\mathsf{top}^k(R(0))$ whence it also points into $\mathsf{top}^j(R(0))$. Analogously, one shows that R does not start with pop^j for $j > k$.

If a k-return R starts with pop^k, it follows that $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R{\restriction}_{1,m}) = \mathsf{top}^{k-1}(R(1))$. But this is not allowed if $1 \leq m - 1$. □

Proposition E.12. *Let R be a k-colreturn. The first operation of R is a push or col^k. If the first operation of R is col^k, then $|R| = 1$.*

Proof. Let $m := |R|$. If the first operation of a run R is a pop then by definition of the history function $\mathsf{hist}(x, R{\restriction}_{0,1})$ does not point into $\mathsf{top}^0(R(0))$ for all positions x in $R(1)$. Thus also $\mathsf{hist}(x, R)$ does not point into $\mathsf{top}^0(R(0))$ for all positions x in $R(m)$, in particular for $x = \mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R)$. This contradicts the definition of a k-colreturn.

In $R(0)$ we have a position $\mathsf{top}^0(R(0)) \xrightarrow{k} x$. Thus, the only collapse operation which can be performed at $R(0)$ is a level k collapse, i.e., col^k. If R starts with col^k, then $\mathsf{hist}(y, R{\restriction}_{0,1}) = \mathsf{top}^0(R(0)) \xrightarrow{k} x$ for some simple x only if y is simple. We conclude that $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R{\restriction}_{1,m})$ is simple which implies $m = 1$. □

We state a last auxiliary lemma nd then prove Lemmas E.6 and E.7.

Lemma E.13. *Let R be a k-return of length at least 2. Let x be the position $\mathsf{hist}(\mathsf{top}^{k-1}(R(|R|)), R{\restriction}_{1,|R|})$. Then one of the following holds.*

1. *x points to the second topmost $(k-1)$-stack of $R(1)$ and the first stack operation of R is of level strictly below k or a push^j for $j > k$.*
2. *x points to the third topmost $(k-1)$-stack of $R(1)$ and the first stack operation of R is a push of level k.*
3. *There is a $j > k$ such that x points to the second topmost $(k-1)$-stack of the second topmost $(j-1)$-stack of $R(1)$ and the first stack operation of R is push^j.*
4. *$x = \mathsf{top}^0(R(1)) \xrightarrow{k} \mathsf{top}^{k-1}(u^k)$, the first stack operation of R is $\mathsf{push}^1_{a,k}$ and the topmost 0-stack of $R(1)$ is (a, k, u^k).*

Proof. Since R is a k-return, Proposition D.5 implies that $\mathsf{hist}(x, R{\restriction}_{0,1})$ points to the second topmost $(k-1)$-stack of $R(0)$. We proceed by case distinction on the stack operation of $S := R{\restriction}_{0,1}$. Due to Proposition E.11 we only have to consider the following cases.

- Assume that S performs a pop^j operation for $j < k$, or a col^j operation for $j < k$, or a push^j operation for $2 \leq j < k$, or a $\mathsf{push}^1_{a,j}$ operation for $j \neq k > 1$. Then x necessarily points to the second topmost $(k-1)$-stack of $R(1)$ (because S makes changes only inside the topmost $(k-1)$-stack of $R(0)$).
- Assume that S performs a push^k for $k \geq 2$, or a $\mathsf{push}^1_{a,j}$ for $j \neq k = 1$. Then x necessarily points to the third topmost $(k-1)$-stack of $R(1)$.
- Assume that S performs a push^j operation with $j > k$. Then either x points to the second topmost $(k-1)$-stack of $R(1)$, or to the second topmost $(k-1)$-stack of the second topmost $(j-1)$-stack of $R(1)$.
- Assume that S performs $\mathsf{push}^1_{a,k}$, and $k \geq 2$. Then either x points to the second topmost $(k-1)$-stack of $R(1)$, or x is of the form $\mathsf{top}^0(R(1)) \xrightarrow{k} \mathsf{top}^{k-1}(u^k)$ where u^k is the k-stack stored in the topmost 0-stack of $R(1)$.
- Assume that S performs $\mathsf{push}^1_{a,k}$, and $k = 1$. Then either x points to the third topmost $(k-1)$-stack of $R(1)$, or x is of the form $\mathsf{top}^0(R(1)) \xrightarrow{k} \mathsf{top}^{k-1}(u^k)$ where u^k is the k-stack stored in the topmost 0-stack of $R(1)$.

□

Proof (Lemma E.6). We first show that every return decomposes as required. Let R be a k-return of length m (by definition $m \geq 1$). Set $S := R{\restriction}_{0,1}$. When S performs a pop^k operation, and $m = 1$, we immediately get case one. Otherwise, we proceed by distinction of the cases of Lemma E.13 for the position $x := \mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R{\restriction}_{1,|R|})$.

- Assume that x points to the second topmost $(k-1)$-stack of $R(1)$. Then $R{\restriction}_{1,m}$ is easily seen to be a k-return; we get case 2 or case 4.
- Assume that x points to the third topmost $(k-1)$-stack of $R(1)$. Then the operation in S was push^k (or $\mathsf{push}^1_{a,j}$ for $k = 1$).

Proof (Claim). There is some $1 < i < m$ such that $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R{\restriction}_{i,m})$ points to the second topmost $(k-1)$-stack of $R(i)$.

Under the assumption that this claim holds, choose the minimal such i. From the choice of i it follows immediately that $U := R{\restriction}_{i,m}$ is a k-return. We show that $T := R{\restriction}_{1,i}$ is also a k-return whence R decomposes as in case 5 of the lemma. Indeed, $\mathsf{hist}(\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), U), T)$ is the third topmost $(k-1)$-stack of $R(1)$. Since $\mathsf{top}^{k-1}(R(i))$ is the $(k-1)$-stack directly on top of $\mathsf{hist}(\mathsf{top}^{k-1}(R(i)), U)$, and because $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R{\restriction}_{j,m})$ is not the topmost $(k-1)$-stack of $R(j)$ for $0 \leq j < m$ (definition of k-return), we can apply Proposition D.8 (variant 1) and conclude that $\mathsf{hist}(\mathsf{top}^{k-1}(R(i)), T)$ is the second topmost $(k-1)$-stack of $R(1)$. By the same proposition, if $\mathsf{hist}(\mathsf{top}^{k-1}(R(i)), T')$ is the topmost $(k-1)$-stack of $T'(0)$ for some proper suffix T' of T, then $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), T' \circ U)$ is the second topmost $(k-1)$-stack of $T'(0)$ which contradicts the minimality of i. Thus, T is a k-return and we showed that R decomposes as described in case 5 of the lemma.

Finally we prove our claim. If the operation leading to $R(m)$ is pop^k, $i = m-1$ is a good candidate. Otherwise (Proposition E.2), this operation is col^k. Then $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R{\restriction}_{m-1,m})$ is not simple. Let $i < m-1$ be the last index for which $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R{\restriction}_{i,m})$ is simple again (such i exists because $i = 0$ is a good candidate). From Corollary E.9, applied to $R{\restriction}_{i,m}$, we immediately obtain that $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R{\restriction}_{i,m})$ is the second topmost $(k-1)$-stack of $R(i)$, so i is a good candidate.

Assume that S performs a push^j operation with $j > k$, and x points to the second topmost $(k-1)$-stack of the second topmost $(j-1)$-stack of $R(1)$. Let $1 < i \leq m$ be minimal such that for $T := R{\restriction}_{1,i}$ and $U := R{\restriction}_{i,m}$ we have $\mathsf{hist}(\mathsf{top}^{j-1}(R(m)), U) = \mathsf{top}^{j-1}(R(i))$. We show that T is a j-return and U is a k-return which gives us case 5.

Due to the minimality of i all proper suffixes T' of T satisfy the inequality $\mathsf{hist}(\mathsf{top}^{j-1}(T(|T|)), T') \neq \mathsf{top}^{j-1}(T'(0))$. Due to Corollary D.10, the position $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), T \circ U)$ points into

$$\mathsf{hist}(\mathsf{top}^{j-1}(R(i)), T) = \mathsf{hist}(\mathsf{top}^{j-1}(R(m)), T \circ U).$$

Thus, $\mathsf{hist}(\mathsf{top}^{j-1}(R(i)), T)$ points to the second topmost $(j-1)$-stack of $T(0) = R(1)$ and we conclude that T is a j-return.

Due to Corollary D.10, we know that $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R{\restriction}_{i,m})$ points into $\mathsf{hist}(\mathsf{top}^{j-1}(R(m)), R{\restriction}_{i,m}) = \mathsf{top}^{j-1}(R(i))$. On the other hand, by Corollary E.4, the only position x in the topmost $(j-1)$-stack of $R(i)$ for which $\mathsf{hist}(x, R{\restriction}_{0,i})$ points to the topmost $(k-1)$-stack of $R(0)$ is x pointing to the second topmost $(k-1)$-stack of $R(i)$. By Proposition D.5 we conclude that $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R{\restriction}_{i,m})$ points to the second topmost $(k-1)$-stack, hence U is a k-return.

Finally, assume that S performs $\mathsf{push}^1_{a,k}$, and x is of the form $\mathsf{top}^0(R(1)) \xrightarrow{k} \mathsf{top}^{k-1}(u^k)$ where u^k is the k-stack stored in the topmost 0-stack of $R(1)$. Let $i > 1$ be minimal such that $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R{\restriction}_{i,m})$ is simple. Recall that $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R)$ points to the second topmost $(k-1)$-stack of $R(0)$.

From Corollary E.9, applied to $R{\restriction}_{0,i}$, we see that $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R{\restriction}_{i,m}) = \mathsf{top}^{k-1}(R(i))$. Since R is a return, this implies $i = m$ and due to the minimality of i, we conclude directly that $R{\restriction}_{1,i}$ is a k-colreturn.

This concludes the proof that every k-return decomposes as required by the lemma.

It is left to show that every run that decomposes as described by the lemma is a k-return. Let R be some run of length m. There are the following cases.

1. If $|R| = 1$ and it performs pop^k, the definition of hist implies that R is a k-return.
2. Assume that R starts with an operation of level at most $k-1$, and continues with a k-return. Since history preserves positions of $(k-1)$-stacks under operations of level at most $k-1$, and such operations also preserve the existence of all $(k-1)$-stacks, the conditions for R being a return are trivially deduced from the fact that $R{\restriction}_{1,m}$ is a return.
3. Assume that R starts with $\mathsf{push}^1_{a,k}$ and continues with a k-colreturn. By definition of a colreturn, the position $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R{\restriction}_{i,m})$ is not simple for all $1 \leq i < m$, whence this position is not $\mathsf{top}^{k-1}(R(i))$. Furthermore, $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R{\restriction}_{1,m})$ points into $\mathsf{top}^0(R(1))$ and has nesting rank 1, so $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R)$ is simple. By Corollary E.9 it follows that $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R)$ is the second topmost $(k-1)$-stack of $R(0)$.
4. Assume that R starts with a push^j for $j > k$ and continues with a k-return. Then we conclude similar to the second case.
5. Assume that R starts with a push^j for $j \geq k$ (including $\mathsf{push}^1_{a,l}$ for $j = 1$) and decomposes as $R = S \circ T \circ U$ where S has length 1, T is a j-return and U is a k-return. We know that $x := \mathsf{hist}(\mathsf{top}^{k-1}(R(m)), U)$ is the second topmost $(k-1)$-stack of $U(0)$. Corollary E.4 applied for run $S \circ T$ implies that $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R), R)$ $(= \mathsf{hist}(x, S \circ T))$ is the second topmost $(k-1)$-stack of $R(0)$.
   For $j = k$, we know that $\mathsf{hist}(\mathsf{top}^{k-1}(U(0)), T') \neq \mathsf{top}^{k-1}(T'(0))$ for every suffix T' of T of positive length. We apply Proposition D.8 (variant 2) to $\mathsf{top}^{k-1}(U(0))$ and x (the second topmost $(k-1)$-stack of $U(0)$ and obtain that $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), T' \circ U) \neq \mathsf{top}^{k-1}(T'(0))$ for every suffix T' of T. From this we conclude directly that R is a k-return.
   For $j > k$, we also see that $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), T' \circ U) \neq \mathsf{top}^{k-1}(T'(0))$ for every suffix T' of T of positive length. Indeed, if $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), T' \circ U) = \mathsf{top}^{k-1}(T'(0))$, then also $\mathsf{hist}(\mathsf{top}^{j-1}(R(m)), T' \circ U) = \mathsf{top}^{j-1}(T'(0))$ (Corollary D.10) which is impossible because T is a j-return. Again, it is easy to conclude that R is a k-return.

□

Proof (Lemma E.7). We first show that every k-colreturn R decomposes as described by the lemma. Set $S := R{\restriction}_{0,1}$ (the definition of a k-colreturn requires $|R| \geq 1$). If R performs col^k and $|R| = 1$ we are in case 1 of Lemma E.7. Otherwise, due to Proposition E.12, the operation in S is push. As in the return case, we look at $x := \mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R{\restriction_{1,|R|}})$. By Corollary D.5 we know that necessarily $\mathsf{hist}(x, S)$ is of the form $\mathsf{top}^0(R(0)) \xrightarrow{k} x'$, where x' is simple (because $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R)$ is of such form). There are the following possibilities.

1. S performs a push^j for $j \geq 2$ and $x = \mathsf{top}^0(R(1)) \xrightarrow{k} x_1$ for some simple x_1. In this case, it is straightforward to see that $R{\restriction_{1,m}}$ is a k-colreturn.
2. Otherwise, S performs a push operation of level j and $x = \mathsf{top}^0(R(0)) \xrightarrow{k} x_1$ for some simple x_1. Notice that $\mathsf{top}^0(R(0))$ is the topmost 0-stack of the second topmost $(j-1)$-stack of $R(1)$. Recall that the last operation of R is col^k whence $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R{\restriction_{m-1,m}})$ points into the topmost 0-stack of $R(m-1)$ and has nesting rank 1. Let $1 < i < m$ be minimal such that $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R{\restriction_{i,m}})$ has nesting rank 1 and points into the topmost $(j-1)$-stack of $R(i)$. Let $T := R{\restriction_{1,i}}$, and let $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R{\restriction_{i,m}}) = y \xrightarrow{k'} y'$. Due to Proposition E.8 (applied for run T), we have $k' = k$, and $\mathsf{hist}(y, T) = \mathsf{top}^0(R(0))$. Due to Corollary D.10 and since y points into $\mathsf{top}^{j-1}(R(i))$, $\mathsf{hist}(\mathsf{top}^{j-1}(R(i)), T)$ contains $\mathsf{top}^0(R(0))$ whence it points to the second topmost $(j-1)$-stack of $R(1)$. The same corollary and the minimality of i implies that for each suffix T' of T of length at least 1 we have $\mathsf{hist}(\mathsf{top}^{j-1}(R(i)), T') \neq \mathsf{top}^{j-1}(T'(0))$. Thus, T is a j-return.
   Let $U := R{\restriction_{i,m}}$ We know that $\mathsf{hist}(y, S \circ T) = \mathsf{top}^0(R(0))$, and that y is in the topmost $(j-1)$-stack of $R(i)$. By Corollary E.4, the only y satisfying this is $y = \mathsf{top}^0(R(i))$. It follows that U is a k-colreturn.

It is left to show that every run R that decomposes as described by the lemma is a k-colreturn. In the first two cases we immediately see that R is a k-colreturn. So assume that $R = S \circ T \circ U$ where S has length 1 and performs a push^j (including $\mathsf{push}^1_{a,l}$ for $j = 1$), T is a j-return and U is a k-colreturn. Let $m := |R|$. As $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), U)$ is of the form $\mathsf{top}^0(U(0)) \xrightarrow{k} x$ for simple x, by Corollary E.4 we immediately obtain that $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R) = \mathsf{top}^0(R(0)) \xrightarrow{k} x$.

In order to prove that R is a k-colreturn, we still have to show that the position $\mathsf{hist}(\mathsf{top}^0(U(0)) \xrightarrow{k} x, T{\restriction_{i,|T|}})$ is not simple for all $0 \leq i \leq |T|$. Heading for a contradiction, assume that there is a greatest index i for which the position $\mathsf{hist}(\mathsf{top}^0(U(0)) \xrightarrow{k} x, T{\restriction_{i,|T|}})$ is simple. Trivially $i < |T|$. If $i = |T| - 1$, the last operation of T has to be $\mathsf{push}^1_{a,k}$, which is impossible in a j-return. So $i \leq |T| - 2$. As $\mathsf{hist}(\mathsf{top}^0(U(0)) \xrightarrow{k} x, T{\restriction_{i+1,|T|}})$ is not simple (maximality of i), it has to be of the form $\mathsf{top}^0(T(i+1)) \xrightarrow{k'} x'$. Proposition E.8 applied to $T{\restriction_{i+1,|T|}}$ implies that $k' = k$ and $\mathsf{hist}(\mathsf{top}^0(U(0)), T{\restriction_{i+1,|T|}}) = \mathsf{top}^0(T(i+1))$. Due to Corollary D.10, also $\mathsf{hist}(\mathsf{top}^{j-1}(U(0)), T{\restriction_{i+1,|T|}}) = \mathsf{top}^{j-1}(T(i+1))$. This is impossible because T is a j-return. $\square$

Up to now, we have only dealt with the general shape of k-returns and k-colreturns. In Section 5 we divided these sets further according to their *change* *level*. We next formally introduce this change level for every k-return and k-colreturn and then complete the proof that the rules given in Section 5 correctly describe returns and colreturns. The *change level* keeps track of the maximal level on which the stack size was changed by the run R.

Definition E.14. Let R be a k-return or a k-colreturn. For $1 \leq i \leq n$, let x^i be the size of the topmost i-stack of $R(0)$; similarly y^i for $R(|R|)$. Then $\mathsf{chl}(R) := \max\{i : x^i \neq y^i\}$.

Remark E.15. If R is a k-return (k-colreturn), then Proposition E.3 (Lemma E.10, respectively) implies that the size of topmost k-stack of $R(0)$ and of $R(|R|)$ is different, so $\mathsf{chl}(R) \geq k$.

We now give a characterisation of the change level of a k-return or k-colreturn depending on the change level(s) of the subruns occurring in its decomposition.

Lemma E.16. *Let R be a k-return or a k-colreturn.*

1. *If $|R| = 1$, then $\mathsf{chl}(R) = k$,*
2. *If R decomposes as $R = S \circ T$, where S of length 1 performs an operation of level j, and T is a k-return or a k-colreturn, then $\mathsf{chl}(R) = \max\{j, \mathsf{chl}(T)\}$).*
3. *If R decomposes as $R = S \circ T \circ U$ where S of length 1 performs a push^j operation (including $\mathsf{push}^1_{a,l}$ for $j = k = 1$), T is a j-return, and U is a k-return or a k-colreturn, then*

$$\mathsf{chl}(R) = \begin{cases} \mathsf{chl}(U) & \text{if } \mathsf{chl}(T) = j \\ \max\{\mathsf{chl}(T), \mathsf{chl}(U)\} & \text{otherwise.} \end{cases}$$

We begin the proof with an auxiliary proposition saying that k-returns and k-colreturns cannot decrease the size of the stacks of level greater than k.

Proposition E.17. *Let R be a k-return or k-colreturn such that $\mathsf{chl}(R) > k$. Then the size of the topmost $\mathsf{chl}(R)$-stack of $R(0)$ is smaller than the size of the topmost $\mathsf{chl}(R)$-stack of $R(|R|)$.*

Proof. Let $m := |R|$. For a k-return, $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R)$ points into in the topmost k-stack of $R(0)$, so by Corollary D.10, $\mathsf{hist}(\mathsf{top}^k(R(m)), R) = \mathsf{top}^k(R(0))$. For a k-colreturn we also have $\mathsf{hist}(\mathsf{top}^k(R(m)), R) = \mathsf{top}^k(R(0))$, due to Lemma E.10. In both cases, by Lemma D.11, $\mathsf{top}^k(R(0)) \preceq \mathsf{top}^k(R(m))$. It follows that the size of the topmost $\mathsf{chl}(R)$-stack of $R(0)$ is smaller than the size of the topmost $\mathsf{chl}(R)$-stack of $R(m)$ (as for $i > \mathsf{chl}(R)$, the size of the topmost i-stack of $R(0)$ and of $R(m)$ is the same, and for $i = \mathsf{chl}(R) > k$ they differ). $\square$

Next we proof Lemma E.16.

Proof (Lemma E.16).

1. Case 1 is immediate.

2. Assume we have case 2 of the lemma. Notice that neither S nor T can change the size of the i-stack for $i > \max\{j, \mathsf{chl}(T)\}$). If $j \neq \mathsf{chl}(T)$, we see that one of the subruns changes the size of the stack of level $\max\{j, \mathsf{chl}(T)\}$), and the other does not change it, so we get $\mathsf{chl}(R) = \max\{j, \mathsf{chl}(T)\}$). If $j = \mathsf{chl}(T)$, $\mathsf{chl}(T) \geq k$ (Remark E.15) implies that the operation is necessarily push (cf. Propositions E.11 and E.12). Then the size of the stack of level j is increased by S and by T (cf. Proposition E.17). Thus, the claim follows immediately.
3. Next, assume we have case 3 of the lemma. None of the parts S, T, U changes the size of the i-stack for $i > \max\{\mathsf{chl}(T), \mathsf{chl}(U)\}$). If $\mathsf{chl}(T) = j$, Corollary E.4 implies that the topmost j-stack of $R(0)$ and of $U(0)$ is the same, thus $\mathsf{chl}(R) = \mathsf{chl}(U)$. So assume that $\mathsf{chl}(T) > j$. Then the size of the stack of level $\max\{\mathsf{chl}(T), \mathsf{chl}(U)\}$) cannot be decreased by S or T or U (Proposition E.17), and at least one of T and U increases this value. Thus, $\mathsf{chl}(R) = \max\{\mathsf{chl}(T), \mathsf{chl}(U)\}$).

□

E.2 Non-Erasing Runs

Definition E.18. For $0 \leq k \leq l$, let $\mathcal{N}_{k,\varepsilon}$ be the set of top^k-non-erasing runs which is the set of runs R such that position $\mathsf{top}^k(R(0))$ is present in every configuration of R.

Using k-returns we can characterise top^k-non-erasing runs in the following way.

Lemma E.19. *Let R be some run and $0 \leq k \leq n$. R is a top^k-non-erasing run if and only if R has one of the following forms.*

1. $|R| = 0$.
2. *R starts with an operation of level at most k, and continues with a top^k-non-erasing run.*
3. *R starts with a push^j (including arbitrary $\mathsf{push}^1_{a,l}$ for $j = 1$) for $j \geq k+1$, and continues with a top^{j-1}-non-erasing run.*
4. *R starts with a push^j (including arbitrary $\mathsf{push}^1_{a,l}$ for $j = 1$) and decomposes as $R = S \circ T \circ U$, where S has length 1, T is a j-return of change level j, and U is a top^k-non-erasing run.*

We start the proof by giving two propositions useful in the right-to-left implication.

Proposition E.20. *Let $R = S \circ T$ be a run such that S and T are top^k-non-erasing runs for some k. Then R is a top^k-non-erasing run.*

Proof. We claim the following. Take some run such that x and y are simple positions in its initial stack such that $x \preceq y$. If y is present in all configurations of the run, then x is also present in all configurations of the run.

Since $\mathsf{top}^k(R(0))$ is present in all configurations of S, the claim implies that all k-stacks present in $R(0) = S(0)$ are also present in $S(|S|)$. Thus, the topmost k-stack of $T(0) = S(|S|)$ is lexicographically greater or equal than $\mathsf{top}^k(R(0))$. Again using the claim, $\mathsf{top}^k(R(0))$ is present in all configurations of T because $\mathsf{top}^k(T(0))$ is present in all configurations of T.

For the proof of the claim note that the statement of the claim is preserved under composition of runs. Thus, we may consider a run R of length 1 such that $x \preceq y$ are positions in $R(0)$. Since push operations do not delete positions in a stack, we may assume that R performs pop^j or col^j. Since an application of col^j has the same effect as several pop^j, it is sufficient to consider the pop^j case (the col^j-case then follows again by the composition closure argument). Assume that R performs a pop^j and x is present in $R(0)$ but not in $R(1)$. Then x points into or to the topmost $(j-1)$-stack of $R(0)$. Since $x \preceq y$, y must also point into or to the topmost $(j-1)$-stack of $R(0)$. But then y is not present in $R(1)$. □

Proposition E.21. *Let $0 \leq k \leq n$, and let R be a run such that $\mathsf{hist}(y, R) = \mathsf{top}^k(R(0))$ for some position y of $R(|R|)$. Then R is a top^k-non-erasing run.*

Proof. Heading for a contradiction, assume that there is a minimal $i \leq |R|$ such that $x_0 := \mathsf{top}^k(R(0))$ is not present in $R(i)$. All simple positions in $R(i)$ are lexicographically smaller than x_0, because x_0 was removed either by a pop operation, or by a col operation (cf. Remark 2.4). Let x_1 be the simple prefix of $\mathsf{hist}(y, R{\restriction}_{i,|R|})$. Due to Lemma D.11 applied to $R{\restriction}_{0,i}$, $x_0 \preceq x_1$. But this is a contradiction. □

Proof (Lemma E.19). The proof of the right-to-left part is by case distinction on the decomposition of R according to the four cases. Case 1 is trivial and Cases 2 and 3 follow directly from Proposition E.20. We now investigate Case 4. Notice that $\mathsf{hist}(\mathsf{top}^k(T(|T|)), S \circ T) = \mathsf{top}^k(S(0))$: for $k < j$ it follows from Proposition E.3; for $k \geq j$ it follows from Corollary D.10. Thus, Proposition E.21 applied to $S \circ T$ and $y := \mathsf{top}^k(T(|T|))$ tells us that $S \circ T$ is a top^k-non-erasing run. Due to Proposition E.20, also R is a top^k-non-erasing run.

Now concentrate on the left-to-right part. Let R be a run of length m such that $x := \mathsf{top}^k(R(0))$ is present in all configurations of R. If $m = 0$, we are in case 1. Thus, assume that $m \geq 1$. Note that the first operation cannot be col^j or pop^j for $j \geq k+1$ because this would delete position x from $R(1)$ (cf. Remark 2.4). .Hence, one of the following cases applies.

- Assume that the first operation in R is of level at most k. Then $x = \mathsf{top}^k(R(1))$ and x is not removed during $R{\restriction}_{1,m}$. Thus, R decomposes as in case 2.
- Assume that the first operation in R is push^j for some $j \geq k+1$ (in the rest of the proof, push^1 stands for arbitrary $\mathsf{push}^1_{a,k'}$). Furthermore, assume that $y := \mathsf{top}^{j-1}(R(1))$ is present in all configurations of $T := R{\restriction}_{1,m}$. Then R decomposes as as in case 3.
- Otherwise, the first operation is push^j for some $j \geq k+1$ and there is a minimal $l \geq 1$ such that $y := \mathsf{top}^{j-1}(R(1))$ is not present in $R(l)$. We claim that $R{\restriction}_{1,l}$ is a j-return of change level j and that the positions $\mathsf{top}^k(R(l))$

and $\mathsf{top}^k(R(0))$ agree. Hence, $R{\upharpoonright}_{l,m}$ is top^k-non-erasing and R decomposes as in case 4.
Let us proof the claim. Recall that $x = \mathsf{top}^k(R(0))$ is present in all configurations of $R{\upharpoonright}_{0,l}$. Since x points into $x' := \mathsf{top}^{j-1}(R(0))$ or $x' = x$,

$$x' \text{ is present in all configurations of } R{\upharpoonright}_{0,l}. \tag{2}$$

Hence we can apply Lemma D.12 and conclude that $\mathsf{hist}(x', R{\upharpoonright}_{0,l}) = x'$ and

$$\text{if } \mathsf{hist}(x', R{\upharpoonright}_{l',l}) \text{ is simple, it is equal to } x'. \tag{3}$$

Thus, $\mathsf{hist}(x', R{\upharpoonright}_{1,l}) = x'$ because no non-simple position z in $R(1)$ satisfies $x' = \mathsf{hist}(z, R{\upharpoonright}_{0,1})$.
By definition of push^j, x' is the second topmost $(j-1)$-stack in $R(1)$. Since y is directly above x' and present in $R(l')$ for all $1 \leq l' < l$, (2) and (3) imply that $\mathsf{hist}(x', R{\upharpoonright}_{l',l}) \neq \mathsf{top}^{j-1}(R(l'))$. Finally, note that from $R(l-1)$ to $R(l)$ the $(j-1)$-stack above x' (which is at y) is removed but x' is present in $R(l)$. Using Remark 2.4, the operation is pop^j or col^j and y points into the topmost j-stack of $R(l-1)$ whence x' points to the topmost $(j-1)$-stack of $R(l)$.
In summary, $x' = \mathsf{top}^{j-1}(R(l))$, $\mathsf{hist}(x', R{\upharpoonright}_{1,l}) = x'$ is the second topmost $(j-1)$-stack of $R(1)$ and $\mathsf{hist}(x', R{\upharpoonright}_{l',l})$ is not the topmost $(j-1)$-stack of $R(l')$ for all $1 \leq l' < l$. Thus, $R{\upharpoonright}_{1,l}$ is a j-return of change level j and the claim is proved.

$\square$

E.3 Pumping runs

In this subsection we give a definition of pumping runs and prove that the rules from Section 5 describe pumping runs correctly.

Definition E.22. For $x \in \{=, <\}$ and $y \in \{\varepsilon, \not\varepsilon\}$, let $\mathcal{P}_{x,y}$ be the set of runs R such that

- $\mathsf{hist}(\mathsf{top}^0(R(|R|)), R) = \mathsf{top}^0(R(0))$, and
- $\mathsf{top}^0(R(|R|)) = \mathsf{top}^0(R(0))$ if and only if x is $=$, and
- R uses only ε-transitions if and only if $y = \varepsilon$.

A run R is a *pumping run* if it belongs to some $\mathcal{P}_{x,y}$.

Remark E.23. Lemma D.11 implies that for a pumping run $R \in \mathcal{P}_{<,y}$, we have $\mathsf{top}^0(R(0)) \prec \mathsf{top}^0(R(|R|))$. In this sense the final stack of a pumping run is is greater than its initial one.

For the next proofs it is useful to distinguish all k-returns of minimal change level (i.e., of change level k) from those of higher change level.

Definition E.24. We set $\mathcal{R}_{k,=,x} := \mathcal{R}_{k,k,x}$ and $\mathcal{R}_{k,<,x} := \bigcup_{i>k} \mathcal{R}_{k,i,x}$.

Remark E.25. A k-return R in $\mathcal{R}_{k,=,x}$ satisfies $\mathsf{top}^k(R(0)) = \mathsf{top}^k(R(|R|))$. Due to Proposition E.17, a k-return R in $\mathcal{R}_{k,<,x}$ satisfies $\mathsf{top}^k(R(0)) \prec \mathsf{top}^k(R(|R|))$.

In the rest of this subsection we characterise pumping runs using wf-rules.

Lemma E.26. *Let R be some run. R is a pumping run if and only if R has one of the following forms.*

1. $|R| = 0$.
2. *R starts with a push^k of any level (including arbitrary $\mathsf{push}^1_{a,l}$ for $k = 1$), and continues with a pumping run.*
3. *R starts with a push^k of any level (including arbitrary $\mathsf{push}^1_{a,l}$ for $k = 1$), and decomposes as $R = S \circ T \circ U$, where S has length 1, T is a k-return, and U is a pumping run.*

Additionally, assuming that R is a pumping run, $\mathsf{top}^0(R(0)) = \mathsf{top}^0(R(|R|))$ if and only if

- *R is of the first form, or*
- *R is of the last form, and $\mathsf{top}^k(T(0)) = \mathsf{top}^k(T(|T|))$, and $\mathsf{top}^0(U(0)) = \mathsf{top}^0(U(|U|))$*

The characterisation of pumping runs in terms of the well-formed rules presented in Section 5 follows immediately from the previous lemma.

Remark E.27. Observe that $\mathsf{hist}(x, R) = \mathsf{top}^0(R(0))$ implies that the first operation of R is not pop or col. Indeed, after such operation in $R(1)$ we have no position y such that $\mathsf{hist}(y, R\restriction_{0,1}) = \mathsf{top}^0(R(0))$ (which contradicts with Proposition D.5).

Proof (Lemma E.26). The right-to-left direction of the first part is almost immediate. In the third case we have to observe that $\mathsf{hist}(\mathsf{top}^0(U(0)), S \circ T) = \mathsf{top}^0(R(0))$; it follows from Corollary E.4.

Now concentrate on the left-to-right direction of the first part of the lemma. Let R be a pumping run of length m. If $m = 0$, we are in case 1. Thus, assume that $m \geq 1$. Due to the above remark, R starts with a push operation of some level k. Recall that $\mathsf{hist}(x, R\restriction_{0,1}) = \mathsf{top}^0(R(0))$ only if $x = \mathsf{top}^0(R(1))$ or if x points to the topmost 0-stack of the second topmost $(k-1)$-stack. By Proposition D.5, $\mathsf{hist}(\mathsf{top}^0(R(m)), R\restriction_{1,m})$ is one of these positions x. Now there are two cases.

- If $\mathsf{hist}(\mathsf{top}^0(R(m)), R\restriction_{1,m}) = \mathsf{top}^0(R(1))$, then $R\restriction_{1,m}$ is a pumping run and R decomposes as in case 2.
- Otherwise, $\mathsf{hist}(\mathsf{top}^0(R(m)), R\restriction_{1,m})$ points to the topmost 0-stack of the second topmost $(k-1)$-stack of $R(1)$. Due to Corollary D.10, we conclude that $\mathsf{hist}(\mathsf{top}^{k-1}(R(m), R\restriction_{1,m})$ points to the second topmost $(k-1)$-stack of $R(1)$. Let $2 \leq i \leq m$ be minimal such that $\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R\restriction_{i,m}) = \mathsf{top}^{k-1}(R(i))$. Notice that $T := R\restriction_{1,i}$ satisfies all requirements of a k-return.

By Corollary D.10 we know that $\mathsf{hist}(\mathsf{top}^0(R(m)), R{\restriction_{i,m}})$ points into

$$\mathsf{hist}(\mathsf{top}^{k-1}(R(m)), R{\restriction_{i,m}}) = \mathsf{top}^{k-1}(R(i)).$$

On the other hand, by Corollary E.4, the only position x in the topmost $(k-1)$-stack of $R(i)$ for which $\mathsf{hist}(x, R{\restriction_{0,i}}) = \mathsf{top}^0(R(0))$ is $x = \mathsf{top}^0(R(i))$. By Proposition D.5 we conclude that $\mathsf{hist}(\mathsf{top}^0(R(m)), R{\restriction_{i,m}}) = \mathsf{top}^0(R(i))$, hence $U := R{\restriction_{i,m}}$ is a pumping run.

Next we prove the last part of the lemma. If R is of length 0 we immediately get $\mathsf{top}^0(R(0)) = \mathsf{top}^0(R(|R|))$. Let R be a run satisfying item 3 such that $\mathsf{top}^k(T(0)) = \mathsf{top}^k(T(|T|))$ and $\mathsf{top}^0(U(0)) = \mathsf{top}^0(U(|U|))$. Because the operation in S is push^k we also have $\mathsf{top}^k(R(0)) = \mathsf{top}^k(T(|T|))$. By Corollary E.4 we know that the topmost k-stack of $R(0)$ and of $T(|T|)$ are the same, so $\mathsf{top}^0(R(0)) = \mathsf{top}^0(T(|T|)) = \mathsf{top}^0(R(|R|))$.

Finally assume that R is a pumping run of length $m \geq 1$ such that $\mathsf{top}^0(R(m)) = \mathsf{top}^0(R(0))$. Then $\mathsf{hist}(\mathsf{top}^0(R(m)), R) = \mathsf{top}^0(R(m))$. We already have observed that $\mathsf{hist}(\mathsf{top}^0(R(m)), R{\restriction_{1,m}})$ is simple. Due to Lemma D.11 it is lexicographically bounded from above by $\mathsf{top}^0(R(m))$ and from below by $\mathsf{hist}(\mathsf{top}^0(R(m)), R) = \mathsf{top}^0(R(m))$. We conclude that $\mathsf{hist}(\mathsf{top}^0(R(m)), R{\restriction_{1,m}}) = \mathsf{top}^0(R(0))$. From the analysis in the first part, we know that R then satisfies case 3, i.e., it decomposes as $R = S \circ T \circ U$ where S performs only one push^k, T is a k-return and U is a pumping run. Using the same argument again, we conclude that $\mathsf{top}^0(U(0)) = \mathsf{hist}(\mathsf{top}^0(R(m)), U) = \mathsf{top}^0(R(m))$. Using Corollary D.10 we also get that $\mathsf{top}^k(T(0)) = \mathsf{top}^k(R(0)) = \mathsf{top}^k(U(0))$. □

In conclusion, Lemmas E.26, E.19, E.16, E.7 and E.6 show that the Rules from Section 5 describe sets of runs that satisfy the intended meaning described in that Section.

F Sketch of proof of Theorem 1.1

In this section we describe briefly the proof of the pumping lemma The single steps of this proof follow closely the analogous proof for the non-collapsible pushdown systems in [15]. For the details of these steps we refer the reader to Appendix H (which requires Appendix G as combinatorial background).

First we list three propositions, which are consequences of Theorem 3.1 applied to the family $\mathcal{X}$ from Section 5.

Proposition F.1. *Let R be a pumping run of the system $\mathcal{S}$ such that R satisfies $\mathsf{ctype}_{\mathcal{X}}(R(0)) \sqsubseteq \mathsf{ctype}_{\mathcal{X}}(R(|R|))$. Then there is a sequence of runs $(R_i)_{i\in\mathbb{N}}$ such that $\mathsf{ctype}_{\mathcal{X}}(R(|R|)) \sqsubseteq \mathsf{ctype}_{\mathcal{X}}(R_i(|R_i|))$ and*

1. *if $R \in \mathcal{P}_{\not\varepsilon}$ then R_i contains at least i non-ε-transitions,*
2. *if $R \in \mathcal{P}_{<,\varepsilon}$ then the final stack of R_{i+1} is greater than the final stack of R_i, and R_i uses only ε-transitions.*

Proof. Set $R_0 := R$. Application of Theorem 3.1 to R and configuration $R_i(|R_i|)$ yields a pumping run R'_{i+1}. Set $R_{i+1} := R_i \circ R'_{i+1}$. □

Proposition F.2. *Let R be a top^0-non-erasing run and c some configuration such that $\mathsf{ctype}_{\mathcal{X}}(R(0)) \sqsubseteq \mathsf{ctype}_{\mathcal{X}}(c)$. Then there is a top^0-non-erasing run S that starts in c and ends in the same state as R.*

Proposition F.3. *Let R be a run and c a configuration such that $\mathsf{ctype}_{\mathcal{X}}(R(0)) \sqsubseteq \mathsf{ctype}_{\mathcal{X}}(c)$. Then there is a run S which starts in c and ends in the same state as R.*

Let us also comment on the crucial properties of pumping runs and top^0-non-erasing runs. A run $R \in \mathcal{P} \cup \mathcal{N}_0$ ends in a stack which is not smaller than the stack in which R starts; in particular $R \in \mathcal{P}_{<,\varepsilon}$ ends in a strictly greater stack than it starts. The classes $\mathcal{P}$ and $\mathcal{N}_0$ are quite similar. The main differences between them are the following two. The definition of pumping runs is more restrictive, i.e., $\mathcal{P} \subseteq \mathcal{N}_0$ allowing to set $\mathsf{lev}(\mathcal{P}_{\not\varepsilon}) = \mathsf{lev}(\mathcal{P}_{<,\varepsilon}) = 0$ while $\mathsf{lev}(\mathcal{N}_0) > 0$. Thus, Theorem 3.1 gives a stronger transfer property for $\mathcal{P}$ than for $\mathcal{N}_0$. On the other hand, $\mathcal{N}_0$ is closed under prefixes in the sense that for $R \in \mathcal{N}_0$ we also have $R{\restriction}_{0,i} \in \mathcal{N}_0$ for any $i \leq |R|$. For $R \in \mathcal{P}$ we have $R{\restriction}_{0,i} \in \mathcal{N}_0$ for all $i < |R|$ but not always $R{\restriction}_{0,i} \in \mathcal{P}$.

Next we show how these propositions can be used to prove the pumping lemma. This part consists of the following steps (where we write $\mathcal{G}$ for the ε-contraction of the graph of $\mathcal{S}$ of level n).

1. A simple construction shows that we may assume that the state of a reachable configuration c determines whether $c \in \mathcal{G}$ (the state set Q is partitioned as $Q = Q_\varepsilon \cup Q_{\not\varepsilon}$ such that $q \in Q_\varepsilon$ implies that all edges leading to q are labelled ε and $q \in Q_{\not\varepsilon}$ implies that all edges leading to q are not labelled ε). In the rest of the proof we assume that this condition holds.
2. We say a run R *induces a path of length l in $\mathcal{G}$* if $R(0) \in \mathcal{G}$ and $R(|R|) \in \mathcal{G}$ and there are $l+1$ many $i \leq |R|$ such that $R(i) \in \mathcal{G}$. Recall that S_n was defined in Theorem 1.1. We show that every run R starting at a configuration c of distance m from the initial one in $\mathcal{G}$ that induces a path of length S_n in $\mathcal{G}$ contains a pumping subrun $T \in \mathcal{P}_{<,\varepsilon} \cup \mathcal{P}_{\not\varepsilon}$ such that $R = S \circ T \circ U$. Moreover, if $T \in \mathcal{P}_{<,\varepsilon}$ we can show that $U{\restriction}_{0,m}$ is a top^0-non-erasing run for some $m \leq |U|$ such that $U(m) \in \mathcal{G}$.
3. We conclude with the following case distinction.
   - If $T \in \mathcal{P}_{<,\varepsilon}$, let $i \leq |S|$ be maximal such that $S(i)$ is a node of $\mathcal{G}$. We apply Proposition F.1 to T and obtain infinitely many runs $(R_j)_{j\in\mathbb{N}}$ ending in configurations (q, s_i) such that $S{\restriction}_{i,|S|} \circ R_j$ is an ε-labelled path from $S(i)$ to (q, s_i) where s_{i+1} is greater than s_i for all $i \in \mathbb{N}$. We can apply Proposition F.2 to $U{\restriction}_{0,m}$ and (q, s_i) and obtain a top^0-non-erasing run U'_i from (q, s_i). By construction U'_i ends in the same state as $U{\restriction}_{0,m}$. Due to step 1, we conclude that U'_i ends in some node of $\mathcal{G}$ (because $U{\restriction}_{0,m}$ does so and their final states coincide). Let U_i be the minimal prefix of U'_i such that $U_i(|U_i|) \in \mathcal{G}$, i.e., $U_i = U'_i{\restriction}_{0,j}$ such that the transition between $U'_i(j-1)$ and $U'_i(j)$ is the first transition of U'_i not labelled by ε. We know that U_i is also a top^0-non-erasing run, so it ends in a greater or the same stack than it starts. Let g_i be the final

configuration of U_i. Note that $S\restriction_{i,|S|} \circ R_i \circ U_i$ is a run that induces a path of length 1 from $S(i)$ to g_i, i.e., g_i is a successor of $S(i)$ in $\mathcal{G}$. Since the sizes of the stacks s_i increase strictly for each i there is a j such that s_j is bigger than g_i. Since g_j is even bigger than s_j, g_i and g_j cannot coincide. Inductive use of this argument yields a sequence $(i_k)_{k\in\mathbb{N}}$ such that the g_{i_k} are pairwise different successors of $S(i)$ in $\mathcal{G}$. Especially, we conclude that this case cannot occur in a finitely branching ε-contraction.

– Otherwise, $T \in \mathcal{P}_{\not\in}$. Application of Proposition F.1 to T yields a sequence $(R_i)_{i\in\mathbb{N}}$ of runs in $\mathcal{P}_{\not\in}$ starting in $T(0)$ such that R_i contains at least i transitions not labelled ε and such that $\mathsf{ctype}_{\mathcal{X}}(T(|T|)) \sqsubseteq \mathsf{ctype}_{\mathcal{X}}(R_i(|R_i|))$. Application of Proposition F.3 to U and $R_i(|R_i|)$ yields a run U_i from $R_i(|R_i|)$ to some configuration with the same final state as that of U (which is also the same final state as that of R). Using part 1), we conclude that $S \circ R_i \circ U_i$ induces a path of length at least i starting in c for each $i \in \mathbb{N}$. Thus, $\mathcal{G}$ contains paths of arbitrary length starting in c and ending in the same state as R; this completes the proof of the pumping lemma.

G Combinatorics for Theorem 1.1

In this part we collect some combinatorial facts that turn out to be useful in the proof of Theorem 1.1. The first lemma says that if we have a sequence of natural numbers which increase at most by one from one number to the next and if we choose a set G of $2^k - 1$ of these numbers then we find an increasing subsequence of k numbers such that each element of the subsequence is strictly smaller than all following elements of the sequence up to the next occurrence of a number from G. This sequence is intended to contain sizes of a stack during a run; such size can increase by at most 1 (when a push is performed) and can decrease arbitrarily (when a col is performed). For $G \subseteq \mathbb{N}$ with $l-1 \in G$ and $i < l$ we set $n_G(i) := \min\{g \in G : g \geq i\}$.

Lemma G.1. *Let $k \in \mathbb{N} \setminus \{0\}$, let $(a_i)_{0\leq i\leq l}$ be a sequence of natural numbers such that $a_i - a_{i-1} \leq 1$ for all $1 \leq i \leq l$ and such that $a_0 = \min\{a_i : 0 \leq i \leq l\}$. Let $G \subseteq \{0, 1, \dots, l-1\}$ be such that $|G| \geq 2^k - 1$.*

There is an $e \leq l$ such that $e - 1 \in G$ and for

$$H_e := \{i \leq e-1 : a_i \leq a_j \text{ for all } i \leq j \leq e \text{ and } a_i < a_j \text{ for all } i < j \leq n_G(i)\}$$

we have $|H_e| \geq k$.

Proof. The proof is by induction on l. For $0 \leq b \leq e \leq l$ we write $H_{b,e} = H_e \cap \{i \in \mathbb{N} : i \geq b\}$. Note that it suffices to find $0 \leq b \leq e \leq l$ such that $|H_{b,e}| \geq k$. We distinguish the following cases.

1. Assume that $k = 1$. Since $|G| \geq 2^1 - 1 = 1$, we can choose some $e' \in G$. Let $e := e' + 1 \leq l$. Choose $b \leq e'$ maximal such that $a_b = a_0$. By choice, $a_b < a_j$ for all $b < j \leq e'$, and $a_b = a_0 \leq a_e$. Thus, $b \in H_{b,e}$ which settles the claim.

2. Assume that there is some $0 < b \leq l$ such that $G \subseteq \{b, b+1, \ldots, l-1\}$ and $a_b = \min\{a_i : b \leq i \leq e\}$. By induction hypothesis, there is some $e \leq l$ with $e - 1 \in G$ such that $|H_{b,e}| \geq k$.
3. Assume that there is some $1 \leq l' \leq l$ such that $a_i > a_0$ for all $1 \leq i \leq l'$ and $|G \cap \{1, 2, \ldots, l'-1\}| \geq 2^{k-1} - 1$. Since $a_1 - a_0 \leq 1$, it follows that $a_1 = a_0 + 1 = \min\{a_i : 1 \leq i \leq l'\}$. Thus, we can apply the induction hypothesis to the sequence $a_1, a_2, \ldots, a_{l'}$ and $k-1$ and obtain $e \leq l'$ such that $e - 1 \in G$ and $|H_{1,e}| \geq k - 1$. Since $a_0 < a_i$ for all $1 \leq i \leq l'$, we conclude that $H_{0,e} = \{a_0\} \cup H_{1,e}$ contains at least k elements.
4. Assume that none of the above cases holds. Then in particular $k \geq 2$. Let $b \geq 1$ be the smallest index such that $a_b = a_0$. If such b would not exist, case 3 would hold with $l' = l$.
   Let $G' = G \cap \{b, b+1, \ldots, l-1\}$. We have $a_i > a_0$ for $1 \leq i \leq b-1$. Because $b - 1$ cannot be taken as l' in case 3, we either have $b = 1$, or $|G \cap \{1, 2, \ldots, b-2\}| \leq 2^{k-1} - 2$. In the former case, $|G'| \geq 2^k - 1 - 1 \geq 2^{k-1} - 1$ and in the latter case $|G'| \geq (2^k - 1) - 2 - (2^{k-1} - 2) \geq 2^{k-1} - 1$. Since $a_b = a_0 = \min\{a_i : 0 \leq i \leq l\}$, the induction hypothesis applies to the shorter sequence $a_b, a_{b+1}, \ldots, a_l$ and G'. Thus, there is some $e \leq l$ such that $e - 1 \in G' \subseteq G$ and $|H_{b,e}| \geq k - 1$.
   Since we are not in case 2, $G' \neq G$ whence there is some $g \in G$ with $0 \leq g \leq b - 1$. Since $a_0 < a_i$ for all $0 < i \leq b - 1$ we also have $a_0 < a_i$ for all $0 < i \leq g$. Since a_0 is the minimal element of the sequence and since $0 < b$ we have $|H_{0,e}| \geq |\{a_0\} \cup H_{b,e}| \geq k$.

□

Corollary G.2. *Let $k \in \mathbb{N} \setminus \{0\}$ and let $a_0, a_1, \ldots, a_l$ be a sequence of positive natural numbers such that $a_i - a_{i-1} \leq 1$ for $1 \leq i \leq l$. Let $G \subseteq \{0, 1, \ldots, l-1\}$ be such that $|G| \geq a_0 \cdot 2^k$. Then there exist two indices $0 \leq b < e \leq l$ such that*

1. *$e - 1 \in G$,*
2. *$a_b = \min\{a_i : b \leq i \leq e\}$,*
3. *$a_i > a_b$ for each $0 \leq i < b$ and*
4. *$|H_{b,e}| \geq k$.*

Proof. For each $0 \leq j \leq l$ set

$$m_j := \min\{a_i : 0 \leq i \leq j\}.$$

Notice that $m_0, m_1, \ldots, m_l$ is a decreasing sequence of numbers between 1 and a_0. Thus,

$$M_i := \{j : 0 \leq j \leq l : m_j = i\}$$

is a (possibly empty) interval for each $1 \leq i \leq a_0$ and the M_i form a partition of $\{j : 0 \leq j \leq l\}$. Thus, there is at least one $1 \leq i \leq a_0$ such that $G_i := G \cap M_i$ has at least 2^k many elements. Set b and c to be the minimal and maximal element, respectively, of M_i. By definition $a_b = i = \min\{a_j : j \in M_i\}$, and $a_b < a_j$ for all $0 \leq j < b$. We can now apply Lemma G.1 to $(a_j)_{j \in M_i}$ and $G_i \setminus \{c\}$. This shows the existence of some $e \leq l$ such that $e - 1 \in G_i \subseteq G$ and $|H_{b,e}| \geq k$. □

We fix constants $c \geq 2$ and m and define several sequences which are parameterised by c and m. In the next section, we will always use $c = |\mathcal{T}_\mathcal{S}| + 1$ and m will be the length of a fixed path in the graph of $\mathcal{S}$. In the final part of this section, we prove certain properties of these sequences that we will use in the next section.

Definition G.3. 1. Set $M_1 := (m+1) \cdot c$ and $M_j := 2^{M_{j-1}}$ for $j \geq 2$,

2. set $M'_1 = m \cdot c$ and $M'_j := 2^{M'_{j-1}}$ for $j \geq 2$,
3. set $N'_0 := c$ and $N'_j := M'_j \cdot 2^{N'_{j-1}}$ for $j \geq 1$,
4. set $N_0 := c$ and $N_j := M_j \cdot 2^{N_{j-1}}$ for $j \geq 1$, and
5. set $S_1 := (m+1) \cdot 3 \cdot c \cdot 2^c$ and $S_j := 2^{S_{j-1}}$ for $j \geq 2$.

Lemma G.4. *$M_i - M'_i \geq N'_{i-1}$ for all $i \geq 1$.*

Proof. The proof is by induction on i. For $i = 1$ we just have $M_1 - M'_1 = (m+1) \cdot c - m \cdot c = c = N'_0$. For $i \geq 2$ we have

$$\begin{aligned} M_i - M'_i &= 2^{M_{i-1}} - 2^{M'_{i-1}} = 2^{M'_{i-1}}(2^{M_{i-1}-M'_{i-1}} - 1) \\ &\geq 2^{M'_{i-1}}(2^{N'_{i-2}} - 1) \end{aligned}$$

where the inequality holds due to our induction hypothesis. Since $2^k \geq 2k$ for each $k \in \mathbb{N}$, we have

$$2^{M'_{i-1}}(2^{N'_{i-2}} - 1) \geq 2 \cdot M'_{i-1}(2^{N'_{i-2}} - 1).$$

Furthermore, $N'_{i-2} \geq 1$ implies that $2^{N'_{i-2}} - 1 \geq 2^{N'_{i-2}-1}$. Hence,

$$2 \cdot M'_{i-1}(2^{N'_{i-2}} - 1) \geq M'_{i-1}(2 \cdot 2^{N'_{i-2}-1}) = N'_{i-1}.$$

□

Lemma G.5. *$S_j \geq 3N_j$ for all $j \geq 1$.*

Proof. The proof is by induction on j. For $j = 1$ we have

$$S_1 = (m+1) \cdot 3 \cdot c \cdot 2^c = 3 \cdot M_1 \cdot 2^{N_0} = 3N_1.$$

Now assume that $j \geq 2$. Since $N_{j-1} \geq N_0 \geq 2$ holds, $2^{N_{j-1}} \geq 3$. We also have $N_{j-1} = M_{j-1} \cdot 2^{N_{j-2}} \geq M_{j-1}$, so $2^{N_{j-1}} \geq 2^{M_{j-1}} = M_j$. Due to the induction hypothesis, we have

$$\begin{aligned} S_j &= 2^{S_{j-1}} \geq 2^{3N_{j-1}} = 2^{N_{j-1}} \cdot 2^{N_{j-1}} \cdot 2^{N_{j-1}} \\ &\geq 3 \cdot M_j \cdot 2^{N_{j-1}} = 3N_j. \end{aligned}$$

□

H Proof of Theorem 1.1

In this section we complete the proof of our main theorem. We start with several technical lemmas which connect behaviour of runs described in terms of the stack sizes and in terms of the history function (Subsection H.1). Then we give a lemma ensuring that a run R satisfying a certain technical condition has a subrun S which decomposes into a pumping run and a top^0-non-erasing run (Subsection H.2). Next, in Subsection H.3 we use this lemma in order to give a bound on the size of stacks in finitely branching ε-contractions of collapsible pushdown graphs: given such a finitely branching contraction $\mathcal{G}$ and a configuration $c \in \mathcal{G}$ of distance m from the initial configuration, there is a bound on the size of the stack of c in terms of m and of the size and the level of the system $\mathcal{S}$ generating $\mathcal{G}$. Using this bound we derive a pumping construction in Subsection H.4 which proves the main theorem: for $\mathcal{G}$ and c as before, if there is a path starting in c of length above some bound depending on m and on the size and level of $\mathcal{S}$, then there start infinitely many paths in c.

H.1 Technical Lemmas

Lemma H.1. *Let $1 \leq k \leq n$, let R be a run, let x be a position of a k-stack of $R(|R|)$, and y the position of its topmost $(k-1)$-stack. Let a_i be the size of the k-stack of $R(i)$ at $\mathsf{hist}(x, R\restriction_{i,|R|})$ for each $0 \leq i \leq |R|$. Assume that $\mathsf{hist}(x, R) = \mathsf{top}^k(R(0))$ and that $a_0 = \min\{a_i : 0 \leq i \leq |R|\}$. Then $\mathsf{hist}(y, R) = \mathsf{top}^{k-1}(R(0))$.*

Recall that the size of a k-stack is just the number of its $(k-1)$-stacks. Before we prove the lemma, we state a proposition which is an immediate consequence of the definition of the history function (recall that for every position $x_1 \xrightarrow{k} x_2$ we have $x_2 \neq (0, \ldots, 0)$).

Proposition H.2. *Let S be a run of length 1, and y a position of a $(k-1)$-stack in $S(1)$ for some $1 \leq k \leq n$. Assume that the last non-zero coordinate of y and $\mathsf{hist}(y, S)$ differ. Then $\mathsf{hist}(y, S) = \mathsf{top}^{k-1}(S(0))$.*

Proof (Lemma H.1). Set $m := |R|$. Let b_i be the value of the last nonzero coordinate of $\mathsf{hist}(y, R\restriction_{i,m})$ for each $0 \leq i \leq m$ (notice that this is a level k coordinate, as $\mathsf{hist}(y, R\restriction_{i,m})$ always points to a $(k-1)$-stack). We claim that $b_i \geq a_0$ for each $0 \leq i \leq m$. To prove the claim we proceed by induction on i, from m to 0. For $i = m$ we have $b_m = a_m \geq a_0$. Let $i < m$. If $b_i = b_{i+1}$ we are done. By the above proposition, $b_i \neq b_{i+1}$ implies that $\mathsf{hist}(y, R\restriction_{i,m}) = \mathsf{top}^{k-1}(R(i))$. Due to Corollary D.10, $\mathsf{hist}(y, R\restriction_{i,m})$ points into $\mathsf{hist}(x, R\restriction_{i,m})$ whence a_i is the size of the topmost k-stack of $R(i)$. Thus, $b_i = a_i \geq a_0$.

Finally we show that $b_0 \geq a_0$. Let $z := \mathsf{hist}(y, R)$. Due to Proposition D.9 z points into $\mathsf{top}^k(R(0))$, and contains only links of level at most k. If z is simple, we are done: $b_0 \geq a_0$ implies that $z = \mathsf{top}^{k-1}(R(0))$. Otherwise, as z points to a $(k-1)$-stack, the last link in z is of level (at least) k. Recall the notation of the pack function from page 33. Coordinates of level greater than k in $\mathsf{pack}_{\mathsf{nr}(z)}(z)$

are the same as in $\mathsf{top}^k(R(0))$, the level k coordinate is $b_0 \geq a_0$, and coordinates of level smaller than k are zeroes. So $\mathsf{pack}_{\mathsf{nr}(z)}(z) \succeq \mathsf{top}^{k-1}(R(0))$, and points to a $(k-1)$-stack. By Corollary D.14 $z = \mathsf{top}^{k-1}(R(0))$ which contradicts our assumption that z is not simple. □

Lemma H.3. *Let $1 \leq k \leq n$, let R be a run with $m = |R|$, x a position of a k-stack of $R(|R|)$, and a_i the size of the k-stack of $R(i)$ at $\mathsf{hist}(x, R\restriction_{i,|R|})$ for each $0 \leq i \leq |R|$. Assume that $a_0 < a_i$ for all $0 < i < |R|$, and $a_0 \leq a_m$. Then R is a top^0-non-erasing run.*

Proof. If $m = 0$ there is nothing to show. Otherwise, $a_0 < a_1$ whence the first operation of R has to be push^k.

- First assume that $a_m > a_0$ whence $a_1 = \min\{a_i : 1 \leq i \leq m\}$. Let y be the topmost $(k-1)$-stack of the k-stack of $R(m)$ at x. Application of Lemma H.1 to $R\restriction_{1,m}$, x and y implies that $\mathsf{hist}(y, R\restriction_{1,m}) = \mathsf{top}^{k-1}(R(1))$. Due to Proposition E.21 applied to $R\restriction_{1,m}$ and $k-1$ we obtain that $R\restriction_{1,m}$ is a top^{k-1}-non-erasing run. Since $\mathsf{top}^0(R(0)) \preceq \mathsf{top}^{k-1}(R(1))$, $\mathsf{top}^0(R(0))$ cannot be removed by R if $\mathsf{top}^{k-1}(R(1))$ is not removed whence R is a top^0-non-erasing run.
- Otherwise, $a_m = a_0$. We apply the same argument as above, but to $R\restriction_{0,m-1}$. We obtain that $R\restriction_{0,m-1}$ is a top^0-non-erasing run. Since $a_{m-1} > a_m = a_0$ and since only the topmost k-stack can change its size, $x = \mathsf{top}^k(R(|R|))$, and the operation between $R(m-1)$ and $R(m)$ is pop^k or col^k. As the topmost k-stack of $R(m)$ has size a_0 and $\mathsf{top}^0(R(0))$ is present in $R(m-1)$, it is also present in $R(m)$.

□

Below we say that an l-stack s *occurs* in a k-stack t; this includes occurring inside a link, and includes $s = t$.

Lemma H.4. *Let $0 \leq j < k \leq n$, let R be some run, x some position of a k-stack in $R(|R|)$, and a_i the size of the k-stack at $\mathsf{hist}(x, R\restriction_{i,|R|})$ in $R(i)$. Assume that $a_i > a_{|R|}$ for all $0 \leq i < |R|$. Then every j-stack occurring in the k-stack at x in $R(|R|)$ occurs also in the k-stack at $\mathsf{hist}(x, R)$ in $R(0)$.*

Proof. Is is enough to prove, for $1 \leq b \leq a_{|R|}$ that the b-th $(k-1)$-stack of the k-stack at x in $R(|R|)$ is equal to the b-th $(k-1)$-stack of the k-stack at $\mathsf{hist}(x, R)$ in $R(0)$. We prove this by induction on the length of R. For $|R| = 0$ this is immediate. Let $|R| \geq 1$. In the light of the induction assumption, it is enough to prove, for $1 \leq b \leq a_{|R|}$, that the b-th $(k-1)$-stack of the k-stack at $\mathsf{hist}(x, R\restriction_{1,|R|})$ in $R(1)$ is equal to the b-th $(k-1)$-stack of the k-stack at $\mathsf{hist}(x, R)$ in $R(0)$.

- If the first operation in R is of level below k, then $\mathsf{hist}(x, R\restriction_{1,|R|}) = \mathsf{hist}(x, R)$ and only the topmost $(k-1)$-stack is modified; this is not one of the considered $(k-1)$-stacks, as $a_0 > a_{|R|}$.

– If the first operation in R is of level k, then $\mathsf{hist}(x, R\restriction_{1,|R|}) = \mathsf{hist}(x, R)$ and some $(k-1)$-stacks are removed or added, but none of the considered $(k-1)$-stacks, as $a_0, a_1 \geq a_{|R|}$ (this is also true for col^k, as performing col^k is equivalent to performing several pop^k).
– If the first operation in R is of level greater than k, then the whole k-stacks at $\mathsf{hist}(x, R\restriction_{1,|R|})$ in $R(1)$ and at $\mathsf{hist}(x, R)$ in $R(0)$ are the same.

□

Lemma H.5. *Let $1 \leq k \leq n$, R a pumping run and x a position of a k-stack in $R(|R|)$. Assume that the size of the k-stack at x in $R(|R|)$ is greater than that of the k-stack at $\mathsf{hist}(x, R\restriction_{i,|R|})$ in $R(i)$ for some i. Then $\mathsf{top}^0(R(0)) \neq \mathsf{top}^0(R(|R|))$.*

Proof. Let $m := |R|$, and let a_j be the size of the k-stack of $R(j)$ at $\mathsf{hist}(x, R\restriction_{j,|R|})$ for all $0 \leq j \leq m$. Take the maximal b such that $a_b < a_m$ (note that $i \leq b$). Since stack operations increase the number of stacks by at most one, $a_{b+1} = a_b + 1$ whence maximality of b implies $a_{b+1} = \min\{a_j : b+1 \leq j \leq m\}$. Set $S := R\restriction_{b+1,m}$. Notice that $\mathsf{hist}(x, S) = \mathsf{top}^k(S(0))$ because only the topmost k-stack can change its size. Since $\mathsf{hist}(\mathsf{top}^0(R(m)), R) = \mathsf{top}^0(R(0))$, Proposition E.21 implies that R is a top^0-non-erasing run. It means that $\mathsf{top}^{k-1}(R(0))$ is present in $R(b)$. Because the operation between $R(b)$ and $R(b+1) = S(0)$ is necessarily push^k, it implies that $\mathsf{top}^0(R(0)) \prec \mathsf{top}^{k-1}(S(0))$.

Application of Lemma H.1 to S and x shows that $\mathsf{hist}(y, S) = \mathsf{top}^{k-1}(S(0))$ for some position y in $R(m)$. Again using Proposition E.21, we conclude that S is a top^{k-1}-non-erasing run and we obtain

$$\mathsf{top}^0(R(0)) \prec \mathsf{top}^{k-1}(S(0)) \preceq \mathsf{top}^{k-1}(R(m)) \prec \mathsf{top}^0(R(m)).$$

□

H.2 Main Technical Lemma

Below we present our main technical lemma. It shows how to find subruns of long runs which consist of a pumping run followed by a top^0-non-erasing run. Recall that the function $\mathsf{ctype}_{\mathcal{X}}$ maps configurations to a finite set of types. For each collapsible pushdown system $\mathcal{S}$, let $\mathcal{T}_{\mathcal{S}}$ denote the image of $\mathsf{ctype}_{\mathcal{X}}$ with respect to configurations of $\mathcal{S}$.

Lemma H.6. *Let $\mathcal{S}$ be an n-CPS, $0 \leq k \leq n$, R be a run of $\mathcal{S}$, and*

$$G_k \subseteq \{i < |R| : \mathsf{hist}(\mathsf{top}^k(R(|R|)), R\restriction_{i,|R|}) = \mathsf{top}^k(R(i))\}.$$

Furthermore, let s^k be the k-stack of $R(0)$ to which $\mathsf{hist}(\mathsf{top}^k(R(|R|)), R)$ points. For $1 \leq j \leq k$, let r_j be the maximum of the sizes of j-stacks occurring in s^k. Let

$$\hat{N}_0 := |\mathcal{T}_{\mathcal{S}}| + 1 \text{ and } \hat{N}_j = r_j \cdot 2^{\hat{N}_{j-1}} \text{ for } 1 \leq j \leq k.$$

If $|G_k| \geq \hat{N}_k$, then there are $0 \leq x < y < z \leq |R|$ such that

1. $\mathsf{ctype}_{\mathcal{X}}(R(x)) = \mathsf{ctype}_{\mathcal{X}}(R(y))$,
2. $R{\restriction}_{x,y}$ *is a pumping run,*
3. $\mathsf{hist}(\mathsf{top}^k(R(|R|)), R{\restriction}_{y,|R|}) = \mathsf{top}^k(R(y))$,
4. $\mathsf{top}^0(R(x)) \neq \mathsf{top}^0(R(y))$ *or*

$$G_k \cap \{x, x+1, \dots, y-1\} \neq \emptyset,$$

5. $z - 1 \in G_k$, *and*
6. $R{\restriction}_{y,z}$ *is a* top^0*-non-erasing run.*

Proof. We prove the lemma by induction on k. Consider the case that $k = 0$. By assumption $|G_0| \geq |\hat{N}_0| > |\mathcal{T}_{\mathcal{S}}|$. Thus, there are $x, y \in G_0$ with $x < y$ such that $\mathsf{ctype}_{\mathcal{X}}(R(x)) = \mathsf{ctype}_{\mathcal{X}}(R(y))$. Since $x, y \in G_0$,

$$\mathsf{hist}(\mathsf{top}^0(R(|R|)), R{\restriction}_{x,|R|}) = \mathsf{top}^0(R(x)), \text{ and}$$
$$\mathsf{hist}(\mathsf{top}^0(R(|R|)), R{\restriction}_{y,|R|}) = \mathsf{top}^0(R(y)).$$

Due to Proposition D.5, we conclude that

$$\mathsf{hist}(\mathsf{top}^0(R(y)), R{\restriction}_{x,y}) = \mathsf{top}^0(R(x))$$

which means that $R{\restriction}_{x,y}$ is a pumping run. Since $x \in G_0$, we have $G_0 \cap \{x, x+1, \dots, y-1\} \neq \emptyset$. By definition of y, $R{\restriction}_{y,|R|}$ is a pumping run of length at least 1. Due to the characterisation of pumping runs (cf. Lemma E.26), this run starts with some push operation. Thus, for $z := y + 1$, we have $z - 1 \in G_0$ and $R{\restriction}_{y,z}$ is a top^0-non-erasing run. Thus, x, y, and z satisfy the claim of the lemma.

Now consider the case $k \geq 1$ and assume that the lemma holds for all $k' < k$. Let a_i be the size of the k-stack of $R(i)$ at position $\mathsf{hist}(\mathsf{top}^k(R(|R|)), R{\restriction}_{i,|R|})$ for $0 \leq i \leq |R|$. Due to Proposition D.6 we know that $a_i - a_{i-1} \leq 1$ for all $1 \leq i \leq |R|$. By definition $a_0 \leq r_k$ whence $|G_k| \geq \hat{N}_k \geq a_0 \cdot 2^{\hat{N}_{k-1}}$. Hence, we can apply Corollary G.2 to $(a_i)_{0 \leq i \leq |R|}$ and obtain indices $0 \leq b < e \leq |R|$ such that

1. $e - 1 \in G_k$,
2. $a_b = \min\{a_i : b \leq i \leq e\}$,
3. $a_i > a_b$ for all $0 \leq i < b$ and
4. $|H_{b,e}| \geq \hat{N}_{k-1}$ where

$$\begin{aligned} H_{b,e} = \{i : b \leq i \leq e - 1,& \\ & a_i \leq a_j \text{ for all } i \leq j \leq e, \text{ and} \\ & a_i < a_j \text{ for all } i < j \leq n_{G_k}(i)\} \end{aligned}$$

with $n_{G_k}(i) := \min\{g \in G_k : g \geq i\}$.

Set $R' := R{\restriction}_{b,e}$ and $G_{k-1} := \{h - b : h \in H_{b,e}\}$. Let us first assume that the following claims are true:

A) for each $h \in H_{b,e}$ we have $\mathsf{hist}(\mathsf{top}^{k-1}(R(e)), R{\restriction}_{h,e}) = \mathsf{top}^{k-1}(R(h))$,

B) for all $i \leq e-1$, $\mathsf{hist}(\mathsf{top}^k(R(e)), R{\restriction}_{i,e}) = \mathsf{hist}(\mathsf{top}^k(R(|R|)), R{\restriction}_{i,|R|})$, whence a_i is the size of the k-stack in $R(i)$ at $\mathsf{hist}(\mathsf{top}^k(R(e)), R{\restriction}_{i,e})$, and
C) if t^{k-1} is the $(k-1)$-stack at $\mathsf{hist}(\mathsf{top}^{k-1}(R'(|R'|)), R')$, then the size of every j-stack occurring in t^{k-1} for $j \leq k-1$ is bounded by r_j.

We postpone the proof of these claims. Claim A implies (by shifting from R to R') that for each $g \in G_{k-1}$ we have $\mathsf{hist}(\mathsf{top}^{k-1}(R'(|R'|)), R'{\restriction}_{g,|R'|}) = \mathsf{top}^{k-1}(R'(g))$. Together with Claim C this allows us to apply the induction hypothesis to $k-1$, R' and G_{k-1}. We obtain three indices $0 \leq x' < y' < z' \leq |R'|$; let $x = x'+b$, $y = y'+b$, and let z be the smallest index such that $z \geq z'+b$ and $z-1 \in G_k$ (it exists because $z'+b \leq e$ and $e-1 \in G_k$). Note that

1'. $\mathsf{ctype}_{\mathcal{X}}(R(x)) = \mathsf{ctype}_{\mathcal{X}}(R(y))$,
2'. $R{\restriction}_{x,y}$ is a pumping run,
3'. $\mathsf{hist}(\mathsf{top}^{k-1}(R(e)), R{\restriction}_{y,e}) = \mathsf{top}^{k-1}(R(y))$,
4'. $\mathsf{top}^0(R(x)) \neq \mathsf{top}^0(R(y))$ or

$$H_{b,e} \cap \{x, x+1, \ldots, y-1\} \neq \emptyset,$$

5'. $z'+b-1 \in H_{b,e}$, and
6'. $R{\restriction}_{y,z'+b}$ is a top^0-non-erasing run.

Note that items 1' and 2' coincide with items 1 and 2 of the lemma. We now prove items 3 – 6.

3. Due to Corollary D.10, item 3' implies that

$$\mathsf{hist}(\mathsf{top}^k(R(e)), R{\restriction}_{y,e}) = \mathsf{top}^k(R(y)).$$

   Together with Claim B this yields item 3.
4. Assume that $\mathsf{top}^0(R(x)) = \mathsf{top}^0(R(y))$. Due to 4', there is some

$$h \in H_{b,e} \cap \{x, x+1, \ldots, y-1\} \neq \emptyset.$$

   Items 2', 3', and Claim A, after application of Corollary D.10, imply

$$\begin{aligned} \mathsf{hist}(\mathsf{top}^j(R(y)), R{\restriction}_{x,y}) &= \mathsf{top}^j(R(x)), \\ \mathsf{hist}(\mathsf{top}^j(R(e)), R{\restriction}_{y,e}) &= \mathsf{top}^j(R(y)), \quad \text{and} \\ \mathsf{hist}(\mathsf{top}^j(R(e)), R{\restriction}_{h,e}) &= \mathsf{top}^j(R(h)) \end{aligned}$$

   for all $j \geq k-1$. Due to Proposition D.5, this implies that

$$\mathsf{hist}(\mathsf{top}^j(R(b)), R{\restriction}_{a,b}) = \mathsf{top}^j(R(a)) \tag{4}$$

   for each pair $a, b \in \{x, h, y, e\}$ with $a \leq b$. With two applications of Lemma D.11 (to $R{\restriction}_{h,y}$ and $R{\restriction}_{x,h}$) we obtain that $\mathsf{top}^{k-1}(R(h))$ is lexicographically bounded by $\mathsf{top}^{k-1}(R(x)) = \mathsf{top}^{k-1}(R(y))$ from below and from above whence it is this position. Claim B and equation (4) (setting $j = k$) imply that a_x, a_h and a_y are the sizes of the topmost k-stacks of x, h and y, respectively. It follows that $a_x = a_h = a_y$. Since $h \in H_{b,e}$, there exists some $g \in G_k$ such that $x \leq h \leq g$ and $a_j > a_h$ for all $h < j \leq g$. As $a_y = a_h$, we conclude that $g < y$ whence $G_k \cap \{x, x+1, \ldots, y-1\} \neq \emptyset$.

5. $z-1 \in G_k$ is satisfied by definition of z.
6. If $z = z' + b$, items 6 and 6' coincide. Assume that $z > z' + b$. Because $z' + b - 1 \in H_{b,e}$, we know that $a_j > a_{z'+b-1}$ for $z' + b \le j \le z - 1$ because z is minimal such that $z - 1 \ge z' + b - 1$ and $z - 1 \in G_k$. In particular, $z > z' + b$ implies $a_{z'+b} > a_{z'+b-1}$. Since $z \le e$, $z' + b - 1 \in H_{b,e}$ implies also that $a_z \ge a_{z'+b-1}$. Thus, Lemma H.3 can be applied to $R{\restriction}_{z'+b-1,z}$. It follows that $R{\restriction}_{z'+b-1,z}$ is a top^0-non-erasing run. Since $R{\restriction}_{y,z'+b}$ is also a top^0-non-erasing run, $R{\restriction}_{y,z}$ is one as well (cf. Proposition E.20).

Thus, x, y and z satisfy the lemma if Claims A – C hold. We continue with a simultaneous proof of Claims A and B. We start with showing that for each $h \in H_{b,e}$

$$\mathsf{hist}(\mathsf{top}^k(R(|R|)), R{\restriction}_{h,|R|}) = \mathsf{top}^k(R(h)). \tag{5}$$

Consider any $h \in H_{b,e}$. If $h \in G_k$, the condition is satisfied by definition of G_k. Otherwise, we conclude that $a_{h+1} > a_h$. But only the topmost k-stack can change its size whence equation (5) holds.

Recall that $e - 1 \in G_k$, which implies that

$$\mathsf{hist}(\mathsf{top}^k(R(|R|)), R{\restriction}_{e-1,|R|}) = \mathsf{top}^k(R(e-1)). \tag{6}$$

Together with equation (5) this implies

$$\begin{aligned}&\mathsf{hist}(\mathsf{top}^k(R(e-1)), R{\restriction}_{h,e-1}) = \\ &\quad = \mathsf{hist}(\mathsf{top}^k(R(|R|)), R{\restriction}_{h,|R|}) = \mathsf{top}^k(R(h))\end{aligned}$$

for each $h \in H_{b,e}$. By definition of $H_{b,e}$, $a_h = \min\{a_i : h \le i \le e\}$. Additionally, equation (6) implies that a_i (for $b \le i \le e-1$) is the size of the k-stack of $R(i)$ at $\mathsf{hist}(\mathsf{top}^k(R(e-1)), R{\restriction}_{i,e-1})$, whence we may apply Lemma H.1 to $x := \mathsf{top}^k(R(e-1))$ and to $R{\restriction}_{h,e-1}$. This yields

$$\mathsf{hist}(\mathsf{top}^{k-1}(R(e-1)), R{\restriction}_{h,e-1}) = \mathsf{top}^{k-1}(R(h)) \tag{7}$$

for each $h \in H_{b,e}$.

We continue by case distinction on the operation between $e-1$ and e in R.

1. Due to equation (6), the operation at $e-1$ cannot be $\mathsf{pop}^{k'}$ or $\mathsf{col}^{k'}$ for $k' > k$.
2. If the operation at $e-1$ is of level below k or is a push operation, then

   $$\mathsf{hist}(\mathsf{top}^{k-1}(R(e)), R{\restriction}_{e-1,e}) = \mathsf{top}^{k-1}(R(e-1)).$$

   Due to equation (7), this implies Claim A. Together with (6) and Corollary D.10, this implies

   $$\mathsf{hist}(\mathsf{top}^k(R(e)), R{\restriction}_{e-1,e}) = \mathsf{hist}(\mathsf{top}^k(R(|R|)), R{\restriction}_{e-1,|R|}).$$

   Using Proposition D.5, Claim B follows directly.

3. Assume that the operation at $e-1$ is pop^k or col^k. We conclude immediately that
$$\mathsf{hist}(\mathsf{top}^k(R(|R|)), R{\restriction}_{e,|R|}) = \mathsf{top}^k(R(e)),$$
because this is the only position p of $R(e)$ that satisfies $\mathsf{hist}(p, R{\restriction}_{e-1,e}) = \mathsf{top}^k(R(e-1))$ (and because $e-1 \in G_k$). With Proposition D.5, Claim B follows directly.
Furthermore, a_e is the size of the stack of $R(e)$ at $\mathsf{top}^k(R(e))$. By definition of $H_{b,e}$, we have $a_h = \min\{a_i : h \leq i \leq e\}$. Application of Lemma H.1 to $x := \mathsf{top}^k(R(e))$ and to $R{\restriction}_{h,e}$ for each $h \in H_{b,e}$ yields Claim A.

For the proof of Claim C, let t^{k-1} be the $(k-1)$-stack of $R'(0)$ at the position $\mathsf{hist}(\mathsf{top}^{k-1}(R'(|R'|)), R')$ which is by definition the $(k-1)$-stack of $R(b)$ at $\mathsf{hist}(\mathsf{top}^{k-1}(R(e)), R{\restriction}_{b,e})$. Due to Corollary D.10, $\mathsf{hist}(\mathsf{top}^{k-1}(R(e)), R{\restriction}_{b,e})$ points into $\mathsf{hist}(\mathsf{top}^k(R(e)), R{\restriction}_{b,e})$. Hence, for $j \leq k-1$ every j-stack occurring in t^{k-1} occurs also in the k-stack of $R(b)$ at $\mathsf{hist}(\mathsf{top}^k(R(e)), R{\restriction}_{b,e})$. Due to Claim B, a_i is the number of $(k-1)$-stacks of the stack at $\mathsf{hist}(\mathsf{top}^k(R(e)), R{\restriction}_{i,e})$ for all $i \leq b$, and the k-stack of $R(0)$ at $\mathsf{hist}(\mathsf{top}^k(R(e)), R{\restriction}_{0,e})$ is s^k. We have $a_i > a_b$ for all $0 \leq i < b$, so we can apply Lemma H.4 to $R{\restriction}_{0,b}$ and position $\mathsf{hist}(\mathsf{top}^k(R(e)), R{\restriction}_{b,e})$. We conclude that for $j \leq k-1$ every j-stack occurring in t^{k-1} occurs also in s^k. Thus, its size is bounded by r_j. □

H.3 Finitely Branching Epsilon-Contractions

The basic proof concept for the pumping lemma is as follows. If we find a pumping run which starts and ends in configurations of the same type, then we can apply Proposition F.1 to this run and obtain arbitrarily long runs in the graph of the CPS. But if we consider ε-contractions, all runs that we construct may consist of ε-edges except for a bounded number of transitions. In this case, the longer and longer runs would perhaps always induce the same path in the ε-contraction. In this section we show how to overcome this problem in the case of finitely branching ε-contractions.

We first derive a technical condition that allows to conclude that the ε-contraction of some collapsible pushdown graph is infinitely branching. This result basically uses the naive pumping approach described before but we add some assumptions such that we really obtain larger and larger runs that end in larger and larger stacks that belong to the nodes of the ε-contraction. Afterwards, we use this result in order to define a bound on the difference of stack sizes between two nodes of a finitely branching ε-contraction that are connected by an edge. In the next section we use this fact in the pumping construction in the following way: instead of talking about a configuration being in some distance from the initial configuration in the ε-contraction, we talk about a configuration having stack sizes bounded by some numbers.

Without loss of generality (by doubling the number of states of the system), we can assume that for each state q transitions leading to state q are all ε-transitions or are all non-ε-transitions.

Proposition H.7. *Let $\mathcal{S}$ be some CPS of level n such that for each state q transitions leading to state q are all ε-transitions or are all non-ε-transitions. Let R be a run starting in a configuration of the ε-contraction $\mathcal{G}$ of the graph of $\mathcal{S}$. Then $\mathcal{G}$ is infinitely branching if there are positions $0 \leq x < y \leq |R|$ such that*

1. $\mathsf{ctype}_{\mathcal{X}}(R(x)) = \mathsf{ctype}_{\mathcal{X}}(R(y))$,
2. $R{\restriction}_{x,y}$ *is a pumping run in $\mathcal{P}_{>,\varepsilon}$, i.e., a pumping run such that* $\mathsf{top}^0(R(x)) \prec \mathsf{top}^0(R(y))$ *and all edges of $R{\restriction}_{x,y}$ are labelled by ε, and*
3. $R{\restriction}_{y,|R|}$ *is a top^0-non-erasing run ending with a non-ε-transition.*

Proof. Let q be the state of $R(y)$. We apply Proposition F.1 to $R{\restriction}_{x,y}$ and obtain infinitely many ε-labelled runs $(R_i)_{i\in\mathbb{N}}$ from $R(x)$ to $c_i = (q, s_i)$ such that $\mathsf{top}^0(s_i) \prec \mathsf{top}^0(s_{i+1})$ for all $i \in \mathbb{N}$. Now we apply Proposition F.2 to $R{\restriction}_{y,|R|}$ and to c_i. We obtain a top^0-non-erasing run S_i' from c_i. It ends in the same state as R whence it ends with a non-ε-transition. Let S_i be the prefix of S_i' which ends after the first occurrence of a non-ε-transition. Let $z \leq x$ be maximal such that $R(z)$ corresponds to a node of $\mathcal{G}$. Then $U_i := R{\restriction}_{z,y} \circ R_i \circ S_i$ connects $R(z)$ to one of its successors in $\mathcal{G}$ whose stack t_i contains the position $\mathsf{top}^0(s_i)$. Since t_i contains only finitely many positions, and the $(\mathsf{top}^0(s_j))_{j\geq i}$ form an infinite sequence of pairwise distinct positions, for each i there is a $j \geq i$ such that $\mathsf{top}^0(s_j)$ is no position in t_i. This immediately implies $t_i \neq t_j$. By induction, we conclude that the U_i connect $R(z)$ with infinitely many pairwise different successors in $\mathcal{G}$ whence $\mathcal{G}$ is infinitely branching at $R(z)$. □

Now we are prepared to prove that in each finitely branching ε-contraction of a collapsible pushdown system the stack sizes grow only in a bounded manner from each node to its successors. For the combinatorial part in the proof we use the sequences from Definition G.3 without reference.

Lemma H.8. *Let $\mathcal{S}$ be a CPS of level n such that the ε-contraction $\mathcal{G}$ of its configuration graph is finitely branching and such that transitions leading to some state q are either all ε-transitions or all non-ε-transitions. Set $c := |\mathcal{T}_{\mathcal{S}}| + 1$. Let R be a run starting in the initial configuration whose last edge is not labelled by ε and which corresponds to a path of length m in $\mathcal{G}$. The size of every k-stack of $R(|R|)$ is at most M_k for all $1 \leq k \leq n$.*

Proof. The proof is by induction on m. For $m = 0$, the claim is trivial (because $c \geq 2$ and the initial stack of any level has size 1). Assume that we have proven the claim for all paths of length below m and assume that R describes a path of length m in $\mathcal{G}$. Let $R(b)$ correspond to the $(m-1)$-st node of $\mathcal{G}$ on this path and set $S := R{\restriction}_{b,|R|}$.

Heading for a contradiction assume that p is the position of a k-stack in $R(|R|)$ such that the size of this stack is greater than M_k.

For $0 \leq i \leq |S|$, let a_i be the size of the k-stack of $S(i)$ at $\mathsf{hist}(p, S{\restriction}_{i,|S|})$. By induction hypothesis, the size of any k-stack of $S(0)$ is bounded by M_k'. Thus, we have $a_{|S|} > M_k$ and $a_0 \leq M_k'$. Let $G \subseteq \{0, 1, \dots, |S|-1\}$ contain all elements i such that $a_i < a_j$ for all $i < j \leq |S|$. Since $a_i - a_{i-1} \leq 1$ for $1 \leq i \leq |S|$, for each i in $\{M'_k, M'_k+1, \dots, M_k\}$ we have an index j such that $a_j = i$ and $a_j \in G$. Using Lemma G.4 we conclude that $|G| \geq M_k - M'_k \geq N'_{k-1}$. Since M'_k is a bound on the sizes of k-stacks in $S(0)$, it follows that G is big enough in order to apply Lemma H.6 for $k-1$. We want to apply this lemma to the run $T := S{\restriction}_{0,\max(G)+1}$.

In order to satisfy the requirements of this lemma, we have to prove that $\mathsf{hist}(\mathsf{top}^{k-1}(T(|T|)), T{\restriction}_{g,|T|}) = \mathsf{top}^{k-1}(T(g))$ for all $g \in G$. Choose $g \in G$ arbitrarily. Since $a_{g+1} > a_g$, the k-stack at $\mathsf{hist}(p, S{\restriction}_{g,|S|})$ is smaller than that at $\mathsf{hist}(p, S{\restriction}_{g+1,|S|})$. Due to Proposition D.6, this requires that

$$\mathsf{hist}(p, S{\restriction}_{g,|S|}) = \mathsf{top}^k(S(g)), \quad \text{and}$$
$$\mathsf{hist}(p, S{\restriction}_{g+1,|S|}) = \mathsf{top}^k(S(g+1)).$$

Especially, $\mathsf{hist}(p, S{\restriction}_{|T|,|S|}) = \mathsf{top}^k(T(|T|))$ whence a_i for $i \leq |T|$ is the size of the stack at $\mathsf{hist}(\mathsf{top}^k(T(|T|)), T{\restriction}_{i,|T|})$. We conclude that for all $g \in G$ we have

$$\mathsf{hist}(\mathsf{top}^k(T(|T|)), T{\restriction}_{g,|T|}) = \mathsf{top}^k(T(g)).$$

Furthermore, for each $i > g$ we have $a_g < a_i$ whence we can apply Lemma H.1 to $x := \mathsf{top}^k(T(|T|))$ and to the run $R{\restriction}_{g,|T|}$ obtaining that

$$\mathsf{hist}(\mathsf{top}^{k-1}(T(|T|)), T{\restriction}_{g,|T|}) = \mathsf{top}^{k-1}(T(g)).$$

Application of Lemma H.6 to T and $k-1$ yields indices $0 \leq x < y < z \leq |T|$ such that

1. $\mathsf{ctype}_{\mathcal{X}}(S(x)) = \mathsf{ctype}_{\mathcal{X}}(S(y))$,
2. $S{\restriction}_{x,y}$ is a pumping run,
4. $\mathsf{top}^0(S(x)) \neq \mathsf{top}^0(S(y))$ or

$$G \cap \{x, x+1, \dots, y-1\} \neq \emptyset,$$

5. $z - 1 \in G$, and
6. $S{\restriction}_{y,z}$ is a top^0-non-erasing run.

By definition of G, for $g \in G$ and $g < y$ we have $a_g < a_y$. In other words the size of the k-stack in $S(y)$ at $\mathsf{hist}(p, S{\restriction}_{y,|S|})$ is greater than that of the k-stack in $S(g)$ at $\mathsf{hist}(p, S{\restriction}_{g,|S|})$. Thus, if there is a $g \in G \cap \{x, x+1, \dots, y-1\}$, application of Lemma H.5 to $S{\restriction}_{x,y}$ shows that $\mathsf{top}^0(S(x)) \neq \mathsf{top}^0(S(y))$. In the light of Property 4), we always have $\mathsf{top}^0(S(x)) \neq \mathsf{top}^0(S(y))$.

Since $z-1 \in G$, we have $a_i > a_{z-1}$ for all $z \leq i \leq |S|$. Application of Lemma H.3 to $S{\restriction}_{z-1,|S|}$ shows that $S{\restriction}_{z-1,|S|}$ is a top^0-non-erasing run. Since Property 6) implies that $S{\restriction}_{y,z-1}$ is top^0-non-erasing, we conclude using Proposition E.20 that $S{\restriction}_{y,|S|}$ is top^0-non-erasing run.

Recall that the last edge of S is the only edge which is not labelled ε. Thus the assumptions of Proposition H.7 are satisfied by S, x and y whence the lemma yields that $\mathcal{G}$ is infinitely branching. This contradicts our assumption. Thus, we conclude that every j-stack in $R(|R|)$ has size bounded by M_j. $\square$

H.4 Proof of the Pumping Lemma

Having bounded the size of stacks in finitely branching ε-contractions of pushdown graphs, we can prove the main theorem.

Note that – doubling the number of states of the system – we can enforce that for each ε-transition δ_1 and each non-ε-transition δ_2, δ_1 leads to a different state than δ_2.

Having obtained this condition the proof of the main theorem follows from the following theorem.

Theorem H.9. *Let $\mathcal{S}$ be a CPS of level n such that the ε-contraction $\mathcal{G}$ of its graph is finitely branching and such that for each state q transitions leading to state q are all ε-transitions or are all non-ε-transitions. Let c_m be some configuration of distance m from the initial configuration.*

Let $S_1 = (m+1) \cdot C_{\mathcal{S}}$ and $S_j = 2^{S_{j-1}}$ for $2 \leq j \leq n$, where $C_{\mathcal{S}} = 3 \cdot c \cdot 2^c$ with $c = |\mathcal{T}_{\mathcal{S}}| + 1$. Assume also that in $\mathcal{G}$ there exists a path p of length at least S_n which starts in c_m.

Then there are infinitely many paths in $\mathcal{G}$ which start in c_m and end in configurations having the same state as the last configuration of p.

Proof. From Definition G.3 we obtain sequences M_i and N_i. Note that the sequence S_i defined in this lemma and the sequence S_i defined in that definition agree. Due to the existence of p, there is a run R starting in c_m such that S_n transitions in R are not labelled by ε and especially the last transition is not labelled ε. Let G be the set of those $0 \leq i < |R|$ such that the transition between $R(i)$ and $R(i+1)$ is not labelled ε. Since $\mathcal{S}$ is of level n, for any configuration c' of $\mathcal{S}$ the only position of an n-stack in c' is $\mathsf{top}^n(c') = (0, 0, \ldots, 0)$. Especially, every $g \in G$ satisfies $\mathsf{hist}(\mathsf{top}^n(R(|R|)), R{\restriction}_{g,|R|}) = \mathsf{top}^n(R(g))$. Furthermore, we saw in Lemma H.8 that M_i is an upper bound for the size of each i-stack in c_m and for each $1 \leq i \leq m$. Thus, Lemma G.5 implies that $|G| = S_n \geq 3N_n > \hat{N}_n$ and we can apply Lemma H.6 to R. We obtain numbers $0 \leq x < y < z \leq |R|$ such that

1. $\mathsf{ctype}_{\mathcal{X}}(R(x)) = \mathsf{ctype}_{\mathcal{X}}(R(y))$,
2. $R_1 := R{\restriction}_{x,y}$ is a pumping run,
4. $\mathsf{top}^0(R(x)) \neq \mathsf{top}^0(R(y))$ or
$$G \cap \{x, x+1, \ldots, y-1\} \neq \emptyset,$$
5. $z - 1 \in G$, and
6. $R_2 := R{\restriction}_{y,z}$ is a top^0-non-erasing run.

$G \cap \{x, x+1, \ldots, y_1\} = \emptyset$ is equivalent to saying that all labels in R_1 are ε. Moreover, since $\mathsf{top}^0(R(x)) \neq \mathsf{top}^0(R(y))$ in this case, we conclude that $R_1 \in \mathcal{P}_{>,\varepsilon}$. As $z-1 \in G$, $R_{y,z}$ ends by a non-ε-transition. Thus, Proposition H.7 implies that $\mathcal{G}$ is infinitely branching which contradicts our assumptions.

Thus, R_1 contains at least one edge with a label different from ε. Due to Proposition F.1, there are runs $(S_i)_{i \in \mathbb{N}}$ such that

– S_i starts in $R(x)$,
– contains at least i transitions whose label is not ε and
– $\mathsf{ctype}_{\mathcal{X}}(R(x)) \sqsubseteq \mathsf{ctype}_{\mathcal{X}}(S_i(|S_i|))$.

Let T_i be the copy of $R{\restriction}_{y,|R|}$ obtained by application of Proposition F.3 starting in $S_i(|S_i|)$. Then $U_i := R{\restriction}_{0,x} \circ S_i \circ T_i$ is a run from c_m to $e_i := T_i(|T_i|)$ that contains at least i non-ε labelled edges. Furthermore, the state of e_i is the final state of R. Due to our assumption on the pushdown system, this state determines whether the edge to e_i is labelled ε. Since the last edge of R is not labelled ε, the edge to e_i is not labelled ε, whence e_i is a node in $\mathcal{G}$. Thus, U_i induces a path of length at least i starting in c_m and ending in a configuration with the same state as the final configuration of p. □

I Collapsible Pushdown Systems as Tree Generators

In this section we describe how collapsible pushdown system can be used to generate trees and we show that part 2 of Corollary 1.2 follows from Theorem 1.1. We consider ranked, potentially infinite trees. We fix an alphabet A of tree labels and a function $rank\colon A \to \mathbb{N}$. Some node of a tree labelled by $a \in A$ has always $rank(a)$ many children.

We say that a system $\mathcal{S}$ generates a tree over alphabet $(A, rank)$ if it satisfies the following (syntactical and semantical) restrictions.

1. The input alphabet of $\mathcal{S}$ is $A \cup \{0, 1, \dots, m-1\}$, where $m = \max\{rank(a) : a \in A\}$.
2. The state set of $\mathcal{S}$ can be partitioned into $Q_\varepsilon, Q_0, Q_1, \dots, Q_m$ such that the following holds for every stack symbol γ. For each state $q \in Q_\varepsilon$ there is at most one transition (q, γ, a, p, op), and $a \in A \cup \{\varepsilon\}$; $p \in Q_{rank(a)}$ if $a \in A$, and $p \in Q_\varepsilon$ if $a = \varepsilon$. For each state $q \in Q_i$ $(0 \le i \le m)$ there are exactly i transitions (q, γ, a, p, op); for each of them a is a different number from $\{0, 1, \dots, i-1\}$, and for each of them $p \in Q_\varepsilon$. Additionally, the initial state is in Q_ε.
3. From each configuration of $\mathcal{S}$ reachable from the initial one and having a state in Q_ε there exists a run to a configuration having a state in $Q \setminus Q_\varepsilon$. From each configuration of $\mathcal{S}$ reachable from the initial one and having the state in some Q_i $(0 \le i \le m)$, all of the i transitions are applicable.

Definition I.1. The *tree generated by a system* $\mathcal{S}$ has as nodes runs from the initial configuration to a configuration having a state in $Q \setminus Q_\varepsilon$. A node R is labelled by $a \in A$ if the last transition of R is labelled by a. A node S is the i-th child $(0 \le i \le rank(a)-1)$ of R if $S = R \circ T$ where the first edge of T is labelled by i and it is the only edge of T labelled by a number from $\{0, 1, \dots, m-1\}$.

The conditions on $\mathcal{S}$ guarantee that the above definition really defines an A-labelled ranked tree. Condition 2 says that the system behaves in a deterministic way if the state is in Q_ε. It performs several ϵ-transitions and, finally, a transition reading a letter a from A; this generates a tree node having letter a. Immediately after that the state is in $Q_{rank(a)}$, so there are $rank(a)$ possible transitions; they correspond to the children of the node just generated. Condition 3 guarantees that this construction will never block.

Now we come to the proof of the second part of Corollary 1.2. Let $A = \{a, b, c\}$, where $rank(a) = 2$, $rank(b) = 1$, and $rank(c) = 0$). For level n consider the tree T_n in which

- the rightmost path is labelled by a, and
- the left subtree of the i-th a-labelled node is a path consisting of $\exp_n(i)$ many b-labelled nodes, ending with a c-labelled node.

It is known that T_n can be generated by a pushdown system (without collapse) of level $n+1$. (cf. Example 9 in [2], where Blumensath provides a very similar pushdown system).

Assume that there exists a collapsible pushdown system of level n which generates T_n. Let $\mathcal{S}$ be the system obtained from it by replacing every A-labelled transition by an ε-transition (so we leave only labels 0 and 1; we remove a, b, c for simplicity). Let $\mathcal{G}$ be the ε-contraction of the configuration graph of $\mathcal{S}$. Let m be a number such that $\exp_n(m-1) > \exp_{n-1}((m+1) \cdot C_{\mathcal{S}L})$, where $C_{\mathcal{S}L}$ is the constant from Theorem 1.1 for $L = \{0,1\}^*$. Let c_m be the node of $\mathcal{G}$ such that the path from the initial configuration to c_m is labelled by $1^{m-1}0$. By definition of $\mathcal{S}$ such node exists, and in $\mathcal{G}$ there exists a path p from c_m of length $\exp_n(m-1) - 1$ (labelled by zeroes). Application of Theorem 1.1 yields arbitrarily long paths from c_m which contradicts with our assumption about the form of the tree generated by $\mathcal{S}$.

J Decidability of Finite Branching

In this section we show that types can be used to decide whether a given n-CPS $\mathcal{S}$ generates a configuration graph whose ε-contraction is finitely branching. As a consequence we obtain also an algorithm checking whether this ε-contraction is finite, and whether its unfolding into a tree is finite.

Let us remark that the same can be shown in a nontrivial way using decidability of μ-calculus on configuration graphs of n-CPS's, and using (multiple times) the reflection of n-CPS's with respect to the μ-calculus (i.e., the result from [A1]). This algorithm (at least its variant which we have in mind) works in m-EXPTIME for some $m = O(n^2)$; the reason is that each use of the μ-calculus reflection increases the size of the system (more or less) n-times exponentially, and we use it (more or less) n times.

The proof using types is very elegant: first we observe that Proposition H.7 basically says that an ε-contraction of a configuration graph is infinitely branching if and only if it contains a pumping run from $\mathcal{P}_{<,\varepsilon}$ that starts and ends in a stack of the same type. Due to the pumpability of pumping runs, this is the same as saying there are arbitrarily large sequences of pumping runs from $\mathcal{P}_{<,\varepsilon}$. Thus, checking for infinite branching is the same as checking for long sequences of pumping runs. The second ingredient of our proof is the fact that families defined by well-formed rules are closed under composition (cf. Lemma 4.4). Thus, for a well-chosen family $\mathcal{Y}$, the function $\mathsf{ctype}_{\mathcal{Y}}$ yields informations about long sequences of pumping runs and we only have to check whether the initial configuration has a type that witnesses such a sequence in order to decide infinite branching of the ε-contraction of a configuration graph.

As previously we may assume that for each state q transitions leading to state q are all ε-transitions or are all non-ε-transitions. Let $\mathcal{S}$ be such system, and $\mathcal{G}$ be the ε-contraction of the configuration graph of $\mathcal{S}$. Let $\mathcal{X}$ be the family of sets of runs defined in Section 5. Recall that it contains the set $\mathcal{P}_{<,\varepsilon}$ of pumping runs increasing the stack and using only ε-transitions, the set $\mathcal{N}_0$ of top^0-non-erasing-runs, and the set $\mathcal{Q}$ of all runs.

We begin by giving a "if and only if" version of Proposition H.7.

Claim. System $\mathcal{S}$ is infinitely branching if and only if there exists a run R from the initial configuration, and indices $0 \leq x < y \leq |R|$ such that

1. $\mathsf{ctype}_{\mathcal{X}}(R(x)) = \mathsf{ctype}_{\mathcal{X}}(R(y))$,
2. $R{\restriction}_{x,y}$ is a pumping run in $\mathcal{P}_{<,\varepsilon}$, i.e., a pumping run such that $\mathsf{top}^0(R(x)) \prec \mathsf{top}^0(R(y))$ and all edges of $R{\restriction}_{x,y}$ are labelled by ε, and
3. $R{\restriction}_{y,|R|}$ is a top^0-non-erasing run, i.e., a run in $\mathcal{N}_0$, ending with a non-ε-transition.

Proof. The right-to-left implication is just Proposition H.7. For the opposite direction we inspect the proof of Lemma H.8 Assume that $\mathcal{G}$ is infinitely branching. Then for some m the thesis of Lemma H.8 is not satisfied: there exists a run R from the initial configuration which corresponds to a path of length m in $\mathcal{G}$ such that for some k the size of some k-stack of $R(|R|)$ is greater than M_k (as otherwise we trivially have finite branching). Choose the minimal such m. Notice that the proof of Lemma H.8 goes by contradiction: it indeed assumes that such run R exists. As a conclusion on the end of the proof we obtain a run S which satisfies assumptions of Proposition H.7. This is almost what we need, but S does not necessarily start in the initial configuration. However for sure it starts in a reachable configuration, so we can append at the beginning of S the run from the initial configuration to $S(0)$; such run still satisfies the conditions on the right side of our claim. $\square$

Now consider a family $\mathcal{Y}$ (described by wf-rules) containing the set of runs

$$\mathcal{V} = \mathcal{Q} \circ \mathcal{P}_{<,\varepsilon} \circ \mathcal{P}_{<,\varepsilon} \circ \cdots \circ \mathcal{P}_{<,\varepsilon} \circ \mathcal{N}_0,$$

where the number of the $\mathcal{P}_{<,\varepsilon}$ factors is a fixed number greater than the number of possible values of $\mathsf{ctype}_{\mathcal{X}}$. Due to the claim a system is infinitely branching if and only if there is a run from $\mathcal{V}$ starting in the initial configuration and ending by a non-ε-transition. Indeed, if the system is infinitely branching, we have a run R like in the claim. Because $\mathsf{ctype}_{\mathcal{X}}(R(x)) = \mathsf{ctype}_{\mathcal{X}}(R(y))$, we can again produce a pumping run S_1 from $R(y)$ such that $\mathsf{ctype}_{\mathcal{X}}(S_1(0)) \sqsubseteq \mathsf{ctype}_{\mathcal{X}}(S_1(|S_1|))$ (by Theorem 3.1), and then again a pumping run S_2 from $S_1(|S_1|)$. This way we can produce arbitrarily many pumping runs (as many as required in $\mathcal{V}$); let S_m be the last of them. Then we have $\mathsf{ctype}_{\mathcal{X}}(R(y)) \sqsubseteq \mathsf{ctype}_{\mathcal{X}}(S_m(|S_m|))$ and by Proposition F.2 there is a top^0-non-erasing run from $S_m(|S_m|)$ ending in the same state as R, thus ending by a non-ε-transition. The composition of all these runs is in $\mathcal{V}$. Oppositely, assume that we have a run R in $\mathcal{V}$. Because the number of the factors $\mathcal{P}_{<,\varepsilon}$ in the definition of $\mathcal{V}$ is greater than the number of possible values of $\mathsf{ctype}_{\mathcal{X}}$, we can find two indices $x < y$ in R between these factors such that $\mathsf{ctype}_{\mathcal{X}}(R(x)) = \mathsf{ctype}_{\mathcal{X}}(R(y))$. Because $\mathcal{P}_{<,\varepsilon} \circ \mathcal{P}_{<,\varepsilon} \subseteq \mathcal{P}_{<,\varepsilon}$, we see that $R\restriction_{x,y} \in \mathcal{P}_{<,\varepsilon}$. Similarly, because $\mathcal{P}_{<,\varepsilon} \circ \mathcal{N}_0 \subseteq \mathcal{N}_0$, we see that $R\restriction_{y,|R|} \in \mathcal{N}_0$. Thus, we can use the claim and obtain that $\mathcal{G}$ is infinitely branching.

Now the algorithm checking emptiness is very easy: it is enough to check if from the initial configuration there is a run in $\mathcal{V}$ ending by a non-ε-transition. To do that, we compute $\mathsf{type}_{\mathcal{Y}}$ of the initial stack, and we check whether it contains a triple $(q_I, \mathcal{V}, q)$, where q is a state such that all transitions leading to state q are non-ε-transitions. Notice that the number of possible values of $\mathsf{ctype}_{\mathcal{X}}$ is n-times exponential in the size of the system. Thus, also the size of the family $\mathcal{Y}$ is n-times exponential (beside of the whole composition $\mathcal{V}$ it contains also all shorter compositions). Thus, the number of run descriptors for the family $\mathcal{Y}$ is $2n$-times exponential in the size of the system. It follows that the algorithm is in $2n$-EXPTIME.

As a corollary we obtain an algorithm checking whether the ε-contraction of the configuration graph of a given CPS $\mathcal{S}$ is finite. In order to decide this, we convert $\mathcal{S}$ into another system $\mathcal{R}$ such that the ε-contraction of the graph of $\mathcal{S}$ is finite if and only if the ε-contraction of the graph of $\mathcal{R}$ is finitely branching. We again assume that for each state q of $\mathcal{S}$ transitions leading to state q are all ε-transitions or are all non-ε-transitions. In $\mathcal{R}$ we have the same transitions as in $\mathcal{S}$, but all labelled by ε. Additionally, we add a new initial state and a transition labelled different from ε to the old initial state which preserves the stack. Moreover, from each state q such that all transitions leading to state q in $\mathcal{S}$ are not ε-transitions, in $\mathcal{R}$ we make a transition to a new state q_{die} labelled by some letter (there are no transitions from state q_{die}). After this conversion, the whole graph of $\mathcal{S}$ "lives" in the ε-transitions following the initial configuration of $\mathcal{S}$ but every node of the ε-contraction of the graph of $\mathcal{S}$ induces an edge from this initial configuration in the ε-contraction of the graph of $\mathcal{R}$.

Moreover, we also obtain an algorithm checking whether the unfolding into a tree of the ε-contraction of the configuration graph of a given CPS $\mathcal{S}$ is finite. Indeed, a tree is finite if it is finitely branching (which we check as above), and if it does not contain infinite paths. By Theorem 1.1 this tree contains infinite paths if and only if it contains a path (from the initial configuration) of length at least $\exp_{n-1}(C_{\mathcal{S}})$. A run containing at least $\exp_{n-1}(C_{\mathcal{S}})$ non-ε-transitions can be easily defined using wf-rules, thus we can check whether such run exists from the initial configuration by calculating the type of the initial configuration. (Whether the tree contains infinite paths can be also easily expressed in μ-calculus, hence decided using the μ-calculus decidability).

References

A1. C. H. Broadbent, A. Carayol, C.-H. L. Ong, and O. Serre. Recursion schemes and logical reflection. In *LICS*, pages 120–129. IEEE Computer Society, 2010.