On non-structural subtype entailment

Aleksy Schubert
alx@mimuw.edu.pl
Faculty of Mathematics, Informatics and Mechanics,
The University of Warsaw,
ul. Banacha 2,
02–097 Warsaw,
Poland

September 7, 2021

Abstract

We prove that the non-structural subtype entailment problem for finite and regular type expressions is in PSPACE. In this way we close a decidability and complexity gap pending since 1996.

1 Introduction

Application of a function or a procedure to an argument is a source of constraint that the argument must be able to handle (e.g. have fields, methods and other features) at least as much as the function requires. This is the main ground reason to introduce the subtyping relation to type systems in programming languages [Car84, Car88, Mit84, CD78], especially higher-order ones.

One natural way to consider this relationship is to introduce some primary relation on base types and compare type expressions of common structure built with help of type constructors and differing only in the distribution of base types. This approach leads to structural subtyping [Reh98]. In the non-structural form of subtyping, the system is augmented with additional type constructs, in the typical case with $\perp$ and $\top$ base types that represent the least and the greatest types respectively. The most apparent rationale to introduce $\top$ type nowadays is the need to include type constructs enabling gradual typing techniques [Tha88, Wan84, ST06, ST07, CLPS19]. We can assign the type $\top$ to a piece of code to enable its free use in the typed code with hope that means outside the static type system (e.g. dynamic checks, defensive implementation style, careful code inspection etc.) ensure the correctness of the program. Less obvious is the need for the $\perp$ type expression (see the discussion of Wand, O'Keefe and Palsberg [WOP95] concerning the type system of Thatte [Tha88]). Its presence enables the possibility to express an important intent of a programmer that the value of the type is not inspected or modified by the function, it is only moved to one place or another in processed data structures (this is the gist of the example analysed in [WOP95]).

Other uses of type expressions $\perp$ and $\top$ include the possibility to work with heterogeneous containers such as lists and persistent data [Tha88]. It can help in various analyses, e.g. binding-time analysis, analysis of type errors [Gom90], strictness analysis [Jen91] etc.

The mentioned above scenarios leads at first to the *subtype satisfiability* problem, in which one generates constraints based on function applications as well as on predefined operations and asks if the obtained set of subtyping constraints can be satisfied by some substitution. The existence of such a substitution implies that a program can be typed and safely executed (i.e. without referencing values of wrong types). This approach was used in many works on typechecking and type inference [AC91, AC93, ESTZ94, FM90, Mit91, Pot96, Pot98a]. There are algorithms which can decide satisfiability in cubic time [KPS94, PWO97] and in some specific cases even in quadratic [KPS95].

The dual *subtype entailment problem* may be viewed as a form of contextual validity. In this problem one asks if a given constraint $\sigma \leq \tau$ is valid in all situations when its context determined by a set of constraints E is satisfied, i.e. if each substitution S that makes all the constraints in E satisfied makes also $\sigma \leq \tau$ satisfied. In this view the subtype entailment problem can be seen as interesting in its own right. The more so as it does not reduce in the standard way to the satisfiability problem; here the main obstacle is the fact that the language of constraints does not include negation. Consequently, this form of validity requires a separate algorithm. Except from purely theoretic interest in this problem, it has also practical applications. In order to effectively present constraints to programmers or manipulate them programatically in an effective way one needs to minimise sets of constraints. This task to be complete requires solving type entailment problem [Pot01]. Type entailment is important in one more practical programming scenario. When a programmer tries to reimplement some library function she must be careful not to break existing contracts to preserve backwards compatibility. This calls for checks if the constraints generated by the new implementation are solved by all substitutions to the set of constraints obtained from the old version of the library, i.e. in all situations the old library could potentially be used.

In case of structural subtyping, the subtype entailment problem was shown by Henglein and Rehof to be co-NP-complete when only finite type expressions are possible [HR97] and PSPACE-complete in case of recursive type expressions [HR98]. The situation of the non-structural subtype entailment turned out to be more difficult. We know due to the work of Henglein and Rehof [HR98] that the problem is PSPACE-hard. Two ways were proposed to tackle the decidability of the problem. One way consisted in using tuple automata with equality constraints. Unfortunately, the emptiness problem for these automata is undecidable [Tre00] and these automata gave only insights that the generalisation of subtype entailment constraint to full first-order logic formalism results in an undecidable system [SAN$^+$02]. Another attempt reduced the subtype entailment problem to the problem of universality for a generalised version of finite automata, so called *cap automata* [NP03]. Although this approach provided a very nice abstraction of the subtype entailment problem, the task to devise algorithm that checks the universality turned out to be too difficult to complete so far.

In this paper we start with slightly refined constraint decomposition borrowed from Pottier [Pot98b] in Section 3. Next we introduce a highly structural, but infinitary view on the sets of constraints in Section 4. This turns out to be crucial in definition of manipulations on substitutions introduced in Section 5. With help of these manipulations it is possible to prove the main pumping lemma and reduce the size of solutions so that they can be effectively determined in Section 6, where the main result of the paper is presented. We conclude with final remarks in Section 7.

2 Preliminaries

We lay down here the notation for all standard notions used throughout the paper. One can skip this until technical material is introduced and use it as a reference.

The set $\mathcal{T}$ is the set of type expressions built using the grammar

$$\sigma, \tau ::= \bot \mid \top \mid (\sigma \circledcirc \sigma)$$

The main symbol of an expression $\bot$ is $\bot$, of $\top$ is $\top$, and of $\sigma_1 \circledcirc \sigma_2$ is $\circledcirc$. The main symbol of an expression σ is written $\blacktriangle\sigma$. We tacitly identify the expressions with binary trees the nodes of which are labelled with the symbols $\{\bot, \top, \circledcirc\}$. The size $|\sigma|$ of the expression σ is defined as $|\sigma| = 1$ for $\sigma \in \{\bot, \top\}$ and $|\sigma_0 \circledcirc \sigma_1| = |\sigma_0| + |\sigma_1| + 1$. The size $|A|$ of a finite set A is the number of its elements. The depth $\|\sigma\|$ of an expression σ is defined as $\|\bot\| = \|\top\| = 0$

and $\|\sigma_0 \circledcirc \sigma_1\| = \max\{\|\sigma_0\|, \|\sigma_1\|\} + 1$. The set $\mathcal{T}_n = \{\sigma \mid \|\sigma\| = n\}$. We also consider the set $\mathcal{T}^{\text{reg}}$ of regular type expressions, built of the above mentioned symbols. These are trees the internal nodes of which are labelled with $\circledcirc$ and the leaves are labelled with $\top, \bot$. In addition the trees satisfy the regularity condition which says that they have finitely many distinct subtrees. Although the formal treatment of such expressions requires introduction of fixed point operators (in the style of the μ operator used by Amadio and Cardelli [AC93]) we do not do this since this would introduce additional notational burden and it should be clear how to extend the constructions to such expressions. We often refer in this paper to infinite expressions. In all the cases we mean infinite, but regular ones.

We define inequalities on variable-free expressions using the axioms: $\sigma \leq_0 \tau, \bot \leq_n \top, \sigma \circledcirc \tau \leq_n \top$, $\tau \leq_n \tau$ for all σ, τ. In addition, whenever $\sigma_1 \leq_{n-1} \tau_1$ and $\sigma_2 \leq_{n-1} \tau_2$ hold we have also $\sigma_1 \circledcirc \sigma_2 \leq_n \tau_1 \circledcirc \tau_2$. When the operator $\circledcirc$ is in its first arguments contravariant and in its second one covariant (in short, contravariant) then whenever $\sigma_1 \leq_{n-1} \tau_1$ and $\sigma_2 \leq_{n-1} \tau_2$ hold we have also $\tau_1 \circledcirc \sigma_2 \leq_n \sigma_1 \circledcirc \tau_2$. These inequalities define by induction partial orders $\leq_n$ for $n \in \mathbb{N}$ on $\mathcal{T}$. We can now define by coinduction partial orders $\leq$ on $\mathcal{T}$ and $\mathcal{T}^{\text{reg}}$ so that $\sigma \leq \tau$ when $\sigma \leq_n \tau$ for all $n \in \mathbb{N}$.

To express various symmetries in the constructions we need to decorate our notation with subscripts so that $\sigma_1 \leq^0 \sigma_2$ and $\sigma_2 \leq^1 \sigma_1$ are both equivalent to $\sigma_1 \leq \sigma_2$. Moreover, $\bot^0 = \bot$ and $\bot^1 = \top$ as well as $\top^0 = \top$ and $\top^1 = \bot$.

Let $\mathcal{V}$ be the set of type variables, written α, β etc. and $\mathcal{V}^{\bot} = \mathcal{V} \cup \{\bot, \top\}$. In addition to the variable-free expressions, we consider the set $\mathcal{T}_{\mathcal{V}}$ of type expressions with variables

$$\sigma, \tau ::= \bot \mid \top \mid \alpha \mid (\sigma \circledcirc \sigma)$$

where $\alpha \in \mathcal{V}$. For an expression σ, we note by $\mathrm{FV}(\sigma)$ the set of variables that occur in σ. We also write $\mathrm{FV}(A)$ for the set of variables that occur in all expressions in the set A. A type expression σ is atomic when $\sigma \in \mathcal{V}^{\bot}$. The notions and notations for size and depth are extended accordingly to expressions with variables. We also consider the set $\mathcal{T}_{\mathcal{V}}^{\text{reg}}$ of regular type expressions built of the symbols $\{\bot, \top, \circledcirc\}$ and variables of $\mathcal{V}$.

A substitution S is a finite partial function in $\mathcal{V} \rightharpoonup_{\text{fin}} \mathcal{T}_{\mathcal{V}}$ (or $\mathcal{V} \rightharpoonup_{\text{fin}} \mathcal{T}_{\mathcal{V}}^{\text{reg}}$). We write $\mathrm{dom}(S)$ for the domain of S. We write $S[\alpha \leftarrow \tau]$ for a substitution S' such that $S'(\beta) = S(\beta)$ for $\beta \neq \alpha$ and $S'(\alpha) = \tau$. In case $S(\tau)$ has no variables for all $\tau \in \mathrm{dom}(S)$, we say that S is *ground*. Except when stated explicite otherwise, the substitutions considered in this paper are ground. A substitution S is extended to type expressions so that $S(\bot) = \bot$, $S(\top) = \top$, $S(\alpha) = \alpha$ when $\alpha \notin \mathrm{dom}(S)$, $S(\alpha) = S(\alpha)$ when $\alpha \in \mathrm{dom}(S)$ and $S(\sigma_1 \circledcirc \sigma_2) = S(\sigma_1) \circledcirc S(\sigma_2)$.

In addition to inequalities, we consider *inequations* of the form $\sigma \leq \tau$, where $\sigma, \tau \in \mathcal{T}_{\mathcal{V}}$. Given a ground substitution S we can define $S \models \sigma \leq \tau$ to hold when $S(\sigma) \leq S(\tau)$. In that case, we say that S *solves* $\sigma \leq \tau$. This relation naturally extends to a set E of inequations and $S \models E$ holds when $S \models \sigma \leq \tau$ for each $\sigma \leq \tau \in E$. We say that S *solves* E when $S \models E$ and that E *is solvable* when there is S such that $S \models E$. At last we write $\mathrm{FV}(E)$ for the set of variables that occur in all inequations in E. The *size* of an inequation $\sigma \leq \tau$ is the sum $|\sigma| + |\tau|$. We define $|E|_{\circledcirc} = \sum_{\sigma \leq \tau \in E} |\sigma| + |\tau|$. As with inequalities, we write $\leq^0$ for $\leq$ and $\leq^1$ for $\geq$.

We use words over $\{0, 1\}$ to refer to *positions* in type expressions from $\mathcal{T}$ or $\mathcal{T}^{\text{reg}}$ and write $\mathfrak{g} \cdot \mathfrak{g}'$ for concatenation of two words $\mathfrak{g}, \mathfrak{g}' \in \{0, 1\}^*$. If $\mathfrak{g} = i_0 \cdots i_{n-1}$ where $i_0, \ldots, i_{n-1} \in \{0, 1\}$ we write $\mathfrak{g}(k)$ for i_k. If $\mathfrak{h} = \mathfrak{g} \cdot \mathfrak{g}'$ then we write $\mathfrak{h} \swarrow \mathfrak{g}$ for $\mathfrak{g}'$ and $\mathfrak{h} \nearrow \mathfrak{g}'$ for $\mathfrak{g}$. The length n of the position $\mathfrak{g}$ is written $|\mathfrak{g}|$. Given a type expression σ we define $\sigma|_{\varepsilon} = \sigma$ for the empty word ε and $\sigma|_{i \cdot \mathfrak{g}} = \sigma_i|_{\mathfrak{g}}$ for $i \in \{0, 1\}$ when $\sigma = \sigma_0 \circledcirc \sigma_1$. The reference operator $|$ binds stronger than $\circledcirc$ so that $\sigma_0 \circledcirc \sigma_1|_{\mathfrak{g}} = \sigma_0 \circledcirc (\sigma_1|_{\mathfrak{g}})$. If $\sigma|_{\mathfrak{g}} = \tau$ then the main symbol $\blacktriangle\tau$ of τ is called the *label of* σ at $\mathfrak{g}$. The word ε is *accessible* in any σ, while $i \cdot \mathfrak{g}$ is *accessible* in σ when $\sigma = \sigma_0 \circledcirc \sigma_1$ and $\mathfrak{g}$ is accessible in σ_i. We write $\mathfrak{g} \preceq \mathfrak{g}'$ to express that $\mathfrak{g}$ is a prefix of $\mathfrak{g}'$ and use $\prec$ for the strict variant of $\preceq$.

Most of the material in the paper works for both covariant and contravariant binary symbol. The constructions in this paper are presented for the covariant symbol. However, we believe that they can be adapted to work for the contravariant one at the cost of additional variance bookkeeping.

We can now use these notions to make the following observations concerning inequalities.

Proposition 2.1 `(positions and inequalities)`

1. *If* $\sigma_1 \circledcirc \sigma_2 \not\leq \tau_1 \circledcirc \tau_2$ *then either* $\sigma_1 \not\leq \tau_1$ *or* $\sigma_2 \not\leq \tau_2$.
2. *If* $\sigma \not\leq \tau$ *then there is a maximal position* $\mathfrak{g}$ *such that* $\sigma|_{\mathfrak{g}} \not\leq \tau|_{\mathfrak{g}}$. *Moreover,* $\sigma|_{\mathfrak{g}} = \top$ *or* $\tau|_{\mathfrak{g}} = \bot$.

Proof:
The proof is left to the reader. □

Many of the constructions in the paper rely on expression manipulations. To express a type expression modified at a particular position $\mathfrak{g}$ we write $\sigma[\mathfrak{g} \leftarrow \tau]$. This is defined inductively as $\sigma[\varepsilon \leftarrow \tau] = \tau$ and $(\sigma_0 \circledcirc \sigma_1)[0 \cdot \mathfrak{g} \leftarrow \tau] = \sigma_0[\mathfrak{g} \leftarrow \tau] \circledcirc \sigma_1$, $(\sigma_0 \circledcirc \sigma_1)[1 \cdot \mathfrak{g} \leftarrow \tau] = \sigma_0 \circledcirc \sigma_1[\mathfrak{g} \leftarrow \tau]$ and $\bot^j[i \cdot \mathfrak{g} \leftarrow \tau] = \bot^j$ for $j \in \{0, 1\}$. We extend this notation to sets of pairwise incomparable words so that for a set $A = \{\mathfrak{g}_1, \ldots, \mathfrak{g}_n\}$ of pairwise incomparable words we let $\sigma[A \leftarrow \tau] = \sigma[\mathfrak{g}_1 \leftarrow \tau] \cdots [\mathfrak{g}_n \leftarrow \tau]$.

3 Sets of inequations in the entailment problem

The main focus in this paper is on the non-structural type entailment problem defined as follows.

Definition 1 `(entailment)`
Input: A finite set $E = \{\sigma_1 \leq \tau_1, \ldots, \sigma_n \leq \tau_n\}$ of inequations between type expressions together with an inequation $\sigma \leq \tau$.
Question: Does $S \models \sigma \leq \tau$ hold for all substitutions S such that $S \models E$?

This problem is studied in two variants. In the first one type expressions are finite while in the second one infinite (but regular) trees. Henglein and Rehof studied the non-structural subtype entailment and proved it is PSPACE-hard in both variants [HR97].

A natural counterpart of the entailment problem is the satisfiability problem.

Definition 2 `(satisfiability problem)`
Input: A finite set $E = \{\sigma_1 \leq \tau_1, \ldots, \sigma_n \leq \tau_n\}$ of inequations between type expressions.
Question: Is there a substitution S such that $S \models E$ holds?

This problem arose in the context of a small programming language with infinite regular types by Palsberg and O'Keefe [PO95a, PO95b] and by Pottier [Pot96]. The case of finite trees was solved by Palsberg et al. [PWO97].

Theorem 3.1 `(satisfiability problem)`
The non-structural type expressions satisfiability problem for finite types is in PTIME.

It is usually easier to understand problems solutions of which can be formulated in terms of finding a particular situation. The entailment problem is not formulated in this way so we turn our attention to another problem that has this desired characteristics and solution of which determines the solution of the original problem.

Definition 3 `(type inentailment problem)`
Input: A finite set $E = \{\sigma_1 \leq \tau_1, \ldots, \sigma_n \leq \tau_n\}$ of inequations between type expressions together with an inequation $\sigma \leq \tau$.
Question: Is there a substitution S such that $S \models E$ and $S \not\models \sigma \leq \tau$?

Note that in case $S \not\models \sigma \leq \tau$ there is a maximal position $\mathfrak{g}$, called *critical position* such that $S(\sigma)|_{\mathfrak{g}} \not\leq S(\tau)|_{\mathfrak{g}}$, see Prop. 2.1(2). The constructions in this paper aim at finding a solution S to E with such a critical position the length of which can be effectively estimated. We immediately see that the instances of both problems are the same and the answer to the inentailment problem is positive if and only if the answer to the entailment problem is negative. Therefore if the inentailment problem is in PSPACE then entailment problem also is.

Example 1 *Throughout this paper we work on an example of subtype entailment instance* $E, \alpha \leq \beta$, *where $\circledcirc$ is a covariant symbol and*

$$E = \{\alpha_{010} \circledcirc \alpha_{011} \leq \alpha_{01}, \quad (\alpha_{00} \circledcirc \alpha_{01}) \circledcirc \alpha_1 \leq \alpha, \\ \beta \leq \alpha_{01}, \quad \beta_{01} \leq \alpha_{010}, \quad \beta_0 \leq \beta_{00} \circledcirc \beta_{01}, \\ \beta_{01} \leq \beta_{010} \circledcirc \beta_{011}\}$$

Consider a substitution S defined as

$$\begin{array}{l} S(\alpha_{010}) = (\top \circledcirc (\top \circledcirc \top)) \circledcirc (\top \circledcirc \top), \\ S(\alpha_{00}) = S(\alpha_1) = \bot \circledcirc \bot, \\ S(\alpha_{01}) = S(\alpha_{010}) \circledcirc S(\alpha_{011}), \quad S(\alpha_{011}) = \top \circledcirc \top, \\ S(\alpha) = (S(\alpha_{00}) \circledcirc S(\alpha_{01})) \circledcirc (\bot \circledcirc \bot), \\ S(\beta) = (S(\alpha_{00}) \circledcirc (S(\alpha_{01})[0010 \leftarrow \bot \circledcirc \bot])) \circledcirc (\bot \circledcirc \bot), \\ S(\beta_0) = S(\beta_{00}) \circledcirc S(\beta_{01}), \quad S(\beta_1) = S(\alpha_1), \\ S(\beta_{00}) = S(\alpha_{00}), \quad S(\beta_{01}) = S(\beta_{010}) \circledcirc S(\beta_{011}), \\ S(\beta_{010}) = S(\alpha_{010}), \quad S(\beta_{011}) = S(\alpha_{011}). \end{array}$$

Note that $S \models E$, but $S \not\models \alpha \leq \beta$ with critical position $\mathfrak{g} = 010010$ such that $S(\alpha)|_{\mathfrak{g}} = \top \not\leq \bot \circledcirc \bot = S(\beta)|_{\mathfrak{g}}$.

To make constructions in this paper manageable we have to work with sets of inequations in a specific form, which is presented hereafter along with demonstration that restricting to this form is not essential for solvability.

3.1 Sets of small inequations

Many constructions of the current paper rely on finitary labellings of infinite binary trees presented in Section 4 that characterise solutions of inequations. To make their definition simpler we introduce a interesting form of inequations borrowed from Pottier [Pot96, Pot01].

Definition 4 `(set of small inequations)`
We say that an equation is *small* when it has one of the forms:

- $\sigma_0 \leq \sigma_1$ where σ_i is atomic for $i \in \{0, 1\}$,
- $\sigma_0 \leq^j \sigma_1 \circledcirc \sigma_2$ where σ_i is atomic for $i \in \{0, 1, 2\}$ and $j \in \{0, 1\}$.

A set of inequations E is a set of *small* inequations when each its element is small.

The following proposition states that it is enough to restrict entailment instances to such inequations.

Proposition 3.2 `(small are enough, [Pot01])`
For each set of inequations E and inequation $\sigma \leq \tau$ there is an (effectively computable) set of small inequations E' and inequation $\alpha_0 \leq \alpha_1$ such that $E \models \sigma \leq \tau$ is solvable if and only if $E' \models \alpha_0 \leq \alpha_1$ is.

It is necessary to mention here that Pottier uses slightly less general statements in which the inequations generated from the original type inference problem are generated so that they are small. However, the more general statement presented above is an easy folklore result.

Example 2 *We can turn E from Example 1 to the following set of small inequations*

$$E_{\text{small}} = \{\, \alpha_{010} \circledcirc \alpha_{011} \leq \alpha_{01}, \quad \underline{\alpha_0 \circledcirc \alpha_1 \leq \alpha}, \\ \underline{\alpha_{00} \circledcirc \alpha_{01} \leq \alpha_0}, \quad \underline{\alpha_0 \leq \alpha_{00} \circledcirc \alpha_{01}}, \\ \beta \leq \alpha_{01}, \quad \beta_{01} \leq \alpha_{010}, \quad \beta_0 \leq \beta_{00} \circledcirc \beta_{01}, \\ \beta_{01} \leq \beta_{010} \circledcirc \beta_{011} \,\}.$$

The new inequations are underlined above. Note that one inequation from E is missing here.

3.2 Saturated sets of inequations

Given a set of inequations E, we can infer in a natural way some other new inequations that are immediate consequences of those present in E. The main advantage of this motion is that later in the paper it is easier to refer to elements of a set which has tangible and effectively computable elements instead of relying on filters of size that is not immediate to determine.

Definition 5 `(saturated set of inequations)`
A set of small inequations E is *saturated* if the conditions below are met, we assume here that $\sigma_0, \sigma_1, \sigma_2, \sigma_3, \sigma_4 \in \mathcal{V}^\perp$

1. for each $\alpha \in \mathrm{FV}(E)$ we have $\alpha \leq \alpha \in E$,
2. $\bot \leq \top \in E$,
3. if $\sigma_0 \leq \sigma_1 \in E$ and $\sigma_1 \leq \sigma_2 \in E$ as well as σ_0, σ_2 are atomic then $\sigma_0 \leq \sigma_2 \in E$,
4. if $\sigma_0 \circledcirc \sigma_1 \leq \sigma_2 \in E$ and $\sigma_2 \leq \sigma_3 \circledcirc \sigma_4 \in E$ where $\sigma_2 \in \mathrm{FV}(E)$ then $\sigma_0 \leq \sigma_3 \in E$ and $\sigma_1 \leq \sigma_4 \in E$,

5. if $\sigma_0 \leq^i \sigma_1 \circledcirc \sigma_2 \in E$ and $\sigma_3 \leq^i \sigma_0 \in E$ then $\sigma_3 \leq^i \sigma_1 \circledcirc \sigma_2 \in E$,
6. if $\sigma_0 \circledcirc \sigma_1 \leq^i \sigma_2 \in E$ and $\sigma'_j \leq^i \sigma_j \in E$ for $j \in \{0,1\}$ then $\sigma'_0 \circledcirc \sigma'_1 \leq^i \sigma_2 \in E$.

This notion is similar to the notion of the *closed set of constraints* used by Pottier [Pot01] and inspired by Eifrig, Smith and Trifonov [ESTZ94]. However, there are three subtle differences that make some of the constructions further in the paper easier. First, we allow σ_1 in (3) to be any expression, not only variable. Second, closed sets are not closed on the condition (6). Third, we do not forbid contradictory inequations such as $\top \leq \bot$. We later use (6) to obtain nice properties of one of the decorations defined later in this paper.

Example 3 *We continue Example 2. We can obtain an equivalent saturated set of inequations*

$$
\begin{aligned}
E_{\text{sat}} = E_{\text{small}} \cup \{ & \bot \leq \top, \quad \alpha \leq \alpha, \quad \beta \leq \beta, \\
& \alpha_0 \leq \alpha_0, \quad \alpha_1 \leq \alpha_1, \quad \alpha_{00} \leq \alpha_{00}, \\
& \alpha_{01} \leq \alpha_{01}, \quad \alpha_{010} \leq \alpha_{010}, \quad \alpha_{011} \leq \alpha_{011}, \\
& \beta_0 \leq \beta_0, \quad \beta_{00} \leq \beta_{00}, \quad \beta_{01} \leq \beta_{01}, \\
& \beta_{010} \leq \beta_{010}, \quad \beta_{011} \leq \beta_{011} \quad \alpha_{00} \circledcirc \beta \leq \alpha_0, \\
& \beta_{01} \circledcirc \alpha_{011} \leq \alpha_{01}, \quad \beta_0 \leq \beta_{00} \circledcirc \alpha_{010} \ \}
\end{aligned}
$$

As we can see, saturation consists in adding inequations that can be syntactically inferred with help of transitivity and congruence.

Theorem 3.3 `(saturated sets are enough)`
For each set of inequations E there is an (effectively computable) set of saturated inequations E' such that E is solvable if and only if E' is. Moreover $\mathrm{FV}(E) = \mathrm{FV}(E')$.

Proof:
It is easy to see that for each condition in Def. 5 if the inequations in its assumption are solved by S then also the inequations in its conclusion are. So E extended with the latter is solvable if and only if the original set of inequations is. Also such an extension of E does not change the number of variables.

We observe now that the number of type expressions that can occur on sides of inequation is bounded by $n = |\mathrm{FV}(E)| + |\{\bot, \top\}| + (|\mathrm{FV}(E)| + |\{\bot, \top\}|)^2$. Since the number of variables in the set of inequations does not change with the extension steps as above, the number of non-trivial extension steps is bounded by n^2, i.e. each non-trivial step brings in new inequations the sides of which come from a fixed set of size n. □

Assumption From now on we assume that E is a saturated set of small inequations.

3.3 Solutions that avoid constants

Later steps become significantly simpler when we are able to impose an additional assumption on solutions. We say that a ground solution S *avoids constants* when $S \models E$ and in addition for each $\alpha \in \mathrm{FV}(E)$ it holds that $S(\alpha) \notin \{\bot, \top\}$.

As stated in proposition below this restriction on solution does not have significant impact neither on decidability nor on the size of the final solution.

Proposition 3.4 `(solutions that avoid constants)`
There is a family $\mathcal{E}$ of sets of inequations such that E is solvable if and only if for some element $E' \in \mathcal{E}$ there is S' that avoids constants and solves E'.

Moreover, for each $E' \in \mathcal{E}$ we have $|E'|_\circledcirc \leq |E|_\circledcirc$ and if some S' solves $E' \in \mathcal{E}$ and $S' \not\models \alpha_0 \leq \alpha_1$ then there is a solution S of E such that $S \not\models \alpha_0 \leq \alpha_1$.

Proof:
We let $\mathcal{E}$ be the set of elements E' each of which is obtained as $E' = T(E)$ where T is a substitution such that $\mathrm{dom}(T) \subseteq \mathrm{FV}(E)$ and for variables in $\mathrm{dom}(T)$ it assigns either $\bot$ or $\top$.

It is easy to see that the conclusions of the proposition follow for such $\mathcal{E}$. The details are left to the reader. □

The considerations in the section above do not rely on the fact that the solutions substitute finite or infinite type expressions and are valid in both cases.

4 Static decoration

The variables that occur in a set of inequations E give rise to finitary information that helps in constructing and rearranging solutions. This information is conveyed by means of a static decoration in an infinite binary tree. The decoration uses pairs of sets that contain atomic symbols which occur in E. These atomic symbols suffice for our purpose since the set E is a saturated set of small inequations. In particular, these decorations give a criterion on when solutions cannot be finite, which is shown in the proof of Prop. 4.4.

Definition 6 (static decoration)
Let $t = \{0,1\}^*$ be an infinite full binary tree. We assign for each $\alpha \in \mathrm{FV}(E)$ a *static decoration* of a tree node $\mathfrak{g} \in t$ which is a pair $\langle \|\alpha|_{\mathfrak{g}}\|^0, \|\alpha|_{\mathfrak{g}}\|^1 \rangle$ defined inductively as follows for both $i \in \{0,1\}$

1. $\|\alpha|_{\varepsilon}\|^i = \{\sigma \in \mathcal{V}^{\perp} \mid \sigma \leq^i \alpha \in E\}$,
2. $\|\alpha|_{\mathfrak{g}\cdot j}\|^i = \{\sigma_j \in \mathcal{V}^{\perp} \mid \sigma_0 \circledcirc \sigma_1 \leq^i \beta \in E$ and $\beta \in \|\alpha|_{\mathfrak{g}}\|^i\}$.

We extend the notation $\|\alpha|_{\mathfrak{g}}\|^i$ to type expressions so that $\|\sigma_0 \circledcirc \sigma_1|_{j\cdot\mathfrak{g}}\|^i$ for $i, j \in \{0,1\}$ is equal to $\|\sigma_j|_{\mathfrak{g}}\|^i$.

Note that $\alpha \in \|\alpha|_{\varepsilon}\|^i$. Moreover, it is worth mentioning here that the point (6) in Def. 5 makes it possible to refer to inequations instead of some inferred sets of constraints.

The static decoration $\|\alpha|_{\mathfrak{g}}\|^0$ is a set of symbols such that when a solution S is applied to them they are lower bounds for the expression at $\mathfrak{g}$ in the result of applying S to α. Similarly, $\|\alpha|_{\mathfrak{g}}\|^1$ gathers symbols such that when a solution S is applied they are upper bounds for the expression at $\mathfrak{g}$ in $S(\alpha)$. This is precisely formulated in Def. 7 below. In essence, this static decoration may be viewed as a slightly more structured version of constraint graphs used by Pottier [Pot98a, Pot98b].

Example 4 *We continue Example 3. It is easy to see that* $\|\alpha|_{\varepsilon}\|^i = \{\alpha\}$ *for* $i = 0, 1$. *Interestingly enough* $\|\alpha|_{010}\|^0 = \{\beta_{01}, \alpha_{010}\}$ *and* $\|\beta_0|_1\|^1 = \{\beta_{01}, \alpha_{010}\}$. *Observe that this implies that presence of* $\perp$ *on* $S(\alpha)|_{010\cdot\mathfrak{g}}$ *for some solution* S *of* E *and position* $\mathfrak{g}$ *forces presence of* $\perp$ *on* $S(\beta_0)|_{1\cdot\mathfrak{g}}$, *provided that* $1\cdot\mathfrak{g}$ *is accessible in* $S(\beta_0)$.

We need to develop the basic properties of static decorations so that we do not need to refer to its definition too frequently.

Proposition 4.1 (basic properties of static decorations)
Assume that $i \in \{0,1\}$.

1. *For any variables* $\alpha, \beta \in \mathcal{V}$ *and position* $\mathfrak{g}$, *if* $\beta \in \|\alpha|_{\mathfrak{g}}\|^i$ *then* $\|\beta|_{\varepsilon}\|^i \subseteq \|\alpha|_{\mathfrak{g}}\|^i$.
2. *For any variables* $\alpha, \beta \in \mathcal{V}$ *and positions* $\mathfrak{g}, \mathfrak{g}'$, *if* $\beta \in \|\alpha|_{\mathfrak{g}}\|^i$ *then* $\|\beta|_{\mathfrak{g}'}\|^i \subseteq \|\alpha|_{\mathfrak{g}\cdot\mathfrak{g}'}\|^i$.
3. *For any variables* $\alpha, \beta \in \mathcal{V}$ *and positions* $\mathfrak{g}, \mathfrak{g}', \mathfrak{h}$, *if* $\|\beta|_{\mathfrak{g}'}\|^i \subseteq \|\alpha|_{\mathfrak{g}}\|^i$ *then* $\|\beta|_{\mathfrak{g}'\cdot\mathfrak{h}}\|^i \subseteq \|\alpha|_{\mathfrak{g}\cdot\mathfrak{h}}\|^i$.
4. *For any symbols* $\alpha, \beta, \gamma \in \mathcal{V}^{\perp}$ *and position* $\mathfrak{g}$ *if* $\beta \in \|\alpha|_{\mathfrak{g}}\|^i$ *and* $\gamma \leq^i \beta \in E$ *then for each* β' *such that* $\beta' \leq^i \gamma \in E$ *we have* $\beta' \in \|\alpha|_{\mathfrak{g}}\|^i$.
5. *For any symbols* $\alpha, \beta, \gamma \in \mathcal{V}^{\perp}$ *and position* $\mathfrak{g}$ *if* $\beta \in \|\alpha|_{\mathfrak{g}}\|^i$ *and* $\gamma \in \|\alpha|_{\mathfrak{g}}\|^{1-i}$ *then* $\beta \leq^i \gamma \in E$.
6. *For any variables* $\alpha, \beta, \gamma \in \mathcal{V}$ *and positions* $\mathfrak{g}, \mathfrak{g}'$, *if* $\alpha \in \|\beta|_{\mathfrak{g}}\|^i$ *and* $\alpha \in \|\gamma|_{\mathfrak{g}'}\|^{1-i}$ *then* $\|\gamma|_{\mathfrak{g}'}\|^i \subseteq \|\beta|_{\mathfrak{g}}\|^i$.
7. *For any symbols* $\alpha_0, \alpha_1, \beta, \gamma \in \mathcal{V}^{\perp}$ *and position* $\mathfrak{g}$ *if* $\beta \in \|\gamma|_{j\cdot\mathfrak{g}}\|^{1-i}$ *as well as* $\alpha_0 \circledcirc \alpha_1 \leq^i \gamma \in E$ *then* $\beta \in \|\alpha_j|_{\mathfrak{g}}\|^{1-i}$ *for* $j \in \{0,1\}$.
8. *For any variable* $\alpha \in \mathcal{V}^{\perp}$ *and position* $\mathfrak{g}$ *if* $\|\alpha|_{\mathfrak{g}}\|^i \subseteq \{\perp^j\}$ *for some* $j \in \{0,1\}$ *then for each position* $\mathfrak{g}'$ *we have* $\|\alpha|_{\mathfrak{g}\cdot\mathfrak{g}'}\|^i = \emptyset$.

Proof:
The proof of (1) is by case analysis depending on the form of $\mathfrak{g}$. When $\mathfrak{g}$ is ε, we take $\gamma \in \|\beta|_{\varepsilon}\|^i$. This implies that $\gamma \leq^i \beta \in E$.

We can conclude now from $\beta \in \|\alpha|_{\varepsilon}\|^i$ that $\beta \leq^i \alpha \in E$. By definition of the saturated set $\gamma \leq^i \alpha \in E$. Thus $\gamma \in \|\alpha|_{\varepsilon}\|^i$.

When $\mathfrak{g}$ is $\mathfrak{g}' \cdot j$, we again take $\gamma \in \|\beta|_\varepsilon\|^i$. This implies that $\gamma \leq^i \beta \in E$. As $\beta \in \|\alpha|_{\mathfrak{g}'\cdot j}\|^i$, there is an inequation $\gamma_0 \circledcirc \gamma_1 \leq^i \gamma_u \in E$ for some $\gamma_u \in \|\alpha|_{\mathfrak{g}'}\|^i$ and $\beta = \gamma_j$. Since E is saturated, there is an inequation $\gamma'_0 \circledcirc \gamma'_1 \leq^i \gamma_u \in E$ where $\gamma'_j = \gamma$ and $\gamma'_{1-j} = \gamma_{1-j}$ so by definition of the static decoration $\gamma \in \|\alpha|_{\mathfrak{g}'\cdot j}\|^i$.

The proof of (2) is by induction over $n = |\mathfrak{g}'|$. In case $|\mathfrak{g}'| = 0$, we obtain immediately that $\mathfrak{g}' = \varepsilon$, and the proof is by case (1) of the current proposition. In case $\mathfrak{g}' = \mathfrak{g}'' \cdot j$ we take $\gamma \in \|\beta|_{\mathfrak{g}'}\|^i$. This is possible by definition when $\gamma = \gamma_j$ for some $j \in \{0,1\}$ with $\gamma_0 \circledcirc \gamma_1 \leq^i \gamma_u \in E$ and $\gamma_u \in \|\beta|_{\mathfrak{g}''}\|^i$. We obtain by the induction hypothesis that $\gamma_u \in \|\alpha|_{\mathfrak{g}\cdot\mathfrak{g}''}\|^i$. Consequently, $\gamma = \gamma_j \in \|\alpha|_{\mathfrak{g}\cdot\mathfrak{g}''\cdot j}\|^i$ by definition of the static decoration, which completes the proof in this case.

The proof of (3) is by induction over $\mathfrak{h}$. If $\mathfrak{h} = \varepsilon$ then the conclusion follows by assumption. If $\mathfrak{h} = \mathfrak{h}_0 \cdot j$ for some $j \in \{0,1\}$ then we obtain $\|\beta|_{\mathfrak{g}'\cdot\mathfrak{h}_0}\|^i \subseteq \|\alpha|_{\mathfrak{g}\cdot\mathfrak{h}_0}\|^i$ by the induction hypothesis. Consider now any $\gamma \in \|\beta|_{\mathfrak{g}'\cdot\mathfrak{h}_0\cdot j}\|^i$. By definition there are $\gamma_u, \gamma_0, \gamma_1$ such that $\gamma_u \in \|\beta|_{\mathfrak{g}'\cdot\mathfrak{h}_0}\|^i$ and $\gamma_0 \circledcirc \gamma_1 \leq^i \gamma_u \in E$ and $\gamma_j = \gamma$. As $\gamma_u \in \|\beta|_{\mathfrak{g}'\cdot\mathfrak{h}_0}\|^i$, we also have $\gamma_u \in \|\alpha|_{\mathfrak{g}\cdot\mathfrak{h}_0}\|^i$. By case (2) of the current proposition, we obtain that $\|\gamma_u|_j\|^i \subseteq \|\alpha|_{\mathfrak{g}\cdot\mathfrak{h}_0\cdot j}\|^i$. As $\gamma = \gamma_1 \in \|\gamma_u|_j\|^i$, we conclude with $\gamma \in \|\alpha|_{\mathfrak{g}\cdot\mathfrak{h}_0\cdot j}\|^i$ and then that $\|\beta|_{\mathfrak{g}'\cdot\mathfrak{h}_0\cdot j}\|^i \subseteq \|\alpha|_{\mathfrak{g}\cdot\mathfrak{h}_0\cdot j}\|^i$.

The proof of (4) is by observation that $\beta' \leq^i \gamma \in E$ and $\gamma \leq^i \beta \in E$ imply by saturation of E that $\beta' \leq^i \beta \in E$. By definition of decoration it means that $\beta' \in \|\beta|_\varepsilon\|^i$. We can now use the case (1) of the current proposition and combine it with the assumption of the current one to conclude that $\beta' \in \|\alpha|_\mathfrak{g}\|^i$.

The proof of (5) is by induction over $\mathfrak{g}$. If $\mathfrak{g} = \varepsilon$ then by definition of $\|\alpha|_\varepsilon\|^i$ and $\|\alpha|_\varepsilon\|^{1-i}$ we obtain that $\beta \leq^i \alpha \in E$ and $\alpha \leq^i \gamma \in E$. As E is saturated, we obtain that $\beta \leq^i \gamma \in E$, which completes the proof in the base case.

In case $\mathfrak{g} = \mathfrak{g}_0 \cdot j$ for some $\mathfrak{g}_0, j$, we observe that there are some $\beta_0, \beta_1, \beta_u$ such that $\beta_0 \circledcirc \beta_1 \leq^i \beta_u \in E$ and $\beta_u \in \|\alpha|_{\mathfrak{g}_0}\|^i$ with $\beta_j = \beta$. Similarly, there are some $\gamma_0, \gamma_1, \gamma_u$ such that $\gamma_u \leq^i \gamma_0 \circledcirc \gamma_1 \in E$ and $\gamma_u \in \|\alpha|_{\mathfrak{g}_0}\|^{1-i}$ and $\gamma_j = \gamma$. By induction hypothesis we obtain that $\beta_u \leq \gamma_u \in E$. Since E is saturated, we obtain that $\beta_j \leq^i \gamma_j \in E$ and thus $\beta \leq^i \gamma \in E$, which completes the proof of the induction step.

The proof of (6) starts with an observation that for each $\alpha' \in \|\gamma|_{\mathfrak{g}'}\|^i$ we have $\alpha' \leq^i \alpha \in E$ by the case (5) of the current proposition. Thus by the case (4) of the current proposition we obtain that $\alpha' \in \|\beta|_\mathfrak{g}\|^i$.

The proof of (7) is by induction over $\mathfrak{g}$. If $\mathfrak{g} = \varepsilon$, we start with the assumption that $\beta \in \|\gamma|_j\|^{1-i}$. By definition of the static annotation there is an inequation $\beta_0 \circledcirc \beta_1 \leq^{1-i} \gamma \in E$ where $\beta = \beta_j$. As $\alpha_0 \circledcirc \alpha_1 \leq^i \gamma \in E$ and E is saturated, we obtain that $\alpha_j \leq^i \beta_j$. This means by definition that $\beta = \beta_j \in \|\alpha_j|_\varepsilon\|^{1-i}$, which is the desired conclusion.

If $\mathfrak{g} = \mathfrak{g}_0 \cdot k$ for some $k \in \{0,1\}$, we assume first that $\beta \in \|\gamma|_{j\cdot\mathfrak{g}_0\cdot k}\|^{1-i}$. By definition of the static decoration this means that there is an inequation $\beta_0 \circledcirc \beta_1 \leq^{1-i} \beta_u \in E$ such that $\beta_k = \beta$ and $\beta_u \in \|\gamma|_{j\cdot\mathfrak{g}_0}\|^{1-i}$. By the induction hypothesis $\beta_u \in \|\alpha_j|_{\mathfrak{g}_0}\|^{1-i}$ so by definition of the static decoration we obtain $\beta = \beta_k \in \|\alpha_j|_{\mathfrak{g}_0\cdot k}\|^{1-i} = \|\alpha_j|_\mathfrak{g}\|^{1-i}$, which is the desired conclusion.

The proof of (8) relies on the observation that symbols in $\|\alpha|_{\mathfrak{g}\cdot k}\|^i$ for any $k \in \{0,1\}$ cannot be present there as there are no variables in $\|\alpha|_\mathfrak{g}\|^i$. □

Note that the proof of (1) requires the use of case (6) from Def. 5.

Here is the crucial semantic property we expect from static decorations.

Definition 7 `(respecting of a decoration)`
We say that an expression σ *respects the static decoration for a variable* $\alpha \in \mathrm{FV}(E)$ *assuming substitution* S when the conditions hold

1. for each $i \in \{0,1\}$ if $\tau \in \|\alpha|_\mathfrak{g}\|^i$ and $\mathfrak{g}$ is accessible in σ then $S(\tau) \leq^i \sigma|_\mathfrak{g}$,
2. for each $i \in \{0,1\}$ if $\|\alpha|_{\mathfrak{g}\cdot 0}\|^i \cup \|\alpha|_{\mathfrak{g}\cdot 1}\|^i \neq \emptyset$ then $\sigma|_\mathfrak{g} \neq \bot^i$.

Proposition 4.2 `(solution respects decoration of` inequations)

Let S be a solution of E. For each variable $\alpha \in \mathrm{FV}(E)$ the expression $S(\alpha)$ respects the static decoration for α assuming S.

Proof:
The proof is by induction over $\mathfrak{g}$. If $\mathfrak{g} = \varepsilon$, we observe that $\tau \in \|\alpha|_\varepsilon\|^i$ implies that $\tau \leq^i \alpha \in E$. When $S \models E$ we get $S(\tau) \leq^i S(\alpha)$ so the condition (1) of Def. 7 follows.

As for the proof of case (2), we observe that when $\|\alpha|_0\|^i \cup \|\alpha|_1\|^i \neq \emptyset$, then for some β and $j \in \{0,1\}$ we have $\beta \in \|\alpha|_j\|^i$. Consequently, there is $\beta_0 \circledcirc \beta_1 \leq^i \alpha \in E$ where $\beta_j = \beta$. Then $S(\beta_0) \circledcirc S(\beta_1) \leq^i S(\alpha)$ which is impossible, when $S(\alpha)|_\varepsilon = S(\alpha) = \perp^i$.

In case $\mathfrak{g} = \mathfrak{g}' \cdot j$ for some $j \in \{0,1\}$, we prove first the condition (1) in Def. 7. When $\tau \in \|\alpha|_{\mathfrak{g}'\cdot j}\|^i$, we have by definition of static decoration $\beta_0 \circledcirc \beta_1 \leq^i \beta_u \in E$ for some $\beta_u \in \|\alpha|_{\mathfrak{g}'}\|^i$ with $\beta_j = \tau$. By the induction hypothesis, we obtain that $S(\beta_u) \leq^i S(\alpha)|_{\mathfrak{g}'}$. Since $\mathfrak{g}' \cdot j$ is accessible in $S(\alpha)$ we obtain that $S(\alpha)|_{\mathfrak{g}'} = \sigma_0 \circledcirc \sigma_1$ for some σ_0, σ_1 and as $S(\beta_0) \circledcirc S(\beta_1) \leq^i S(\beta_u) \leq^i S(\alpha)|_{\mathfrak{g}'} = \sigma_0 \circledcirc \sigma_1$, we can conclude that $S(\beta_j) \leq^i \sigma_j$. This in turn, implies that $S(\beta_j) \leq^i S(\alpha)|_{\mathfrak{g}'\cdot j}$, which concludes the proof of this case.

For the proof of condition (2) in Def. 7, we assume $\|\alpha|_{\mathfrak{g}'\cdot j\cdot 0}\|^i \cup \|\alpha|_{\mathfrak{g}'\cdot j\cdot 1}\|^i \neq \emptyset$, i.e. there is some variable $\beta \in \|\alpha|_{\mathfrak{g}'\cdot j\cdot 0}\|^i \cup \|\alpha|_{\mathfrak{g}'\cdot j\cdot 1}\|^i$. Consequently, there is some $\beta_u \in \|\alpha|_{\mathfrak{g}'\cdot j}\|^i$ and $\beta_0 \circledcirc \beta_1 \leq^i \beta_u \in E$ with some $\beta_j = \beta$. If, contrary to the conclusion, $S(\alpha)|_{\mathfrak{g}} = \perp^i$ then by the case (1) in Def. 7, obtained by the induction hypothesis, we conclude that $S(\beta_u) \leq^i S(\alpha)|_{\mathfrak{g}} = \perp^i$ and this makes impossible for S to satisfy $\beta_0 \circledcirc \beta_1 \leq^i \beta_u \in E$. This contradiction concludes the proof of this case.

□

The static annotations make it also possible to formulate criteria on accessibility. The criterion (2) below is especially interesting as it ascertains accessibility in many crucial situations in our proof.

Proposition 4.3 (accessible positions)
Let S be a solution of E and $\alpha \in \mathrm{FV}(E)$.

1. *If $\|\alpha|_{\mathfrak{g}}\|^0 \neq \emptyset$ and $\|\alpha|_{\mathfrak{g}}\|^1 \neq \emptyset$ then $\mathfrak{g}$ is accessible in $S(\alpha)$.*
2. *If $\beta \in \|\alpha|_{\mathfrak{g}}\|^i$ and $\beta \in \|\alpha'|_{\mathfrak{g}'}\|^{1-i}$ and $\mathfrak{g} \cdot \mathfrak{g}''$ is accessible in $S(\alpha)$ and $\mathfrak{g}' \cdot \mathfrak{g}''$ in $S(\alpha')$ then $\mathfrak{g}''$ is accessible in $S(\beta)$.*

Proof:
We prove (1) by induction over $\mathfrak{g}$. If $\mathfrak{g} = \varepsilon$ then $\mathfrak{g}$ is accessible by definition so the conclusion follows.

If $\mathfrak{g} = \mathfrak{g}' \cdot i$ where $i \in \{0,1\}$ then we conclude by definition of static decoration (Def. 6) that $\|\alpha|_{\mathfrak{g}'}\|^0 \neq \emptyset$ and $\|\alpha|_{\mathfrak{g}'}\|^1 \neq \emptyset$ so by the induction hypothesis $\mathfrak{g}'$ is accessible in $S(\alpha)$. Since $\|\alpha|_{\mathfrak{g}'\cdot i}\|^0 \neq \emptyset$ there is $\beta_u \in \|\alpha|_{\mathfrak{g}'}\|^0$ and β_0, β_1 such that $\beta_0 \circledcirc \beta_1 \leq \beta_u \in E$ (by Def. 6). Similarly, there is $\gamma_u \in \|\alpha|_{\mathfrak{g}'}\|^1$ and γ_0, γ_1 such that $\gamma_u \leq \gamma_0 \circledcirc \gamma_1 \in E$. As S respects the static decoration (Prop. 4.2), this implies that $S(\beta_0) \circledcirc S(\beta_1) \leq S(\beta_u) \leq S(\alpha)|_{\mathfrak{g}'} \leq S(\gamma_u) \leq S(\gamma_0) \circledcirc S(\gamma_1)$ and so $S(\alpha)|_{\mathfrak{g}'} = \pi_0 \circledcirc \pi_1$ for some π_0, π_1. Consequently, $\mathfrak{g}' \cdot i$ is accessible in $S(\alpha)$.

For the proof of (2) we observe that S respects decorations of E by Prop. 4.2. We prove now a stronger statement that for each $\mathfrak{h} \preceq \mathfrak{g}''$ it holds that $\mathfrak{h}$ is accessible in $S(\beta)$ and $S(\alpha')|_{\mathfrak{g}'\cdot\mathfrak{h}} \leq^i S(\beta)|_{\mathfrak{h}} \leq^i S(\alpha)|_{\mathfrak{g}\cdot\mathfrak{h}}$.

The proof is by induction over the length $|\mathfrak{h}|$. For $|\mathfrak{h}| = 0$, we observe that respecting of decorations means $S(\alpha')|_{\mathfrak{g}'} \leq^i S(\beta) = S(\beta)|_\varepsilon \leq^i S(\alpha)|_{\mathfrak{g}}$, which is the desired conclusion in the base case.

When $|\mathfrak{h}| > 0$, we observe that $\mathfrak{h} = \mathfrak{h}_0 \cdot k$ for some $k \in \{0,1\}$. By the induction hypothesis we obtain that $\mathfrak{h}_0$ is accessible in $S(\beta)$ and $S(\alpha')|_{\mathfrak{g}'\cdot\mathfrak{h}_0} \leq^i S(\beta)|_{\mathfrak{h}_0} \leq^i S(\alpha)|_{\mathfrak{g}\cdot\mathfrak{h}_0}$. Since $\mathfrak{g} \cdot \mathfrak{h}_0 \cdot k$ is accessible in $S(\alpha)$, we conclude that $S(\alpha)|_{\mathfrak{g}\cdot\mathfrak{h}_0} = \sigma_0 \circledcirc \sigma_1$ and $S(\alpha)|_{\mathfrak{g}\cdot\mathfrak{h}_0\cdot k} = \sigma_k$ for $k \in \{0,1\}$. Since $\mathfrak{g}' \cdot \mathfrak{h}_0 \cdot k$ is accessible in $S(\alpha')$, we conclude that $S(\alpha')_{\mathfrak{g}'\cdot\mathfrak{h}_0} = \tau_0 \circledcirc \tau_1$ and $S(\alpha')_{\mathfrak{g}'\cdot\mathfrak{h}_0\cdot k} = \tau_k$ for $k \in \{0,1\}$. Inequalities $\tau_0 \circledcirc \tau_1 \leq^i S(\beta)|_{\mathfrak{h}_0} \leq^i \sigma_0 \circledcirc \sigma_1$ imply that $\mathfrak{h}_0 \cdot k$ is accessible in $S(\beta)$.

Moreover, $S(\alpha')|_{\mathfrak{g}\cdot\mathfrak{h}_0\cdot k} = \tau_k \leq S(\beta)|_{\mathfrak{h}_0\cdot k} \leq \sigma_k = S(\alpha)|_{\mathfrak{g}\cdot\mathfrak{h}_0\cdot k}$ so the inductive step is proved.

□

The static decorations give also a nice criterion on when solutions cannot be finite.

Proposition 4.4 `(border depth)`
Let S be a finite solution of E. For each variable $\alpha \in \mathrm{FV}(E)$ and each position $\mathfrak{g} \in \{0,1\}^*$ such that $|\mathfrak{g}| > |\mathrm{FV}(E)|^2 + 1$ either $\|\alpha|_{\mathfrak{g}}\|^0 = \emptyset$ or $\|\alpha|_{\mathfrak{g}}\|^1 = \emptyset$.

Proof:
This proposition can be proved by a pumping argument akin to pumping technique from automata theory. We associate with a position $\mathfrak{g}$ a set $A_{\mathfrak{g}}$ of all strings w over $\mathrm{FV}(E)^2$ defined so that $w = w_0 \cdot w_2 \cdots w_{|\mathfrak{g}|-1}$ and $w_0 = \langle \alpha_0, \alpha_1 \rangle \in \|\alpha|_{\varepsilon}\|^0 \times \|\alpha|_{\varepsilon}\|^1$, and if $\beta_{i,0} \circledcirc \beta_{i,1} \leq^i \gamma_i$ for $w_j = \langle \gamma_0, \gamma_1 \rangle$ and $i \in \{0,1\}$ then $w_{j+1} = \langle \beta_{0,\mathfrak{g}(j+1)}, \beta_{1,\mathfrak{g}(j+1)} \rangle$ for $j \in \{0, \ldots, |\mathfrak{g}|-1\}$. Observe that for each $w \in A_{\mathfrak{g}}$ and $j \in \{0, \ldots, |w|-1\}$, we have $w(0) \in \|\alpha|_{\varepsilon}\|^0 \times \|\alpha|_{\varepsilon}\|^1$ and $w(j) \in \|\alpha|_{\mathfrak{g}(0)\cdots\mathfrak{g}(j-1)}\|^0 \times \|\alpha|_{\mathfrak{g}(0)\cdots\mathfrak{g}(j-1)}\|^1$ for $j > 0$. Moreover, if there is some $w \in A_{\mathfrak{g}}$ and positions $\mathfrak{g}_0, \mathfrak{g}_1, \mathfrak{g}_2$ such that $\mathfrak{g}_0 \cdot \mathfrak{g}_1 \cdot \mathfrak{g}_2 = \mathfrak{g}$ and $\mathfrak{g}_1 \neq \varepsilon$ as well as a pair $\langle \beta_0, \beta_1 \rangle$ such that $w(|\mathfrak{g}_0|) = w(|\mathfrak{g}_0 \cdot \mathfrak{g}_1|) = \langle \beta_0, \beta_1 \rangle$, then for each $k \in \mathbb{N}$ and $\mathfrak{g}' = \mathfrak{g}_0 \cdot \mathfrak{g}_1^k$ we obtain that $\langle \beta_0, \beta_1 \rangle \in \|\alpha|_{\mathfrak{g}'}\|^0 \times \|\alpha|_{\mathfrak{g}'}\|^1$.

We continue the proof by contradiction. If $\mathfrak{g}$ is a position such that $|\mathfrak{g}| > n = |\mathrm{FV}(E)|^2$ and $A_{\mathfrak{g}} \neq \emptyset$ then for each $w \in A_{\mathfrak{g}}$ there are positions at which the symbols in w repeat, i.e. there are $0 \leq k < l \leq |w|$ such that $w(k) = w(l)$. By observation above, we obtain that there are some $\mathfrak{g}_0, \mathfrak{g}_1$ such that $w(k) \in \|\alpha|_{\mathfrak{g}'}\|^0 \times \|\alpha|_{\mathfrak{g}'}\|^1$ for each $\mathfrak{g}' = \mathfrak{g}_0 \cdot (\mathfrak{g}_1)^m$. Prop. 4.3(1) implies that $\mathfrak{g}_0 \cdot (\mathfrak{g}_1)^m$ for each m is accessible, which is in contradiction with the fact that $S(\alpha)$ has bounded depth.

This proves that for positions $\mathfrak{g}$ such that $|\mathfrak{g}| > |\mathrm{FV}(E)|^2$ we have $\|\alpha|_{\mathfrak{g}}\|^0 \cap \mathrm{FV}(E) = \emptyset$ or $\|\alpha|_{\mathfrak{g}}\|^1 \cap \mathrm{FV}(E) = \emptyset$. It is still possible that $\|\alpha|_{\mathfrak{g}}\|^0 \subseteq \{\bot, \top\}$ or $\|\alpha|_{\mathfrak{g}}\|^1 \subseteq \{\bot, \top\}$. However, once $\|\alpha|_{\mathfrak{g}}\|^i \subseteq \{\bot, \top\}$, we obtain $\|\alpha|_{\mathfrak{g}\cdot j}\|^i = \emptyset$ for each $j \in \{0,1\}$. Therefore, if for some $\mathfrak{g}$ of length $|\mathrm{FV}(E)|^2 + 1$ we have still $\|\alpha|_{\mathfrak{g}}\|^0 \neq \emptyset$ and $\|\alpha|_{\mathfrak{g}}\|^1 \neq \emptyset$ then for $\mathfrak{g}$ where $|\mathfrak{g}| > |\mathrm{FV}(E)|^2 + 1$ we have $\|\alpha|_{\mathfrak{g}}\|^0 = \emptyset$ or $\|\alpha|_{\mathfrak{g}}\|^1 = \emptyset$. $\square$

Assumption From now on we let n_E be $|\mathrm{FV}(E)|^2 + 1$.

Static annotations make it also possible to reason in such a fashion that a presence of $\bot^i$ at some position in some expression causes also some other expressions to assume $\bot^i$ on some positions. The lemma below is an example of such a situation. These, in turn, help to define modifications on substitutions that preserve solvability. This lemma is not used directly in the proof, but it helps greatly in understanding the crucial definition of dynamic decoration (Def. 9).

Lemma 4.5 `(extreme values propagate)`
Let $\mathfrak{g}$ be a position such that $S(\alpha)|_{\mathfrak{g}} = \bot^i$. Assume that for some $\mathfrak{g}' \preceq \mathfrak{g}$ we have $\beta \in \|\alpha|_{\mathfrak{g}'}\|^i$. If $\mathfrak{g}'' = \mathfrak{g} \swarrow \mathfrak{g}'$ then for each $\tau \in \|\beta|_{\mathfrak{g}''}\|^i$ it holds that $S(\tau) = \bot^i$.

Proof:
Let us take $\mathfrak{g}''$ such that $\mathfrak{g} = \mathfrak{g}' \cdot \mathfrak{g}''$ and $\beta \in \|\alpha|_{\mathfrak{g}'}\|^j$. Let us take any $\tau \in \|\beta|_{\mathfrak{g}''}\|^i$.

One of the basic properties of static decorations (Prop. 4.1(2)) implies that $\|\beta|_{\mathfrak{g}''}\|^j \subseteq \|\alpha|_{\mathfrak{g}'\cdot\mathfrak{g}''}\|^j$ as we have $\beta \in \|\alpha|_{\mathfrak{g}'}\|^j$. This means that $\tau \in \|\alpha|_{\mathfrak{g}}\|^j$. As S is a solution, $S(\alpha)$ respects decoration for α (see Prop. 4.2). This means that $S(\tau) \leq^j S(\alpha)|_{\mathfrak{g}} = \bot^i$, and in turn $S(\tau) = \bot^i$. $\square$

4.1 Extension along a position

The main construction of the current paper consists in distinguishing a critical path in a solution of E. Once this is done we want to guarantee by means of suitably defined inequations that this position is accessible in all possible solutions as well as the labels at its end are appropriate. To obtain the guarantee we augment E with additional inequations. Before we define the way it is done, we need a definition of expressions that are used to enforce the form of solutions.

Let us assume that we have an infinite set of variables that are not in $\mathrm{FV}(E)$ and have the form $\alpha^{\mathfrak{g}}_{i,j}$ where $i, j \in \{0,1\}$ and $\mathfrak{g}$ is a position. The intent here is that the annotation $\mathfrak{g}$ tells that variable takes part in enforcing of the labels at position $\mathfrak{g}$. The annotation i tells that the variable is the i-th child of the expression being defined, and the annotation j tells that the variable takes part in enforcing in the solution for the original variable α_j. We define the following expressions.

- $B_E^{\mathfrak{g}}(\bot,i) = \bot$, $B_E^{\mathfrak{g}}(\top,i) = \top$, $B_E^{\mathfrak{g}}(\circledcirc,i) = \alpha_{0,i}^{\mathfrak{g}} \circledcirc \alpha_{1,i}^{\mathfrak{g}}$.

This makes it possible to define our notion of extension.

Definition 8 `(extensions along a position)`
Let $\alpha_0, \alpha_1 \in \mathrm{FV}(E)$ and $\mathfrak{g}$ be a position. Let $s_1, s_2 \in \{\bot, \top, \circledcirc\}$. We define recursively $(\alpha_0, \alpha_1, s_0, s_1, \mathfrak{g})$-extension of E as follows:

- $(\alpha_0, \alpha_1, s_0, s_1, \varepsilon)$-extension of E is

$$E \cup \{\, B_E^{\varepsilon}(s_0,0) \le \alpha_0,\ \alpha_0 \le B_E^{\varepsilon}(s_0,0), \\ B_E^{\varepsilon}(s_1,1) \le \alpha_1,\ \alpha_1 \le B_E^{\varepsilon}(s_1,1)\,\}$$

- $(\alpha_0, \alpha_1, s_0, s_1, i \cdot \mathfrak{g})$-extension of E is

$$E' \cup \{\, B_E^{i\cdot\mathfrak{g}}(\circledcirc,0) \le \alpha_0, \alpha_0 \le B_E^{i\cdot\mathfrak{g}}(\circledcirc,0), \\ B_E^{i\cdot\mathfrak{g}}(\circledcirc,1) \le \alpha_1, \alpha_1 \le B_E^{i\cdot\mathfrak{g}}(\circledcirc,1)\}$$

  where E' is $(\alpha_{i,0}^{i\cdot\mathfrak{g}}, \alpha_{i,1}^{i\cdot\mathfrak{g}}, s_0, s_1, \mathfrak{g})$-extension of E.

This notion has the following property.

Proposition 4.6 `(properties of extensions along a position)`
Let $\alpha_0, \alpha_1 \in \mathrm{FV}(E)$. Let E' be an $(\alpha_0, \alpha_1, s_0, s_1, \mathfrak{g})$-extension of E.

1. *If $S \models E'$ then $S \models E$ and for every $i \in \{0,1\}$ $\blacktriangle S(\alpha_i)|_{\mathfrak{g}} = s_i$.*
2. *If $S \models E$ and for every $i \in \{0,1\}$ we have $\blacktriangle S(\alpha_i)|_{\mathfrak{g}} = s_i$ then $S \models E'$.*
3. *If E is saturated and does not contain $\sigma \le^i \bot^i$ where $\sigma \notin \mathrm{FV}(E) \cup \{\bot^i\}$ for any $i \in \{0,1\}$ then E' is saturated and does not contain $\sigma \le^i \bot^i$ with $\sigma \neq \bot^i$ for any $i \in \{0,1\}$.*

Proof:
The proof is by induction over $n = |\mathfrak{g}|$. Details are left to the reader. □

As proofs do not rely on finitness of substitutions, the propositions in this section, except for one clearly marked Prop. 4.4, remain true both when S substitutes only finite type expressions and when S substitutes infinite ones.

5 Manipulations on solutions

We define below modifications of substitutions and give fine criteria on when they preserve solvability i.e. in which cases if the original substitution is a solution then the resulting one remains such. The definitions of the modifications rely on a notion of dynamic decoration. This decoration provides a generalisation of the conditions mentioned in Lemma 4.5. Given a variable α and a position $\mathfrak{g}$, it settles the set of positions in solutions for variables of $\mathrm{FV}(E)$ that have to be modified to maintain solvability in case a constant $\bot^i$ is put at a position $\mathfrak{g}$ in $S(\alpha)$. In this way, we distinguish for each variable β positions $\mathfrak{h}$ such that the expressions $S(\beta)|_{\mathfrak{h}}$ are in direct relation $\le^i$ with $S(\alpha)|_{\mathfrak{g}}$.

In the definition below, we consider quadruples where the first coordinate contains such β and the last one contains $\mathfrak{h}$. The intermediate coordinates are introduced here for technical reasons to pinpoint exactly the reason why the solution on the variable at the position must be decorated with $\bot^i$ once the solution on α at $\mathfrak{g}$ is. The relations used in the definition below are presented in Fig. 1.

Definition 9 `(dynamic decoration)`
Consider a variable $\alpha \in \mathrm{FV}(E)$. For a position $\mathfrak{g}$ we define sets of quadruples, called *quasi-dynamic decorations*, as

$$[\![\alpha|_{\mathfrak{g}}]\!]^i = \{\, \langle \beta, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2 \rangle \mid \exists \beta' \in \mathrm{FV}(E) \text{ s.t. } \\ (1)\ \beta' \in \|\alpha|_{\mathfrak{h}_0}\|^i \text{ and} \\ (2)\ \beta' \in \|\beta|_{\mathfrak{h}_1}\|^{1-i} \text{ and} \\ (3)\ \mathfrak{g} \not\swarrow \mathfrak{h}_0 = \mathfrak{h}_2 \not\swarrow \mathfrak{h}_1 \}.$$

We also let

$$\overline{[\![\alpha|_{\mathfrak{g}}]\!]}^i(\beta, \mathfrak{h}) = \{\mathfrak{h}_2 \,|\, \langle \beta, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2 \rangle \in [\![\alpha|_{\mathfrak{g}}]\!]^i \text{ and} \\ \text{either } \mathfrak{h}_2 \preceq \mathfrak{h} \text{ or } \mathfrak{h} \preceq \mathfrak{h}_2 \}.$$

Finally, we define the *dynamic decoration* as:

$$(\!|\alpha|_{\mathfrak{g}}|\!)^i = \{\langle \beta, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2 \rangle \in [\![\alpha|_{\mathfrak{g}}]\!]^i \mid \\ \mathfrak{h}_2 = \min_{\preceq} \overline{[\![\alpha|_{\mathfrak{g}}]\!]}^i(\beta, \mathfrak{h}_2)\}.$$

It is important to realise that the notion of dynamic decoration does not depend on any substitution or

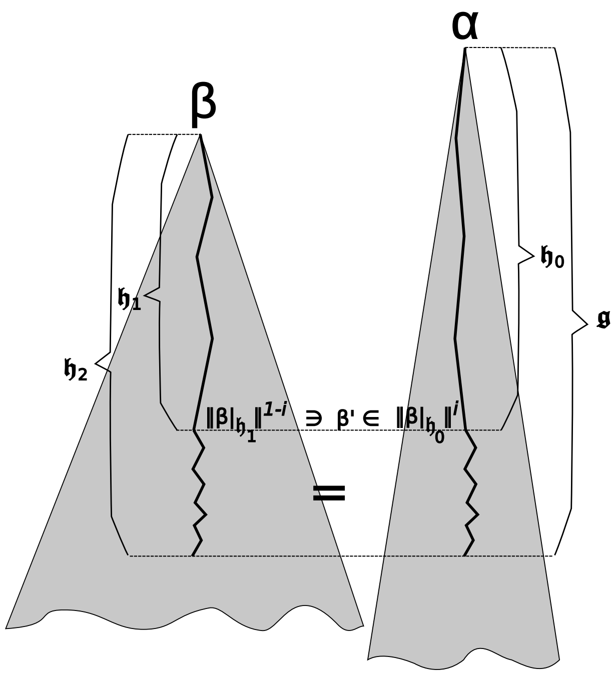

Figure 1: The element $\langle \beta, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2 \rangle$ belongs to the quasi-dynamic decoration $\llbracket \alpha|_{\mathfrak{g}} \rrbracket^i$ when the static decoration $\|\alpha|_{\mathfrak{h}_0}\|^i$ for α at $\mathfrak{h}_0$ contains some variable β' which is also an element of the static decoration $\|\beta|_{\mathfrak{h}_1}\|^{1-i}$ for β at $\mathfrak{h}_1$. Moreover, the position $\mathfrak{h}_0$ can be extended to $\mathfrak{g}$ with the same string as the position $\mathfrak{h}_1$ to $\mathfrak{h}_2$.

solution. It is defined solely in terms of the original problem input.

Example 5 *We can use the observation made in Example 4 that* $\|\alpha|_{010}\|^0 = \{\beta_{01}, \alpha_{010}\}$ *and* $\|\beta_0|_1\|^1 = \{\beta_{01}, \alpha_{010}\}$ *to conclude that* $\langle \beta_0, 010, 1, 1 \cdot \mathfrak{h} \rangle \in \llbracket \alpha|_{010 \cdot \mathfrak{h}} \rrbracket^0$.

We have to develop a little bit of the theory of these dynamic decorations. In particular, we would like to provide criteria on which elements are implied to belong to dynamic decoration provided that a particular inequation $\sigma_1 \leq \sigma_2$ is in E.

Proposition 5.1 `(properties of dynamic decorations)`

1. *If* $\langle \beta, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2 \rangle \in \llbracket \alpha|_{\mathfrak{g}} \rrbracket^i$ *and* $\gamma \leq^i \beta \in E$ *then* $\langle \gamma, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2 \rangle \in \llbracket \alpha|_{\mathfrak{g}} \rrbracket^i$.
2. *If* $\langle \beta, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2 \rangle \in \llbracket \alpha|_{\mathfrak{g}} \rrbracket^i$*, with* $j \preceq \mathfrak{h}_2$ *and* $\beta_0 \circledcirc \beta_1 \leq^i \beta \in E$ *then* $\langle \beta_j, \mathfrak{h}_0, \mathfrak{h}_1 \swarrow j, \mathfrak{h}_2 \swarrow j \rangle \in \llbracket \alpha|_{\mathfrak{g}} \rrbracket^i$.
3. *If* $\langle \beta, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2 \rangle \in \llbracket \alpha|_{\mathfrak{g}} \rrbracket^i$ *and* $\beta_u \leq^i \beta_0 \circledcirc \beta_1 \in E$ *with* $\beta_j = \beta$ *for some* $j \in \{0, 1\}$ *then* $\langle \beta_u, \mathfrak{h}_0, j \cdot \mathfrak{h}_1, j \cdot \mathfrak{h}_2 \rangle \in \llbracket \alpha|_{\mathfrak{g}} \rrbracket^i$.
4. *If* $\langle \beta, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2 \rangle \in \llbracket \alpha|_{\mathfrak{g}} \rrbracket^i$ *and* $\sigma_0 \leq^i \sigma_1 \in E$ *with* $\gamma = \sigma_0|_{\mathfrak{g}_0}$ *and* $\beta = \sigma_1|_{\mathfrak{g}_1}$ *as well as* $\mathfrak{g}_0 \preceq \mathfrak{h}_2$ *then* $\langle \gamma, \mathfrak{h}_0, (\mathfrak{g}_1 \cdot \mathfrak{h}_1) \swarrow \mathfrak{g}_0, (\mathfrak{g}_1 \cdot \mathfrak{h}_2) \swarrow \mathfrak{g}_0 \rangle \in \llbracket \alpha|_{\mathfrak{g}} \rrbracket^i$.
5. *If* $\langle \beta, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2 \rangle \in \llparenthesis \alpha|_{\mathfrak{g}} \rrparenthesis^i$ *and* $\sigma_0 \leq^i \sigma_1 \in E$ *with* $\gamma = \sigma_0|_{\mathfrak{g}_0}$ *and* $\beta = \sigma_1|_{\mathfrak{g}_1}$ *then there are some* $\mathfrak{h}'_0, \mathfrak{h}'_1$ *and* $\mathfrak{h}'_2 \preceq (\mathfrak{g}_1 \cdot \mathfrak{h}_2) \swarrow \mathfrak{g}_0$ *such that* $\langle \gamma, \mathfrak{h}'_0, \mathfrak{h}'_1, \mathfrak{h}'_2 \rangle \in \llparenthesis \alpha|_{\mathfrak{g}} \rrparenthesis^i$.
6. *For each* $\langle \beta, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2 \rangle, \langle \beta, \mathfrak{h}'_0, \mathfrak{h}'_1, \mathfrak{h}'_2 \rangle \in \llparenthesis \alpha|_{\mathfrak{g}} \rrparenthesis^i$ *if* $\mathfrak{h}_2 \neq \mathfrak{h}'_2$ *then neither* $\mathfrak{h}_2 \prec \mathfrak{h}'_2$ *nor* $\mathfrak{h}'_2 \prec \mathfrak{h}_2$.
7. *Let* S *be a solution of* E *and* $S(\alpha)|_{\mathfrak{g}} = \perp^i$*. For each* $\langle \beta, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2 \rangle \in \llbracket \alpha|_{\mathfrak{g}} \rrbracket^i$ *we have* $S(\beta)|_{\mathfrak{h}_2} = \perp^i$.

Proof:

<u>For the proof of the case (1)</u> we consider $\langle \beta, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2 \rangle \in \llbracket \alpha|_{\mathfrak{g}} \rrbracket^i$. By definition of the decoration $\llbracket \alpha|_{\mathfrak{g}} \rrbracket^i$, there is β' such that $\beta' \in \|\alpha|_{\mathfrak{h}_0}\|^i$ and $\beta' \in \|\beta|_{\mathfrak{h}_1}\|^{1-i}$ and $\mathfrak{g} \swarrow \mathfrak{h}_0 = \mathfrak{h}_2 \swarrow \mathfrak{h}_1$. Note that $\gamma \leq^i \beta \in E$ implies that $\beta \in \|\gamma|_\varepsilon\|^{1-i}$. By Prop. 4.1(2) we obtain that $\beta' \in \|\gamma|_{\mathfrak{h}_1}\|^{1-i}$. Thus, we obtain that $\langle \gamma, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2 \rangle \in \llbracket \alpha|_{\mathfrak{g}} \rrbracket^i$.
<u>For the proof of the case (2)</u> we consider $\langle \beta, \mathfrak{h}_0, j \cdot \mathfrak{h}_1, j \cdot \mathfrak{h}_2 \rangle \in \llbracket \alpha|_{\mathfrak{g}} \rrbracket^i$. By definition of $\llbracket \alpha|_{\mathfrak{g}} \rrbracket^i$, there is β' such that $\beta' \in \|\alpha|_{\mathfrak{h}_0}\|^i$ and $\beta' \in \|\beta|_{j \cdot \mathfrak{h}_1}\|^{1-i}$. Note that $\beta_0 \circledcirc \beta_1 \leq^i \beta \in E$ implies that $\beta_j \in \|\beta|_j\|^i$. Of course $\beta_j \in \|\beta_j|_\varepsilon\|^{1-i}$. By Prop. 4.1(6) we obtain that $\|\beta|_j\|^{1-i} \subseteq \|\beta_j|_\varepsilon\|^{1-i}$. By Prop. 4.1(3) we obtain further that $\|\beta|_{j \cdot \mathfrak{h}_1}\|^{1-i} \subseteq \|\beta_j|_{\mathfrak{h}_1}\|^{1-i}$ and consequently $\beta' \in \|\beta_j|_{\mathfrak{h}_1}\|^{1-i}$. In this way we obtain the

first two conditions necessary to get $\langle\beta_j, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2\rangle \in [\![\alpha|_{\mathfrak{g}}]\!]^i$. We also have $\mathfrak{g} \not\swarrow \mathfrak{h}_0 = j \cdot \mathfrak{h}_2 \not\swarrow j \cdot \mathfrak{h}_1 = \mathfrak{h}_2 \not\swarrow \mathfrak{h}_1$, which is the third condition.

For the proof of the case (3) we consider $\langle\beta, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2\rangle \in [\![\alpha|_{\mathfrak{g}}]\!]^i$. By definition of $[\![\alpha|_{\mathfrak{g}}]\!]^i$, there is β' such that $\beta' \in \|\alpha|_{\mathfrak{h}_0}\|^i$ and $\beta' \in \|\beta|_{\mathfrak{h}_1}\|^{1-i}$. Note that $\beta_u \leq^i \beta_0 \circledcirc \beta_1 \in E$ with $\beta_j = \beta$ implies that $\beta \in \|\beta_u|_j\|^{1-i}$. This implies by Prop. 4.1(2) that $\|\beta|_{\mathfrak{h}_1}\|^{1-i} \subseteq \|\beta_u|_{j\cdot\mathfrak{h}_1}\|^{1-i}$. Consequently, $\beta' \in \|\beta_u|_{j\cdot\mathfrak{h}_1}\|^{1-i}$ and first two conditions necessary to get $\langle\beta_u, \mathfrak{h}_0, j\cdot\mathfrak{h}_1, j\cdot\mathfrak{h}_2\rangle \in [\![\alpha|_{\mathfrak{g}}]\!]^i$ are fulfilled. We also have $\mathfrak{g} \not\swarrow \mathfrak{h}_0 = \mathfrak{h}_2 \not\swarrow \mathfrak{h}_1 = (j\cdot\mathfrak{h}_2) \not\swarrow (j\cdot\mathfrak{h}_1)$.

For the proof of the case (4), we observe that this case is a compact reformulation of cases (1)-(3).

For the proof of the case (5) we take $\langle\beta, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2\rangle \in (\!|\alpha|_{\mathfrak{g}}|\!)^i$. This implies further that $\langle\beta, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2\rangle \in [\![\alpha|_{\mathfrak{g}}]\!]^i$. By case (4) of this proposition, we obtain that $\langle\gamma, \mathfrak{h}_0, (\mathfrak{g}_1\cdot\mathfrak{h}_1) \not\swarrow \mathfrak{g}_0, (\mathfrak{g}_1\cdot\mathfrak{h}_2) \not\swarrow \mathfrak{g}_0\rangle \in [\![\alpha|_{\mathfrak{g}}]\!]^i$. This means that $\overline{[\![\alpha|_{\mathfrak{g}}]\!]}^i(\gamma, (\mathfrak{g}_1\cdot\mathfrak{h}_2) \not\swarrow \mathfrak{g}_0) \neq \emptyset$ and by definition

$$\mathfrak{h}_2' = \min_{\preceq} \overline{[\![\alpha|_{\mathfrak{g}}]\!]}^i(\gamma, (\mathfrak{g}_1\cdot\mathfrak{h}_2) \not\swarrow \mathfrak{g}_0) \preceq (\mathfrak{g}_1\cdot\mathfrak{h}_2) \not\swarrow \mathfrak{g}_0$$

so $\langle\gamma, \mathfrak{h}_0', \mathfrak{h}_1', \mathfrak{h}_2'\rangle \in (\!|\alpha|_{\mathfrak{g}}|\!)^i$ for some $\mathfrak{h}_0', \mathfrak{h}_1'$.

For the proof of the case (6) we observe that the conclusion is obvious since the elements of $(\!|\alpha|_{\mathfrak{g}}|\!)^i$ are minimal positions.

For the proof of the case (7) we observe that $\langle\beta, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2\rangle \in [\![\alpha|_{\mathfrak{g}}]\!]^i$ means that there is β' such that $\beta' \in \|\alpha|_{\mathfrak{h}_0}\|^i$ and $\beta' \in \|\beta|_{\mathfrak{h}_1}\|^{1-i}$. As $S(\alpha), S(\beta')$ and $S(\beta)$ respect static decoration (see Prop. 4.2) we obtain that $S(\beta')|_{\mathfrak{g}\not\swarrow\mathfrak{h}_0} \leq^i S(\alpha)|_{\mathfrak{g}} = \perp^i$ and $S(\beta)|_{\mathfrak{h}_2} \leq^i S(\beta')|_{\mathfrak{h}_2\not\swarrow\mathfrak{h}_1}$ By definition of $\langle\beta, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2\rangle \in [\![\alpha|_{\mathfrak{g}}]\!]^i$ we also have $\mathfrak{g} \not\swarrow \mathfrak{h}_0 = \mathfrak{h}_2 \not\swarrow \mathfrak{h}_1$ so consequently $S(\beta)|_{\mathfrak{h}_2} \leq^i \perp^i$.

□

Assumption For the rest of this section we fix, in addition to E, a substitution S such that $S \models E$.

The just defined dynamic decoration is used to formulate two definitions of substitution transformations. We show later how they can serve as basic steps in pumping lemma that make it possible to find solutions of computable size. Here is the first of the two definitions.

Definition 10 `(substitution cutting)`
We define the substitution $S^{\mathfrak{g}}_{\alpha,i}$ for $i = 0, 1$ as

$$S^{\mathfrak{g}}_{\alpha,i}(\beta) = S(\beta)[A \leftarrow \perp^i]$$
where $A = \{\mathfrak{h}'' \mid \langle\beta, \mathfrak{h}, \mathfrak{h}', \mathfrak{h}''\rangle \in (\!|\alpha|_{\mathfrak{g}}|\!)^i$ and $\mathfrak{h}''$ is accessible in $S(\beta)\}$.

Note that when $A = \emptyset$ we obtain $S^{\mathfrak{g}}_{\alpha,i}(\beta) = S(\beta)$.

Proposition 5.2 `(properties of cutting)`
Let σ be an expression.

1. *If $\mathfrak{h}$ is accessible in $S^{\mathfrak{g}}_{\alpha,i}(\sigma)$ then $\mathfrak{h}$ is accessible in $S(\sigma)$.*
2. *No new $\perp^{1-i}$ is introduced by $S^{\mathfrak{g}}_{\alpha,i}$, i.e. if $S^{\mathfrak{g}}_{\alpha,i}(\sigma)|_{\mathfrak{h}} = \perp^{1-i}$ then $S(\sigma)|_{\mathfrak{h}} = \perp^{1-i}$.*
3. *Let $\sigma_0 \leq \sigma_1 \in E$. If $S^{\mathfrak{g}}_{\alpha,i}(\sigma_i)|_{\mathfrak{h}} = S(\sigma_{1-i})|_{\mathfrak{h}} = \perp^{1-i}$ as well as $\mathfrak{h}$ is accessible in $S^{\mathfrak{g}}_{\alpha,i}(\sigma_{1-i})$ then $S^{\mathfrak{g}}_{\alpha,i}(\sigma_{1-i})|_{\mathfrak{h}} = \perp^{1-i}$.*

Proof:

For the proof of the case (1) we first note that for each expression τ and a set A, each position $\mathfrak{g}$ accessible in $\tau[A \leftarrow \perp^i]$ is accessible in τ. This is proved by an easy induction over $|\tau|$, which is left to the reader. The conclusion follows since $S^{\mathfrak{g}}_{\alpha,i}(\sigma) = S(\sigma)[A \leftarrow \perp^i]$ for some A.

For the proof of the case (2) we obtain the conclusion by an immediate instantiation of a slightly more general statement that if $\tau[A \leftarrow \perp^i]|_{\mathfrak{h}} = \perp^{i-1}$ then $\tau|_{\mathfrak{h}} = \perp^{i-1}$. This statement is proved in turn by induction over the size n of $\tau[A \leftarrow \perp^i]$.

In case $n = 1$, we observe that $\tau[A \leftarrow \perp^i]$ is equal to either $\perp^i$ or τ. In the former case the assumption of the case is false so the conclusion follows. In the latter case the conclusion is trivially true.

In case $n > 1$, we first eliminate the case when $\mathfrak{h} = \varepsilon$ and $\tau = \perp^i$. These imply that $\tau[A \leftarrow \perp^i] = \perp^i$, which is different than assumed $\perp^{1-i}$, contradiction.

The case $\mathfrak{h} = \varepsilon$ and $\tau = \perp^{1-i}$ is trivial. It remains to consider cases where $\tau = \tau_0 \circledcirc \tau_1$. If $\mathfrak{h} = \varepsilon$, we obtain immediately contradiction. Suppose now that $\mathfrak{h} = j \cdot \mathfrak{h}'$ for some j. We see that $\perp^{1-i} = \tau[A \leftarrow \perp^i]|_{\mathfrak{h}} = ((\tau_0 \circledcirc \tau_1)[A \leftarrow \perp^i])|_{j\cdot\mathfrak{h}'} = \tau_j[A_j \leftarrow \perp^i]|_{\mathfrak{h}'}$. We can now apply the induction hypothesis and conclude that $\tau_j|_{\mathfrak{h}'} = \perp^{1-i}$, which implies that $\tau|_{\mathfrak{h}} = (\tau_0 \circledcirc \tau_1)|_{\mathfrak{h}} = (\tau_0 \circledcirc \tau_1)|_{j\cdot\mathfrak{h}'} = \tau_j|_{\mathfrak{h}'} = \perp^{1-i}$.

For the proof of the case (3) we proceed by contradiction. Suppose that, contrary to the conclusion, the inequality $S^{\mathfrak{g}}_{\alpha,i}(\sigma_{1-i})|_{\mathfrak{h}} \neq \perp^{1-i}$ holds. Since $S(\sigma_{1-i})|_{\mathfrak{h}} = \perp^{1-i}$, we obtain by the contraposition of the case (1) of the current lemma that $S^{\mathfrak{g}}_{\alpha,i}(\sigma_{1-i})|_{\mathfrak{h}} \in \{\perp^i, \perp^{1-i}\}$ and then that $S^{\mathfrak{g}}_{\alpha,i}(\sigma_{1-i})|_{\mathfrak{h}} = \perp^i$. This means that $\mathfrak{h} = \mathfrak{h}_0 \cdot \mathfrak{h}_1$ where $\sigma_{1-i}|_{\mathfrak{h}_0} = \beta$ for some variable β and $\langle \beta, \mathfrak{g}_1, \mathfrak{g}_2, \mathfrak{h}_1 \rangle \in (\!|\alpha|_{\mathfrak{g}}|\!)^i$ for some $\mathfrak{g}_1, \mathfrak{g}_2$. Since $\sigma_i \leq^i \sigma_{1-i}$, we can conclude by Prop. 5.1(4) that $\langle \gamma, \mathfrak{g}_1, (\mathfrak{h}_0 \cdot \mathfrak{g}_2) \not\swarrow \mathfrak{h}'_0, (\mathfrak{h}_0 \cdot \mathfrak{h}_1) \not\swarrow \mathfrak{h}'_0 \rangle \in [\![\alpha|_{\mathfrak{g}}]\!]^i$ where $\gamma = \sigma_i|_{\mathfrak{h}'_0}$ and $\mathfrak{h}'_0 \preceq \mathfrak{h}$. We observe now that $\perp^i = S^{\mathfrak{g}}_{\alpha,i}(\gamma)|_{(\mathfrak{h}_0\cdot\mathfrak{h}_1)\not\swarrow\mathfrak{h}'_0} = S^{\mathfrak{g}}_{\alpha,i}(\sigma_i)|_{\mathfrak{h}'_0\cdot((\mathfrak{h}_0\cdot\mathfrak{h}_1)\not\swarrow\mathfrak{h}'_0)}$. We consider now the position $\mathfrak{h}'_0 \cdot ((\mathfrak{h}_0 \cdot \mathfrak{h}_1) \not\swarrow \mathfrak{h}'_0) = \mathfrak{h}_0 \cdot \mathfrak{h}_1 = \mathfrak{h}$. Consequently, $S^{\mathfrak{g}}_{\alpha,i}(\sigma_i)|_{\mathfrak{h}} = \perp^i$, which contradicts the assumption that $S^{\mathfrak{g}}_{\alpha,i}(\sigma_i)|_{\mathfrak{h}} = \perp^{1-i}$. $\square$

Lemma 5.3 `(solution cutting is a solution)`
If $\mathfrak{g}$ is a position such that $\|\alpha|_{\mathfrak{g}}\|^i \subseteq \{\perp^i\}$ for some $i \in \{0,1\}$ then $S^{\mathfrak{g}}_{\alpha,i} \models E$. Moreover, if S avoids constants then $S^{\mathfrak{g}}_{\alpha,i}$ does.

Proof:
The proof is by contradiction. Suppose that $S^{\mathfrak{g}}_{\alpha,i}$ does not solve E. In that case there is a pair of expressions σ_0, σ_1 such that $S^{\mathfrak{g}}_{\alpha,i}(\sigma_0) \not\leq S^{\mathfrak{g}}_{\alpha,i}(\sigma_1)$ even though $\sigma_0 \leq \sigma_1 \in E$. Suppose $\mathfrak{g}'$ is a maximal position such that $S^{\mathfrak{g}}_{\alpha,i}(\sigma_0)|_{\mathfrak{g}'} \not\leq S^{\mathfrak{g}}_{\alpha,i}(\sigma_1)|_{\mathfrak{g}'}$. This means that at least one of the expressions $S^{\mathfrak{g}}_{\alpha,i}(\sigma_0)|_{\mathfrak{g}'}, S^{\mathfrak{g}}_{\alpha,i}(\sigma_1)|_{\mathfrak{g}'}$ is a constant. We can exclude the situation that $S^{\mathfrak{g}}_{\alpha,i}(\sigma_j)|_{\mathfrak{g}'} = \perp^j$ for any $j \in \{0,1\}$ since this would make $S^{\mathfrak{g}}_{\alpha,i}(\sigma_0)|_{\mathfrak{g}'} \not\leq S^{\mathfrak{g}}_{\alpha,i}(\sigma_1)|_{\mathfrak{g}'}$ impossible. Therefore, we have (*) $S^{\mathfrak{g}}_{\alpha,i}(\sigma_j)|_{\mathfrak{g}'} = \perp^{1-j}$ for some $j \in \{0,1\}$.

In case $i = j$, we must have $S^{\mathfrak{g}}_{\alpha,i}(\sigma_i)|_{\mathfrak{g}'} = S(\sigma_i)|_{\mathfrak{g}'} = \perp^{1-i}$ as no new $\perp^{1-i}$ is introduced by $S^{\mathfrak{g}}_{\alpha,i}$ (see Prop. 5.2(2)). Since S is a solution, this means $S(\sigma_{1-i})|_{\mathfrak{g}'} = \perp^{1-i}$, but then Prop. 5.2(3) implies that $S^{\mathfrak{g}}_{\alpha,i}(\sigma_{1-i})|_{\mathfrak{g}'} = \perp^{1-i}$, which contradicts the assumption that $S^{\mathfrak{g}}_{\alpha,i}(\sigma_0)|_{\mathfrak{g}'} \not\leq S^{\mathfrak{g}}_{\alpha,i}(\sigma_1)|_{\mathfrak{g}'}$.

In case $i \neq j$, we obtain immediately that $i = 1 - j$. In this situation (*) can be formulated as $S^{\mathfrak{g}}_{\alpha,i}(\sigma_{1-i})|_{\mathfrak{g}'} = \perp^i$. In case also $S(\sigma_{1-i})|_{\mathfrak{g}'} = \perp^i$ we obtain that $S(\sigma_i)|_{\mathfrak{g}'} = \perp^i$ as $\sigma_i \leq^i \sigma_{1-i} \in E$. Since $\mathfrak{g}'$ is accessible in $S^{\mathfrak{g}}_{\alpha,i}(\sigma_i)$ we must have $S^{\mathfrak{g}}_{\alpha,i}(\sigma_i)|_{\mathfrak{g}'} = \perp^i$ as in that case the label at the position $\mathfrak{g}'$ in $S(\sigma_i)$ cannot be different than in $S^{\mathfrak{g}}_{\alpha,i}(\sigma_i)$. This again contradicts the assumption that $S^{\mathfrak{g}}_{\alpha,i}(\sigma_0)|_{\mathfrak{g}'} \not\leq S^{\mathfrak{g}}_{\alpha,i}(\sigma_1)|_{\mathfrak{g}'}$.

Suppose now that $S(\sigma_{1-i})|_{\mathfrak{g}'} \neq \perp^i$. This means that $S(\sigma_{1-i})|_{\mathfrak{g}'}$ is different than $S^{\mathfrak{g}}_{\alpha,i}(\sigma_{1-i})|_{\mathfrak{g}'} = \perp^i$. By definition of $S^{\mathfrak{g}}_{\alpha,i}$, we obtain through a routine case analysis of the forms of σ_i and σ_{1-i} that there are some $\mathfrak{g}_0, \mathfrak{g}_1, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2, \mathfrak{h}'_2$ such that

- $\sigma_i|_{\mathfrak{g}_0} \in \mathrm{FV}(E)$ and $\sigma_{1-i}|_{\mathfrak{g}_1} \in \mathrm{FV}(E)$,
- $\langle \sigma_{1-i}|_{\mathfrak{g}_1}, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2 \rangle \in (\!|\alpha|_{\mathfrak{g}}|\!)^i$,
- $\mathfrak{g}_1 \cdot \mathfrak{h}_2 = \mathfrak{g}' = \mathfrak{g}_0 \cdot \mathfrak{h}'_2$.

Observe also that $\mathfrak{g}'$ is accessible in $S(\sigma_i)$ as it is accessible in $S^{\mathfrak{g}}_{\alpha,i}(\sigma_i)$, see Prop. 5.2(1). We are going to show that $S(\sigma_i)$ was modified so that $S^{\mathfrak{g}}_{\alpha,i}(\sigma_i)|_{\mathfrak{g}'} = \perp^i$, which completes the argument as then $S^{\mathfrak{g}}_{\alpha,i}(\sigma_{1-i})|_{\mathfrak{g}'} = \perp^i = S^{\mathfrak{g}}_{\alpha,i}(\sigma_i)|_{\mathfrak{g}'}$, and this contradicts the assumption that $S^{\mathfrak{g}}_{\alpha,i}(\sigma_0)|_{\mathfrak{g}'} \not\leq S^{\mathfrak{g}}_{\alpha,i}(\sigma_1)|_{\mathfrak{g}'}$.

For the proof of $S^{\mathfrak{g}}_{\alpha,i}(\sigma_i)|_{\mathfrak{g}'} = \perp^i$ we observe that form $\sigma_i \leq^i \sigma_{1-i} \in E$, by Prop. 5.1(5) we obtain some positions $\mathfrak{h}'_0, \mathfrak{h}'_1$ and $\mathfrak{h}''_2 \preceq (\mathfrak{g}_1 \cdot \mathfrak{h}_2) \not\swarrow \mathfrak{g}_0$ such that $\langle \sigma_i|_{\mathfrak{g}_0}, \mathfrak{h}'_0, \mathfrak{h}'_1, \mathfrak{h}''_2 \rangle \in (\!|\alpha|_{\mathfrak{g}}|\!)^i$. Since $\mathfrak{g}_0 \cdot \mathfrak{h}''_2 \preceq \mathfrak{g}_0 \cdot ((\mathfrak{g}_1 \cdot \mathfrak{h}_2) \not\swarrow \mathfrak{g}_0) = \mathfrak{g}'$ and as $\mathfrak{g}'$ is accessible in $S^{\mathfrak{g}}_{\alpha,i}(\sigma_i)$, we actually have $\mathfrak{g}_0 \cdot \mathfrak{h}''_2 = \mathfrak{g}'$. By definition of $S^{\mathfrak{g}}_{\alpha,i}$ we obtain therefore that $S^{\mathfrak{g}}_{\alpha,i}(\sigma_i)|_{\mathfrak{g}'} = S^{\mathfrak{g}}_{\alpha,i}(\sigma_i|_{\mathfrak{g}_0})|_{\mathfrak{h}''_2} = \perp^i$.

Note that the set A in Def. 10 cannot contain ε. This is because for each quadruple $\langle \beta, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2 \rangle \in (\!|\alpha|_{\mathfrak{g}}|\!)^i$ we can obtain $\mathfrak{h}_2 \neq \varepsilon$. Indeed, there must be some $\beta' \in \|\alpha|_{\mathfrak{h}_1}\|^i$ and this is impossible for $\mathfrak{h}_0 \succeq \mathfrak{g}$ as $\|\alpha|_{\mathfrak{g}}\|^i \subseteq \{\perp^i\}$. Consequently, the suffix $\mathfrak{g} \not\swarrow \mathfrak{h}_0$ of $\mathfrak{h}_2$ is not empty and so $\mathfrak{h}_2$ itself too. This means that $S^{\mathfrak{g}}_{\alpha,i}$ does not substitute a constant for any additional variable as compared with S. So if S avoids constants then $S^{\mathfrak{g}}_{\alpha,i}$ does too. $\square$

In order to complete our construction we need also some way to make expressions substituted by solutions bigger. In the definition of the second substitution manipulation below we replace a constant $\perp^i$ at a position $\mathfrak{g}$ with an expression of the form $\perp^i \circledcirc \perp^i$. However, once the constant is replaced we also have to replace the same constant with $\perp^i \circledcirc \perp^i$ on all positions that have to be modified to maintain solvability (i.e. ones that were in relation $\geq^i$ with $S(\alpha)|_{\mathfrak{g}}$ and were decorated with $\perp^i$, which means they belong to $(\!|\alpha|_{\mathfrak{g}}|\!)^{1-i}$). This is expressed precisely in the following definition.

Definition 11 `(substitution extending)`
We define the substitution $\hat{S}^{\mathfrak{g}}_{\alpha,i}$ for $i \in \{0,1\}$ as

$$\hat{S}^{\mathfrak{g}}_{\alpha,i}(\beta) = S(\beta)[A \leftarrow \perp^i \circledcirc \perp^i] \text{ where } A = \{\mathfrak{h} \mid (\langle\beta, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}\rangle \in (\!|\alpha|_{\mathfrak{g}}|\!)^{1-i} \text{ and } \mathfrak{h} \text{ is accessible in } S(\beta) \text{ and } S(\beta)|_{\mathfrak{h}} = \perp^i) \text{ or } (\beta = \alpha \text{ and } \mathfrak{h} = \mathfrak{g})\}.$$

Here are some basic properties of the solution extending.

Proposition 5.4 `(basic properties of solution extending)`
Assume that $i \in \{0,1\}$.

1. *If $\beta \neq \alpha$ and $\mathfrak{h}$ is accessible in $S(\beta)$ or $\beta = \alpha$ and $\mathfrak{h} \not\succeq \mathfrak{g}$ is accessible in $S(\beta)$ then $\mathfrak{h}$ is accessible in $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\beta)$.*

2. *If $\beta \neq \alpha$ and $\mathfrak{h}$ is accessible in $S(\beta)$ or if $\beta = \alpha$ and $\mathfrak{h} \not\succeq \mathfrak{g}$ is accessible is $S(\beta)$ then the label $\blacktriangle\hat{S}^{\mathfrak{g}}_{\alpha,i}(\beta)|_{\mathfrak{h}} = \circledcirc$ or $\blacktriangle\hat{S}^{\mathfrak{g}}_{\alpha,i}(\beta)|_{\mathfrak{h}} = \blacktriangle S(\beta)|_{\mathfrak{h}}$.*

3. *For each inequation $\sigma_0 \leq \sigma_1 \in E$ and all positions $\mathfrak{h}$ that are not accessible in $S(\sigma_j)$ for $j \in \{0,1\}$ but are accessible in $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_j)$ it holds that $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_j)|_{\mathfrak{h}} = \perp^i$.*

4. *If $\sigma_0 \leq \sigma_1 \in E$ and we have $S(\sigma_i)|_{\mathfrak{h}} = S(\sigma_{1-i})|_{\mathfrak{h}} = \perp^i$ and $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_i)|_{\mathfrak{h}} = \perp^i \circledcirc \perp^i$ then $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_{1-i})|_{\mathfrak{h}} = \perp^i \circledcirc \perp^i$.*

Proof:
The proofs of cases (1)–(3) are left to the reader.

We provide the proof for the case (4). We observe first that $\sigma_{1-i} \leq^{1-i} \sigma_i \in E$. Assume that $\sigma_i|_{\mathfrak{h}'} = \beta$ and $\sigma_{1-i}|_{\mathfrak{h}''} = \gamma$ where $\beta, \gamma \in \mathrm{FV}(E)$ and $\mathfrak{h}', \mathfrak{h}''$ are prefixes of $\mathfrak{h}$.

We consider two possible subcases depending on the reason why $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_i)|_{\mathfrak{h}} = \perp^i \circledcirc \perp^i$, i.e. when either (i) $\langle\beta, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h} \not\!\!\angle\, \mathfrak{h}'\rangle \in (\!|\alpha|_{\mathfrak{g}}|\!)^{1-i}$ for some $\mathfrak{h}_0, \mathfrak{h}_1$ or (ii) $\beta = \alpha$ and $\mathfrak{h} \not\!\!\angle\, \mathfrak{h}' = \mathfrak{g}$.

For the case (i) we apply Prop. 5.1(4), with $1-i$ playing the role of i in the proposition, and obtain that $\langle\gamma, \mathfrak{h}'_0, \mathfrak{h}'_1, (\mathfrak{h}' \cdot (\mathfrak{h} \not\!\!\angle\, \mathfrak{h}')) \not\!\!\angle\, \mathfrak{h}''\rangle \in [\![\alpha|_{\mathfrak{g}}]\!]^{1-i}$ for some $\mathfrak{h}'_0, \mathfrak{h}'_1$. Observe that $(\mathfrak{h}' \cdot (\mathfrak{h} \not\!\!\angle\, \mathfrak{h}')) \not\!\!\angle\, \mathfrak{h}'' = \mathfrak{h} \not\!\!\angle\, \mathfrak{h}''$. Since $\mathfrak{h}$ is accessible in $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_{1-i})$, we obtain even more, i.e. $\langle\gamma, \mathfrak{h}'_0, \mathfrak{h}'_1, \mathfrak{h} \not\!\!\angle\, \mathfrak{h}''\rangle \in (\!|\alpha|_{\mathfrak{g}}|\!)^{1-i}$. By definition of $\hat{S}^{\mathfrak{g}}_{\alpha,i}$ this results in required $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_{1-i})|_{\mathfrak{h}} = \perp^i \circledcirc \perp^i$.

For the case (ii) we first prove that $\langle\gamma, \mathfrak{h}'', \mathfrak{h}', \mathfrak{h} \not\!\!\angle\, \mathfrak{h}''\rangle \in [\![\alpha|_{\mathfrak{h} \not\angle \mathfrak{h}'}]\!]^{1-i}$. For this we observe that in case $\mathfrak{h}'' = \varepsilon$ we have $\alpha \in \|\alpha|_{\mathfrak{h}''}\|^{i-1}$ and $\alpha \in \|\gamma|_{\mathfrak{h}'}\|^i$ while when $\mathfrak{h}'' \neq \varepsilon$ we have $\gamma \in \|\alpha|_{\mathfrak{h}''}\|^{i-1}$ and $\gamma \in \|\gamma|_{\mathfrak{h}'}\|^i$. In both cases there is some β' such that $\beta' \in \|\alpha|_{\mathfrak{h}''}\|^{i-1}$ and $\beta' \in \|\gamma|_{\mathfrak{h}'}\|^i$. So first two conditions in definition of $(\!|\alpha|_{\mathfrak{h} \not\angle \mathfrak{h}'}|\!)^{1-i}$ hold. We also prove the third one, i.e. $\mathfrak{g} \not\!\!\angle\, \mathfrak{h}'' = (\mathfrak{h} \not\!\!\angle\, \mathfrak{h}'') \not\!\!\angle\, \mathfrak{h}'$. Indeed, $\mathfrak{g} \not\!\!\angle\, \mathfrak{h}'' = (\mathfrak{h} \not\!\!\angle\, \mathfrak{h}') \not\!\!\angle\, \mathfrak{h}''$ and since one of $\mathfrak{h}', \mathfrak{h}''$ is ε we can exchange them and obtain $(\mathfrak{h} \not\!\!\angle\, \mathfrak{h}') \not\!\!\angle\, \mathfrak{h}'' = (\mathfrak{h} \not\!\!\angle\, \mathfrak{h}'') \not\!\!\angle\, \mathfrak{h}'$, which is the required result. These all combine to all requirements of the definition of $\langle\gamma, \mathfrak{h}'', \mathfrak{h}', \mathfrak{h} \not\!\!\angle\, \mathfrak{h}''\rangle \in [\![\alpha|_{\mathfrak{h} \not\angle \mathfrak{h}'}]\!]^{1-i}$. Since $\mathfrak{h} \not\!\!\angle\, \mathfrak{h}''$ is accessible in $S(\gamma)$, we actually have $\langle\gamma, \mathfrak{h}'', \mathfrak{h}', \mathfrak{h} \not\!\!\angle\, \mathfrak{h}''\rangle \in (\!|\alpha|_{\mathfrak{h} \not\angle \mathfrak{h}'}|\!)^{1-i}$. As by assumption, we have $S(\gamma)|_{\mathfrak{h} \not\angle \mathfrak{h}''} = \perp^i$, it turns out that $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\gamma)|_{\mathfrak{h} \not\angle \mathfrak{h}''} = \hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_{1-i})|_{\mathfrak{h}} = \perp^i \circledcirc \perp^i$, which is the required conclusion. $\square$

Application of the substitution defined in Def. 11 to solutions turns them into solutions provided that we extend a position with no disturbing variables in decoration on the appropriate side (i.e. the side i in the formulation below).

Lemma 5.5 `(solution extending is a solution)`
If $\mathfrak{g}$ is a position such that $\|\alpha|_{\mathfrak{g}}\|^i \subseteq \{\perp^i\}$ for some

$i \in \{0,1\}$ *and* $S(\alpha)|_{\mathfrak{g}} = \bot^i$ *then* $\hat{S}^{\mathfrak{g}}_{\alpha,i} \models E$.

Moreover, if S *avoids constants then* $\hat{S}^{\mathfrak{g}}_{\alpha,i}$ *does.*

Proof:
Suppose that $\hat{S}^{\mathfrak{g}}_{\alpha,i}$ does not solve E. In that case there is a pair of expressions σ_0, σ_1 such that $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_0) \not\leq \hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_1)$ even though $\sigma_0 \leq \sigma_1 \in E$. Suppose $\mathfrak{g}'$ is a maximal position such that $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_0)|_{\mathfrak{g}'} \not\leq \hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_1)|_{\mathfrak{g}'}$. This means that at least one of the expressions $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_0)|_{\mathfrak{g}'}, \hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_1)|_{\mathfrak{g}'}$ is a constant. We can exclude the situation that $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_j)|_{\mathfrak{g}'} = \bot^j$ for $j \in \{0,1\}$ since this would make $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_j)|_{\mathfrak{g}'} \not\leq^j \hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_{1-j})|_{\mathfrak{g}'}$ or equivalently $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_0)|_{\mathfrak{g}'} \not\leq \hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_1)|_{\mathfrak{g}'}$ impossible.

Let $j \in \{0,1\}$ be such that (*) $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_j)|_{\mathfrak{g}'} = \bot^{1-j}$. We consider two subcases depending on whether $j = i$.

<u>In case $j = i$</u>, we can present (*) as $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_i)|_{\mathfrak{g}'} = \bot^{1-i}$. Moreover, $S(\sigma_i)|_{\mathfrak{g}'} = \bot^{1-i}$. Indeed $\mathfrak{g}'$ is accessible in $S(\sigma_i)$ as otherwise all positions that are not accessible in $S(\sigma_i)$ but are accessible in $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_i)$ are decorated in $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_i)$ with the label $\bot^i \neq \bot^{1-i}$ (see Prop. 5.4(3)), and this would contradict the condition (*). In addition, $S(\sigma_i)|_{\mathfrak{g}'} = \bot^i$ or $S(\sigma_i)|_{\mathfrak{g}'} = \rho_0 \circledcirc \rho_1$ for some ρ_0, ρ_1 would make (*) impossible either by similar reasoning.

Moreover, for each position $\mathfrak{g}'' \preceq \mathfrak{g}'$ we have $S(\sigma_{1-i})|_{\mathfrak{g}''} \neq \bot^i$ as its negation would mean that $S \not\models \sigma_0 \leq \sigma_1$. The same contradiction we obtain when $S(\sigma_{1-i})|_{\mathfrak{g}'} = \rho_0 \circledcirc \rho_1$ for some ρ_0, ρ_1. The only possibility that remains is that $S(\sigma_{1-i})|_{\mathfrak{g}'} = \bot^{1-i}$. Let $\sigma_{1-i}|_{\mathfrak{h}} = \beta \in \mathrm{FV}(E)$ where $\mathfrak{h} \preceq \mathfrak{g}'$. By definition of $\hat{S}^{\mathfrak{g}}_{\alpha,i}$, we obtain that $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_{1-i})|_{\mathfrak{g}'} = \bot^{1-i}$, which contradicts $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_0)|_{\mathfrak{g}'} \not\leq \hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_1)|_{\mathfrak{g}'}$.

<u>In case $j \neq i$</u> we observe first that $i = 1 - j$. We can rewrite here (*) as $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_{1-i})|_{\mathfrak{g}'} = \bot^i$. In this situation we can eventually obtain $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_i)|_{\mathfrak{g}'} \leq^i \hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_{1-i})|_{\mathfrak{g}'}$, contrary to the assumption. Indeed:

- In case $\mathfrak{g}'$ is accessible in $S(\sigma_i)$ we recall first that $\sigma_i \leq^i \sigma_{1-i} \in E$. We can obtain now that $\mathfrak{g}'$ is accessible in $S(\sigma_{1-i})$ too. Indeed, if it is not accessible in $S(\sigma_{1-i})$ then $S(\sigma_{1-i})|_{\mathfrak{g}''} = \bot^i$ where $\mathfrak{g}'' \cdot k = \mathfrak{g}'$ for some $k \in \{0,1\}$ as only this can make $\mathfrak{g}'$ accessible in $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_{1-i})$. However, $S \models E$ implies in this situation $\bot^i = S(\sigma_{1-i})|_{\mathfrak{g}''} \leq^{1-i} S(\sigma_i)|_{\mathfrak{g}''}$ ($\mathfrak{g}''$ is accessible in $S(\sigma_i)$ as $\mathfrak{g}'$ is). Consequently, $S(\sigma_i)|_{\mathfrak{g}''} = \bot^i$, which contradicts the assumption that $\mathfrak{g}'$ is accessible in $S(\sigma_i)$.

  Since $\mathfrak{g}'$ is accessible both in $S(\sigma_{1-i})$ and in $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_{1-i})$ as well as $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_{1-i})|_{\mathfrak{g}'} = \bot^i$, we obtain that $S(\sigma_{1-i})|_{\mathfrak{g}'} = \bot^i$. Since $\sigma_i \leq^i \sigma_{1-i} \in E$, we obtain further that $S(\sigma_i)|_{\mathfrak{g}'} = S(\sigma_{1-i})|_{\mathfrak{g}'} = \bot^i$.

  If we show in addition that $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_i)|_{\mathfrak{g}'} = \bot^i$ then we reach contradiction with $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_0)|_{\mathfrak{g}'} \not\leq \hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_1)|_{\mathfrak{g}'}$ as both sides are equal to $\bot^i$.

  To show the requested $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_i)|_{\mathfrak{g}'} = \bot^i$ we proceed by contradiction. Suppose that $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_i)|_{\mathfrak{g}'} \neq \bot^i$. As $S(\sigma_i)|_{\mathfrak{g}'} = \bot^i$, the equality $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_i)|_{\mathfrak{g}'} = \bot^i \circledcirc \bot^i$ must hold. We can now apply Prop. 5.4(4) and obtain that $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_{1-i})|_{\mathfrak{g}'} = \bot^i \circledcirc \bot^i$, which is in contradiction with condition (*).

- In case $\mathfrak{g}'$ is not accessible in $S(\sigma_i)$, but is in $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_i)$, we must have $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_i)|_{\mathfrak{g}'} = \bot^i$ by definition of $\hat{S}^{\mathfrak{g}}_{\alpha,i}$. However, this means that $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_{1-i})|_{\mathfrak{g}'} = \bot^i \leq^i \bot^i = \hat{S}^{\mathfrak{g}}_{\alpha,i}(\sigma_i)|_{\mathfrak{g}'}$, which contradicts the assumption on the inequation.

Observe that if $A \neq \emptyset$ and some $\mathfrak{g} \in A$ is accessible in τ then $\tau[A \leftarrow \bot^i \circledcirc \bot^i] \notin \{\bot, \top\}$ so if for each $\beta \in \mathrm{FV}(E)$ we have $S(\beta) \notin \{\bot, \top\}$ then also for each $\beta \in \mathrm{FV}(E)$ we have $\hat{S}^{\mathfrak{g}}_{\alpha,i}(\beta) \notin \{\bot, \top\}$. □

We can iterate the extension as in the above lemma and obtain under certain conditions a solution in which a particular path is accessible.

Proposition 5.6 `(extending solution towards a position)`
Let $\alpha_0, \alpha_1 \in \mathrm{FV}(E)$ *and* S *avoids constants. If* $\mathfrak{g}$ *is the maximal position accessible in both* $S(\alpha_0)$ *and* $S(\alpha_1)$ *and* $\mathfrak{g} \preceq \mathfrak{g}'$ *then for each* $\mathfrak{h}$ *such that* $\mathfrak{g} \preceq \mathfrak{h} \preceq \mathfrak{g}'$ *there is a substitution* $S_{\mathfrak{h}}$ *such that* $S_{\mathfrak{h}} \models E$ *and* $\mathfrak{h}$ *is accessible both in* $S_{\mathfrak{h}}(\alpha_0)$ *and in* $S_{\mathfrak{h}}(\alpha_1)$.

Moreover, each position accessible in $S(\beta)$ for $\beta \in \mathrm{FV}(E)$ is accessible in $S_{\mathfrak{h}}(\beta)$ and $S_{\mathfrak{h}}$ avoids constants.

Proof:
The proof is by induction over $n = |\mathfrak{h}| - |\mathfrak{g}|$. The case of $n = 0$ follows immediately by the assumptions of the current proposition.

For the case of $n = |\mathfrak{h}| - |\mathfrak{g}| > 0$, we obtain by the induction hypothesis $\mathfrak{h}_{n-1}$ such that $\mathfrak{h} = \mathfrak{h}_{n-1} \cdot k$ for some $k \in \{0, 1\}$ and $\mathfrak{g} \preceq \mathfrak{h}_{n-1} \preceq \mathfrak{g}'$ and $|\mathfrak{h}_{n-1}| - |\mathfrak{g}| = n - 1$ as well as a substitution T such that $T \models E$ and $\mathfrak{h}_{n-1}$ is accessible in both $T(\alpha_0)$ and $T(\alpha_1)$. If $\mathfrak{h}$ is accessible in both $T(\alpha_0)$ and $T(\alpha_1)$ then T suffices for our induction step. If $\mathfrak{h}$ is not accessible in $T(\alpha_0)$, but accessible in $T(\alpha_1)$ then we observe that $T(\alpha_0)|_{\mathfrak{h}_{n-1}} = \perp^k$ for some $k \in \{0, 1\}$ as $\mathfrak{h}$ is not accessible in $T(\alpha_0)$. Since S avoids constants and respects the static decoration, we obtain that $\|\alpha_0|_{\mathfrak{h}_{n-1}}\|^i \subseteq \{\perp^k\}$. Observe that it means that $\|\alpha_0|_{\mathfrak{h}_{n-1}}\|^k \subseteq \{\perp^k\}$, as by the induction hypothesis $T(\beta) \notin \{\perp, \top\}$ for $\beta \in \mathrm{dom}(T)$. We can now take $S_{\mathfrak{h}} = \hat{T}^{\mathfrak{h}}_{\alpha_0,k}$ and by definition of the substitution, $\mathfrak{h}$ is accessible both in $S_{\mathfrak{h}}(\alpha_0)$ and $S_{\mathfrak{h}}(\alpha_1)$. Moreover, Lemma 5.5 implies that $S_{\mathfrak{h}} \models E$ and $S_{\mathfrak{h}}(\beta) \notin \{\perp, \top\}$ for all $\beta \in \mathrm{FV}(E)$. If $\mathfrak{h}$ is not accessible in $T(\alpha_1)$, but accessible in $T(\alpha_0)$ then we use the symmetric argument for $S_{\mathfrak{h}} = \hat{T}^{\mathfrak{h}}_{\alpha_1,k}$ and some $k \in \{0, 1\}$. At last when $\mathfrak{h}$ is not accessible both in $T(\alpha_0)$ and in $T(\alpha_1)$ then we can use similar argument to obtain the conclusion for $S_{\mathfrak{h}} = \hat{T'}^{\mathfrak{h}}_{\alpha_1,k'}$ where $T' = \hat{T}^{\mathfrak{h}}_{\alpha_0,k''}$ for appropriate $k', k'' \in \{0, 1\}$.

By Prop. 5.4(1) for any variable $\beta \in \mathrm{FV}(E)$ all positions accessible in $S(\beta)$ are also accessible in all solutions mentioned in the proof above.

Moreover, substitutions used in this proof only extend the expressions in the image of S so that each position accessible in $S(\beta)$ for $\beta \in \mathrm{FV}(E)$ is accessible in $S_{\mathfrak{h}}(\beta)$ and T' avoids constants. □

Recall that restriction from this proposition that S avoids constants is rather mild as observed in Section 3.3.

Note that all the proofs in this section do not depend on the fact that variables are substituted by finite nor infinite type expressions. Therefore, all the statements of the propositions hold in both cases.

6 Pumping lemma and modest solutions

We turn our focus now to inentailment problems with inputs of the form E, $\alpha_0 \leq \alpha_1$. We would like to use operations on substitutions that were defined so far to transform an original solution S to this problem into some S' with a critical position of computable size. However, we have to be careful in choosing the candidate for the new such position. If we choose some $\mathfrak{g}$ so that $\mathfrak{g}$ is accessible in the new $S'(\alpha_0)$ and such that $S'(\alpha_0)|_{\mathfrak{g}}$ is a constant then $\mathfrak{g}$ in $S'(\alpha_1)$ must be accessible too for this position to be critical. That is why we have to cautiously ensure that our cutting of the original substitution does not break accessibility we need. We can express the necessary requirements using the following definition.

Definition 12 `(shortening cuts)`
We define the set of *semi-shortenings resulting* at $\mathfrak{g}$ as

$$\curlyvee_i^{\beta_0 \rightsquigarrow \beta_1}(\mathfrak{g}) = \{\mathfrak{h}_2 \preceq \mathfrak{g} \mid \exists \langle \beta_1, \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}_2 \rangle \in [\![\beta_0|_{\mathfrak{g}}]\!]^i\}.$$

We use the following generalisation of this notion $\curlyvee_i^{\beta_0 \rightsquigarrow A}(\mathfrak{g}) = \bigcup_{\beta \in A} \curlyvee_i^{\beta_0 \rightsquigarrow \beta}(\mathfrak{g})$.

It is important to observe here that the notion of semi-shortenings does not depend on any particular substitution or solution.

It should be obvious that sets $\curlyvee_i^{\beta_0 \rightsquigarrow \beta_1}(\mathfrak{g})$ are finite. Let us make another simple observation that if $\curlyvee_i^{\beta_0 \rightsquigarrow \beta_1}(\mathfrak{g}) = \emptyset$ and $\mathfrak{g}$ is accessible in $S(\beta_1)$ then it is also accessible in $S^{\mathfrak{g}}_{\beta_0,i}(\beta_1)$ and $\blacktriangle S(\beta_1)|_{\mathfrak{g}} = \blacktriangle S^{\mathfrak{g}}_{\beta_0,i}(\beta_1)|_{\mathfrak{g}}$. We note also that for all $\beta, \mathfrak{g}, i$, we have $\curlyvee_i^{\beta \rightsquigarrow \beta}(\mathfrak{g}) = \{\mathfrak{g}\}$.

Assumption To enable application of Prop. 5.6, we assume that S avoids constants for the rest of Section 6.

Proposition 6.1 `(shortenings shrink)`
Assume that for some $i \in \{0, 1\}$ and variable $\beta \in \mathrm{FV}(E)$, we have $\|\beta|_{\mathfrak{g}}\|^i \subseteq \{\perp^i\}$. Let $A \subseteq \mathrm{FV}(E)$. If $\curlyvee_i^{\beta \rightsquigarrow A}(\mathfrak{g}) \not\subseteq \{\mathfrak{g}\}$ then there is $j \in \{0, 1\}$ such that

$$|\curlyvee_i^{\beta \rightsquigarrow A}(\mathfrak{g} \cdot j)| < |\curlyvee_i^{\beta \rightsquigarrow A}(\mathfrak{g})|$$

and if $\Upsilon_i^{\beta\rightsquigarrow A}(\mathfrak{g}) \subseteq \{\mathfrak{g}\}$ *then for* $j \in \{0,1\}$

$$|\ \Upsilon_i^{\beta\rightsquigarrow A}\ (\mathfrak{g}\cdot j)| = |\ \Upsilon_i^{\beta\rightsquigarrow A}\ (\mathfrak{g})|.$$

Proof:
We define now a 1-1 function $f : \Upsilon_i^{\beta\rightsquigarrow A}(\mathfrak{g}\cdot j) \to \Upsilon_i^{\beta\rightsquigarrow A}(\mathfrak{g})$ for both $j \in \{0,1\}$. The function is defined as $f(\mathfrak{h}) = \mathfrak{h} \nearrow j$. If this is a well defined function then it is obviously 1-1. It remains to prove that this function is well defined.

Consider any $\mathfrak{h} \in \Upsilon_i^{\beta\rightsquigarrow A}(\mathfrak{g}\cdot j)$. This means that for some $\beta' \in A$ and $\mathfrak{h}_0, \mathfrak{h}_1$ there is $\langle \beta', \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}\rangle \in [\![\beta|_{\mathfrak{g}\cdot j}]\!]^i$ such that $\mathfrak{h} \preceq \mathfrak{g}\cdot j$.

We observe first that $\mathfrak{h} \nearrow j$ is well defined. Indeed, the definition of $[\![\beta|_{\mathfrak{g}\cdot j}]\!]^i$ implies that $\mathfrak{g}\cdot j \swarrow \mathfrak{h}_0 = \mathfrak{h} \swarrow \mathfrak{h}_1$ as well as that $\|\beta|_{\mathfrak{h}_0}\|^i$ contains a variable. When $\mathfrak{h}_0 = \mathfrak{g}\cdot j$, we immediately obtain that $\|\beta|_{\mathfrak{h}_0}\|^i$, contrary to the fact that $\langle \beta', \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h}\rangle \in [\![\beta|_{\mathfrak{g}\cdot j}]\!]^i$, does not contain a variable as $\|\beta|_{\mathfrak{g}}\|^i$ does not. Consequently, $\mathfrak{g}\cdot j \swarrow \mathfrak{h}_0 = \mathfrak{h} \swarrow \mathfrak{h}_1$ must end with j and then obviously $\mathfrak{h}$, too.

It remains to show that $\mathfrak{h} \nearrow j \in \Upsilon_i^{\beta\rightsquigarrow A}(\mathfrak{g})$. For this, we show that $\langle \beta', \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h} \nearrow j\rangle \in [\![\beta|_{\mathfrak{g}}]\!]^i$. From the definition of $[\![\beta|_{\mathfrak{g}\cdot j}]\!]^i$ we obtain that there is a variable β'' such that (i) $\beta'' \in \|\beta|_{\mathfrak{h}_0}\|^i$, (ii) $\beta'' \in \|\beta'|_{\mathfrak{h}_1}\|^{1-i}$, (iii) $\mathfrak{g}\cdot j \swarrow \mathfrak{h}_0 = \mathfrak{h} \swarrow \mathfrak{h}_1$. We use the same β'' as the witness that $\langle \beta', \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{h} \nearrow j\rangle \in [\![\beta|_{\mathfrak{g}}]\!]^i$ and check the conditions that make this statement hold. The conditions (i) and (ii) above are identical as first two conditions in the definition of $[\![\beta|_{\mathfrak{g}}]\!]^i$. The third condition follows as

$$\mathfrak{g} \swarrow \mathfrak{h}_0 = ((\mathfrak{g} \swarrow \mathfrak{h}_0)\cdot j) \nearrow j = (\mathfrak{g}\cdot j \swarrow \mathfrak{h}_0) \nearrow j \overset{\text{(iii)}}{=} (\mathfrak{h} \swarrow \mathfrak{h}_1) \nearrow j = (\mathfrak{h} \nearrow j) \swarrow \mathfrak{h}_1.$$

This proves for both $j = 0$ and 1 that

$$|\ \Upsilon_i^{\beta\rightsquigarrow A}\ (\mathfrak{g}\cdot j)| \leq |\ \Upsilon_i^{\beta\rightsquigarrow A}\ (\mathfrak{g})|.$$

It remains to prove that when $\Upsilon_i^{\beta\rightsquigarrow A}(\mathfrak{g}) \not\subseteq \{\mathfrak{g}\}$ for one of $j \in \{0,1\}$, this inequality is strict. Suppose that for some $j \in \{0,1\}$ we have $|\ \Upsilon_i^{\beta\rightsquigarrow A}\ (\mathfrak{g}\cdot j)| = |\ \Upsilon_i^{\beta\rightsquigarrow A}(\mathfrak{g})|$. Observe now that for each $\mathfrak{h} \in \Upsilon_i^{\beta\rightsquigarrow A}(\mathfrak{g})$ we have $\mathfrak{h}\cdot j \preceq \mathfrak{g}\cdot j$. This implies that for all $\mathfrak{h} \neq \mathfrak{g}$ and $\mathfrak{h} \in \Upsilon_i^{\beta\rightsquigarrow A}(\mathfrak{g})$ we have $\mathfrak{h}\cdot(1-j) \not\preceq \mathfrak{g}\cdot(1-j)$. This means that when there is $\mathfrak{h} \neq \mathfrak{g}$ such that $\mathfrak{h} \in \Upsilon_i^{\beta\rightsquigarrow A}(\mathfrak{g})$ this $\mathfrak{h}$ does not belong to the image of the function $f : \Upsilon_i^{\beta\rightsquigarrow A}(\mathfrak{g}\cdot(1-j)) \to \Upsilon_i^{\beta\rightsquigarrow A}(\mathfrak{g})$ defined as $f(\mathfrak{h}) = \mathfrak{h} \nearrow (1-j)$. Consequently, f is 1-1, but is not an epimorphism so $|\ \Upsilon_i^{\beta\rightsquigarrow A}(\mathfrak{g}\cdot(1-j))| < |\ \Upsilon_i^{\beta\rightsquigarrow A}(\mathfrak{g})|$.

Note that when $\Upsilon_i^{\beta\rightsquigarrow A}(\mathfrak{g}) = \emptyset$ we also have $\Upsilon_i^{\beta\rightsquigarrow A}(\mathfrak{g}\cdot j) = \emptyset$ for $j \in \{0,1\}$ as f defined above would not be well defined otherwise. In case $\Upsilon_i^{\beta\rightsquigarrow A}(\mathfrak{g}) = \{\mathfrak{g}\}$, there is some $\beta' \in A$ such that $\langle \beta', \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{g}\rangle \in [\![\beta|_{\mathfrak{g}}]\!]^i$, but then of course $\langle \beta', \mathfrak{h}_0, \mathfrak{h}_1, \mathfrak{g}\cdot j\rangle \in [\![\beta|_{\mathfrak{g}\cdot j}]\!]^i$. This means $\mathfrak{g}\cdot j \in \Upsilon_i^{\beta\rightsquigarrow A}(\mathfrak{g}\cdot j)$ and as f is 1-1 this is the only member of the set. Consequently, when $\Upsilon_i^{\beta\rightsquigarrow A}(\mathfrak{g}) \subseteq \{\mathfrak{g}\}$ we have $|\ \Upsilon_i^{\beta\rightsquigarrow A}\ (\mathfrak{g}\cdot j)| = |\ \Upsilon_i^{\beta\rightsquigarrow A}\ (\mathfrak{g})|$, which concludes the proof. □

Note that the proof above does not depend on any solution of E.

Proposition 6.2 (`immediate fixing position with a constant`)
Let $\alpha_0, \alpha_1 \in \mathrm{FV}(E)$. *Assume that for some* $i \in \{0,1\}$, *we have* $\|\alpha_0|_{\mathfrak{g}}\|^i \subseteq \{\perp^i\}$. *If* $\mathfrak{g}\cdot\mathfrak{g}'$ *is accessible in* $S(\alpha_0)$ *and* $S(\alpha_1)$ *as well as* $\Upsilon_i^{\alpha_0\rightsquigarrow\alpha_0}(\mathfrak{g}\cdot\mathfrak{g}') = \{\mathfrak{g}\cdot\mathfrak{g}'\}$ *and* $\Upsilon_i^{\alpha_0\rightsquigarrow\alpha_1}(\mathfrak{g}\cdot\mathfrak{g}') = \emptyset$ *then there is a substitution* S' *such that*

- $S' \models E$*;*
- $\mathfrak{g}\cdot\mathfrak{g}'$ *is accessible in* $S'(\alpha_0)$*;*
- $S'(\alpha_0)|_{\mathfrak{g}\cdot\mathfrak{g}'} = \perp^i$*;*
- $\mathfrak{g}\cdot\mathfrak{g}'$ *is accessible in* $S'(\alpha_1)$*,; and*
- *the main symbols* $\blacktriangle S'(\alpha_1)|_{\mathfrak{g}\cdot\mathfrak{g}'}$ *and* $\blacktriangle S(\alpha_1)|_{\mathfrak{g}\cdot\mathfrak{g}'}$ *are equal.*

Moreover, S' *avoids constants.*

Proof:
Consider $S_{\alpha_0,i}^{\mathfrak{g}\cdot\mathfrak{g}'}$. By Lemma 5.3, we obtain $S_{\alpha_0,i}^{\mathfrak{g}\cdot\mathfrak{g}'} \models E$. The position $\mathfrak{g}\cdot\mathfrak{g}'$ is accessible in $S_{\alpha_0,i}^{\mathfrak{g}\cdot\mathfrak{g}'}(\alpha_0)$ as $\Upsilon_i^{\alpha_0\rightsquigarrow\alpha_0}(\mathfrak{g}\cdot\mathfrak{g}') = \{\mathfrak{g}\cdot\mathfrak{g}'\}$. At last $S_{\alpha_0,i}^{\mathfrak{g}\cdot\mathfrak{g}'}(\alpha_0)|_{\mathfrak{g}\cdot\mathfrak{g}'} = \perp^i$ by definition of $S_{\alpha_0,i}^{\mathfrak{g}\cdot\mathfrak{g}'}$. We prove by contradiction that $\mathfrak{g}\cdot\mathfrak{g}'$ is accessible in $S_{\alpha_0,i}^{\mathfrak{g}\cdot\mathfrak{g}'}(\alpha_1)$ and that $\blacktriangle S_{\alpha_0,i}^{\mathfrak{g}\cdot\mathfrak{g}'}(\alpha_1)|_{\mathfrak{g}\cdot\mathfrak{g}'} = \blacktriangle S(\alpha_1)|_{\mathfrak{g}\cdot\mathfrak{g}'}$. For this, assume that

$\mathfrak{g}\cdot\mathfrak{g}'$ is not accessible in $S^{\mathfrak{g}\cdot\mathfrak{g}'}_{\alpha_0,i}(\alpha_1)$ or $\mathfrak{g}\cdot\mathfrak{g}'$ is accessible in $S^{\mathfrak{g}\cdot\mathfrak{g}'}_{\alpha_0,i}(\alpha_1)$, but $\blacktriangle S^{\mathfrak{g}\cdot\mathfrak{g}'}_{\alpha_0,i}(\alpha_1)|_{\mathfrak{g}\cdot\mathfrak{g}'} \neq \blacktriangle S(\alpha_1)|_{\mathfrak{g}\cdot\mathfrak{g}'}$. In each of the cases we obtain a position $\mathfrak{g}'' \preceq \mathfrak{g}\cdot\mathfrak{g}'$, being the maximal prefix of $\mathfrak{g}\cdot\mathfrak{g}'$ accessible in $S^{\mathfrak{g}\cdot\mathfrak{g}'}_{\alpha_0,i}(\alpha_1)$, but such that by definition of $S^{\mathfrak{g}\cdot\mathfrak{g}'}_{\alpha_0,i}$ there is $\langle\alpha_1,\mathfrak{h}_1,\mathfrak{h}_2,\mathfrak{g}''\rangle \in (\!|\alpha_0|_{\mathfrak{g}\cdot\mathfrak{g}'}|\!)^i$ for some $\mathfrak{h}_1,\mathfrak{h}_2$. Thus $\mathfrak{g}'' \in \Upsilon^{\alpha_0\leadsto\alpha_1}_i(\mathfrak{g}\cdot\mathfrak{g}')$, which is in contradiction with the assumption that this set of semi-shortenings is empty.

Note that $S^{\mathfrak{g}\cdot\mathfrak{g}'}_{\alpha_0,i}$ avoids constants as $\|\alpha_0|_{\mathfrak{g}\cdot\mathfrak{g}'}\|^i \subseteq \{\perp^i\}$. □

Lemma 6.3 `(shortening minimisation)`
Assume that $\alpha_0,\alpha_1 \in \mathrm{FV}(E)$,

- *$\mathfrak{g}$ is accessible in $S(\alpha_0)$ and $S(\alpha_1)$;*
- *for each $\mathfrak{h} \preceq \mathfrak{g}$ there is no $\beta' \in \mathrm{FV}(E)$ such that $\beta' \in \|\alpha_0|_{\mathfrak{h}}\|^i$ and $\beta' \in \|\alpha_1|_{\mathfrak{h}}\|^{1-i}$;*
- *$\|\alpha_0|_{\mathfrak{g}}\|^i \subseteq \{\perp^i\}$.*

Then there is a substitution S' and a position $\mathfrak{g}' \succeq \mathfrak{g}$ such that

1. *$S' \models E$;*
2. *$\Upsilon^{\alpha_0\leadsto\alpha_0}_i(\mathfrak{g}') = \{\mathfrak{g}'\}$ and $\Upsilon^{\alpha_0\leadsto\alpha_1}_i(\mathfrak{g}') = \emptyset$;*
3. *$\mathfrak{g}'$ is accessible in $S'(\alpha_0)$ and $S'(\alpha_1)$;*
4. *$S'(\alpha_0)|_{\mathfrak{g}'} = \perp^i$ and $\blacktriangle S'(\alpha_1)|_{\mathfrak{g}'} = \circledcirc$;*
5. *$|\mathfrak{g}' \not\preceq \mathfrak{g}| \leq |\mathfrak{g}|$.*

Moreover, S' avoids constants.

Proof:
For each position $\mathfrak{h}'\cdot 0$ where $\mathfrak{g} \preceq \mathfrak{h}'$ we consider solutions $T_{\mathfrak{h}'\cdot 0}$ of E such that $\mathfrak{h}'\cdot 0$ is accessible in both $T_{\mathfrak{h}'\cdot 0}(\alpha_0)$ and $T_{\mathfrak{h}'\cdot 0}(\alpha_1)$. These exist and satisfy $T_{\mathfrak{h}'\cdot 0} \models E$ by Prop. 5.6.

We choose a minimal sequence of positions $\mathfrak{h}_0 = \varepsilon \prec \mathfrak{h}_1 \prec \cdots \prec \mathfrak{h}_n$ where $|\mathfrak{h}_n| \leq |\mathfrak{g}|$ and $|\mathfrak{h}_k|+1 = |\mathfrak{h}_{k+1}|$ and $|\Upsilon^{\alpha_0\leadsto A}_i(\mathfrak{g}\cdot\mathfrak{h}_k)| > |\Upsilon^{\alpha_0\leadsto A}_i(\mathfrak{g}\cdot\mathfrak{h}_{k+1})| \geq 1$ where $A = \{\alpha_0,\alpha_1\}$ and $|\Upsilon^{\alpha_0\leadsto A}_i(\mathfrak{g}\cdot\mathfrak{h}_n)| = 1$. This is possible by Prop. 6.1. This implies that $\Upsilon^{\alpha_0\leadsto\alpha_0}_i(\mathfrak{g}\cdot\mathfrak{h}_n) = \{\mathfrak{g}\cdot\mathfrak{h}_n\}$. In addition, we have either (*) $\Upsilon^{\alpha_0\leadsto\alpha_1}_i(\mathfrak{g}\cdot\mathfrak{h}_n) = \emptyset$ or (**) $\Upsilon^{\alpha_0\leadsto\alpha_1}_i(\mathfrak{g}\cdot\mathfrak{h}_n) = \{\mathfrak{g}\cdot\mathfrak{h}_n\}$. If we assume (*) then we can take $S' = (T')^{\mathfrak{g}\cdot\mathfrak{h}_n}_{\alpha_0,i}$ where $T' = T_{\mathfrak{g}\cdot\mathfrak{h}_n\cdot 0}$ together with $\mathfrak{g}' = \mathfrak{g}\cdot\mathfrak{h}_n$ and prove the conditions of the conclusion

1. $S' \models E$ by Prop. 5.6 and Lemma 5.3.
2. $\Upsilon^{\alpha_0\leadsto\alpha_0}_i(\mathfrak{g}\cdot\mathfrak{h}_n) = \{\mathfrak{g}\cdot\mathfrak{h}_n\}$ and $\Upsilon^{\alpha_0\leadsto\alpha_1}_i(\mathfrak{g}\cdot\mathfrak{h}_n) = \emptyset$, by the mentioned above properties of $\mathfrak{g}\cdot\mathfrak{h}_n$ in this case.
3. $\mathfrak{g}'$ is accessible in $S'(\alpha_0)$ and $S'(\alpha_1)$, by the choice of T' and as $\Upsilon^{\alpha_0\leadsto A}_i(\mathfrak{g}\cdot\mathfrak{h}_n) = \{\mathfrak{g}\cdot\mathfrak{h}_n\}$.
4. $S'(\alpha_0)|_{\mathfrak{g}'} = \perp^i$ and $\blacktriangle S'(\alpha_1)|_{\mathfrak{g}'} = \circledcirc$ holds since $\blacktriangle T'(\alpha_1)|_{\mathfrak{g}'} = \blacktriangle T'(\alpha_0)|_{\mathfrak{g}'} = \circledcirc$ and by definition of $S' = (T')^{\mathfrak{g}\cdot\mathfrak{h}_n}_{\alpha_0,i}$. Note that $\blacktriangle S'(\alpha_1)|_{\mathfrak{g}'} = \circledcirc$ results from the fact that there is no $\langle\alpha_1,\mathfrak{h}_0,\mathfrak{h}_1,\mathfrak{g}'\rangle \in (\!|\alpha_0|_{\mathfrak{g}'}|\!)^i$ as $\Upsilon^{\alpha_0\leadsto\alpha_1}_i(\mathfrak{g}\cdot\mathfrak{h}_n) = \emptyset$.
5. $|\mathfrak{g}' \not\preceq \mathfrak{g}| \leq |\mathfrak{g}|$ as $|\mathfrak{h}_n| \leq |\mathfrak{g}|$ above.

If we assume (**) then this is possible by definition only when there is $\langle\alpha_1,\mathfrak{h}',\mathfrak{h}',\mathfrak{h}_2\rangle \in [\![\alpha_0|_{\mathfrak{g}\cdot\mathfrak{h}_n}]\!]^i$ for some $\mathfrak{h}',\mathfrak{h}_2$. Since $\|\alpha_0|_{\mathfrak{g}}\|^i \subseteq \{\perp^i\}$, this is possible only for $\mathfrak{h}' \prec \mathfrak{g}$. By definition of dynamic decoration we obtain that there is $\beta' \in \|\alpha_0|_{\mathfrak{h}_0}\|^i$ and $\beta' \in \|\alpha_1|_{\mathfrak{h}_0}\|^{1-i}$. However this is impossible by the assumptions of the current lemma. □

Lemma 6.4 `(pumping out lemma)`
Assume that $\alpha_0,\alpha_1 \in \mathrm{FV}(E)$ and

- *$\mathfrak{g} = \mathfrak{g}_1\cdot\mathfrak{g}_2\cdot\mathfrak{g}_3\cdot\mathfrak{g}_4$ is a position such that $|\mathfrak{g}_1| > n_E$ with $\|\alpha_k|_{\mathfrak{g}_1}\|^j = \|\alpha_k|_{\mathfrak{g}_1\cdot\mathfrak{g}_2}\|^j$ for all $j \in \{0,1\}$ and $k \in \{0,1\}$;*
- *$\mathfrak{g}$ is accessible in $S(\alpha_0)$ and $S(\alpha_1)$;*
- *$\Upsilon^{\alpha_0\leadsto\alpha_0}_i(\mathfrak{g}) = \{\mathfrak{g}\}$ and $\Upsilon^{\alpha_0\leadsto\alpha_1}_i(\mathfrak{g}) = \emptyset$;*
- *$S(\alpha_0)|_{\mathfrak{g}} = \perp^i$ and $S(\alpha_1)|_{\mathfrak{g}} \neq \perp^i$;*
- *$\|\alpha_0|_{\mathfrak{g}_1\cdot\mathfrak{g}_2\cdot\mathfrak{g}_3}\|^i \subseteq \{\perp^i\}$.*

Then there is a substitution S' and a position $\mathfrak{g}'_4$ such that

1. *$S' \models E$;*

2. $\mathfrak{g}_1 \cdot \mathfrak{g}_3 \cdot \mathfrak{g}'_4$ *is accessible in* $S'(\alpha_0)$ *and* $S'(\alpha_1)$*;*

3. $S'(\alpha_0)|_{\mathfrak{g}_1\cdot\mathfrak{g}_3\cdot\mathfrak{g}'_4} = \bot^i$ *and* $\blacktriangle S'(\alpha_1)|_{\mathfrak{g}_1\cdot\mathfrak{g}_3\cdot\mathfrak{g}'_4} = \circledcirc$*;*

4. $\Upsilon_i^{\alpha_0\rightsquigarrow\alpha_0}(\mathfrak{g}_1\cdot\mathfrak{g}_3\cdot\mathfrak{g}'_4) = \{\mathfrak{g}_1\cdot\mathfrak{g}_3\cdot\mathfrak{g}'_4\}$ *and* $\Upsilon_i^{\alpha_0\rightsquigarrow\alpha_1}(\mathfrak{g}_1\cdot\mathfrak{g}_3\cdot\mathfrak{g}'_4) = \emptyset$*;*

5. *and* $|\mathfrak{g}'_4| \leq |\mathfrak{g}_1\cdot\mathfrak{g}_3|$*.*

Moreover, S' *avoids constants.*

Proof:
Consider the position $\mathfrak{g}_1 \cdot \mathfrak{g}_3$. If this position is accessible in $S(\alpha_0)$ and $S(\alpha_1)$ then we take $S_0 = S$, and extend S towards $\mathfrak{g}_1 \cdot \mathfrak{g}_3$ (see Prop. 5.6) and let S_0 be the resulting solution. We obtain

- $\alpha_0, \alpha_1 \in \mathrm{FV}(E)$ and $S_0 \models E$ (either by assumption or by Prop. 5.6).
- $\mathfrak{g}_1 \cdot \mathfrak{g}_3$ is accessible in $S_0(\alpha_0)$ and $S(\alpha_1)$ (either by assumption or by Prop. 5.6).
- For each $\mathfrak{h} \preceq \mathfrak{g}_1 \cdot \mathfrak{g}_3$ there is no $\beta' \in \mathrm{FV}(E)$ such that $\beta' \in \|\alpha_0|_{\mathfrak{h}}\|^i$ and $\beta' \in \|\alpha_1|_{\mathfrak{h}}\|^{1-i}$. This holds since when for some $\mathfrak{h}' \preceq \mathfrak{g}_1\cdot\mathfrak{g}_3$ it does not, we can obtain $\mathfrak{h}'' \preceq \mathfrak{g}_1\cdot\mathfrak{g}_2\cdot\mathfrak{g}_3$ that has the same static decorations as static decorations at $\mathfrak{h}'$ (i.e. $\langle\|\alpha_j|_{\mathfrak{h}''}\|^0, \|\alpha_j|_{\mathfrak{h}''}\|^1\rangle = \langle\|\alpha_j|_{\mathfrak{h}'}\|^0, \|\alpha_j|_{\mathfrak{h}'}\|^1\rangle$ for both $j \in \{0,1\}$). From this we immediately obtain $\langle\alpha_1, \mathfrak{h}'', \mathfrak{h}'', \mathfrak{g}\rangle \in (\!|\alpha_0|_{\mathfrak{g}}|\!)^i$. This implies that $\mathfrak{g} \in \Upsilon_i^{\alpha_0\rightsquigarrow\alpha_1}(\mathfrak{g})$ which contradicts one of the assumptions.
- $\|\alpha_0|_{\mathfrak{g}_1\cdot\mathfrak{g}_3}\|^i \subseteq \{\bot^i\}$ since $\|\alpha_0|_{\mathfrak{g}_1\cdot\mathfrak{g}_2\cdot\mathfrak{g}_3}\|^i \subseteq \{\bot^i\}$ and decorations at $\mathfrak{g}_1$ are the same as at $\mathfrak{g}_1\cdot\mathfrak{g}_2$.
- S_0 avoids constants as S does (either by assumption or by Prop. 5.6).

We can now apply the shortening lemma above (Lemma 6.3) and obtain all the required conclusions. □

It may be interesting to observe that the constructions in the proof above can be slightly generalised so that not only the segment $\mathfrak{g}_2$ of $\mathfrak{g}_1\cdot\mathfrak{g}_2\cdot\mathfrak{g}_3\cdot\mathfrak{g}_4$ is pumped out, but also to pump this segment in and obtain positions of the form $\mathfrak{g}_1\cdot(\mathfrak{g}_2)^k\cdot\mathfrak{g}_3\cdot\mathfrak{g}'_4$ for any k and appropriate $\mathfrak{g}'_4$.

The constructions in this subsection do not depend on the fact that finite or infinite type expressions are substituted for variables.

6.1 Construction of modest solutions

The pumping from the preceding section is the basic block which can serve to construct solutions of E that are of computable size and that retain the property that they do not solve $\alpha_0 \leq \alpha_1$. This step is the only step that goes in a slightly different way for finite type expressions than for infinite ones. We start with presentation of the finite case.

Lemma 6.5 `(short critical position witness, finite)`
Let $\alpha_0, \alpha_1 \in \mathrm{FV}(E)$ *and* $\alpha_0 \leq \alpha_1 \notin E$*.*

Assume that S *avoids constants and* $\mathfrak{g}$ *is a position such that* $S(\alpha_i)|_{\mathfrak{g}} = \bot^{1-i}$ *and* $\mathfrak{g}$ *is accessible in* $S(\alpha_{1-i})$ *with* $S(\alpha_{1-i})|_{\mathfrak{g}} \neq \bot^{1-i}$*.*

Then there is a substitution S' *that avoids constants and positions* $\mathfrak{g}', \mathfrak{g}''$ *such that*

1. $S' \models E$*;*
2. $\mathfrak{g}'' \preceq \mathfrak{g}'$*;*
3. $\Upsilon_i^{\alpha_0\rightsquigarrow\alpha_0}(\mathfrak{g}') = \{\mathfrak{g}'\}$ *and* $\Upsilon_i^{\alpha_0\rightsquigarrow\alpha_1}(\mathfrak{g}') = \emptyset$*;*
4. $\|\alpha_i|_{\mathfrak{g}''}\|^{1-i} \subseteq \{\bot^{1-i}\}$*;*
5. $|\mathfrak{g}'| \leq 2n_E + 2^{2(|\mathrm{FV}(E)|+2)+1}$ *and* $|\mathfrak{g}''| \leq 2n_E + 2^{2(|\mathrm{FV}(E)|+2)+1}$*;*
6. $S'(\alpha_i)|_{\mathfrak{g}'} = \bot^{1-i}$ *and* $S'(\alpha_{1-i})|_{\mathfrak{g}'} \neq \bot^{1-i}$*.*

Proof:
Let us first assume that $|\mathfrak{g}| \leq 2n_E + 2^{2(|\mathrm{FV}(E)|+2)+1}$ (5) then we can reach the conclusion of the current lemma for $S' = S$ and $\mathfrak{g}' = \mathfrak{g}$ and $\mathfrak{g}''$ defined below. The places where particular conclusion is reached are marked with numbers in parentheses.

By the assumptions of the lemma we see that $S \models E$ (1) as well as $S(\alpha_i)|_{\mathfrak{g}'} = \bot^{1-i}$ and $S(\alpha_{1-i})|_{\mathfrak{g}'} \neq \bot^{1-i}$ (6). Next, we observe that $\|\alpha_i|_{\mathfrak{g}}\|^{1-i} \subseteq \{\bot^{1-i}\}$. This holds as S avoids constants and $S(\alpha_i)|_{\mathfrak{g}} = \bot^{i-1}$ is possible only when $\|\alpha_i|_{\mathfrak{h}}\|^{1-i} \subseteq \{\bot^{1-i}\}$ for some

$\mathfrak{h} \preceq \mathfrak{g}$. Consider now the shortest $\mathfrak{g}'' \preceq \mathfrak{g}$ (2) such that $\|\alpha_i|_{\mathfrak{g}''}\|^{1-i} \subseteq \{\perp^{1-i}\}$ (4).

We observe now that for each $\mathfrak{h} \preceq \mathfrak{g}''$ there is no $\beta' \in \mathrm{FV}(E)$ such that $\beta' \in \|\alpha_i|_{\mathfrak{h}}\|^{1-i}$ and $\beta' \in \|\alpha_{1-i}|_{\mathfrak{h}}\|^{i}$. If such $\mathfrak{h}$ exists then $\langle \alpha_{1-i}, \mathfrak{h}, \mathfrak{h}, \mathfrak{g}\rangle \in [\![\alpha_i|_{\mathfrak{g}}]\!]^{1-i}$ and this means that $S(\alpha_{1-i})|_{\mathfrak{g}} = \perp^{1-i}$ (see Prop. 5.1(7)), which contradicts the assumptions of the lemma. We can now conclude that $\Upsilon_i^{\alpha_0 \rightsquigarrow \alpha_0}(\mathfrak{g}') = \{\mathfrak{g}'\}$ and $\Upsilon_i^{\alpha_0 \rightsquigarrow \alpha_1}(\mathfrak{g}') = \emptyset$ by Lemma 6.3, which matches the conclusion (3) of the current lemma.

Let us assume that $|\mathfrak{g}| > 2n_E + 2^{2(|\mathrm{FV}(E)|+2)+1}$. We can now decompose $\mathfrak{g}$ as $\mathfrak{g}_1 \cdot \mathfrak{g}_2 \cdot \mathfrak{g}_3 \cdot \mathfrak{g}_4$ so that $|\mathfrak{g}_1| \geq n_E$, with (*) $\|\alpha_k|_{\mathfrak{g}_1}\|^j = \|\alpha_k|_{\mathfrak{g}_1\cdot\mathfrak{g}_2}\|^j$ for all $j = 0,1$ and $k = 0,1$ and (**) $\|\alpha_i|_{\mathfrak{g}_1\cdot\mathfrak{g}_2\cdot\mathfrak{g}_3}\|^j \subseteq \{\perp^{1-i}\}$. Note that (*) is possible since there are more than $(2^{|\mathrm{FV}(E)|+2})^2$ positions $\mathfrak{h} \preceq \mathfrak{g}$ longer than n_E such that one of $\|\alpha_0|_{\mathfrak{h}}\|^j$ for $j \in \{0,1\}$ is empty (see Prop. 4.4). Therefore, one pair of the decorations $\langle \|\alpha_0|_{\mathfrak{h}}\|^0, \|\alpha_0|_{\mathfrak{h}}\|^1\rangle$, $\langle \|\alpha_1|_{\mathfrak{h}}\|^0, \|\alpha_1|_{\mathfrak{h}}\|^1\rangle$ must repeat on these positions. Moreover, (**) holds as S avoids constants and $S(\alpha_i)|_{\mathfrak{g}} = \perp^{i-1}$ can hold only when $\|\alpha_i|_{\mathfrak{h}}\|^j \subseteq \{\perp^{1-i}\}$ for some $\mathfrak{h} \preceq \mathfrak{g}$. Note that when $\|\alpha_i|_{\mathfrak{h}}\|^j \subseteq \{\perp^{1-i}\}$ then also $\|\alpha_i|_{\mathfrak{h}'}\|^j \subseteq \{\perp^{1-i}\}$ for all $\mathfrak{h}' \succ \mathfrak{h}$. Therefore, we can assume there is $\mathfrak{g}_3$ such that $\mathfrak{g}_1 \cdot \mathfrak{g}_2 \cdot \mathfrak{g}_3 \preceq \mathfrak{g}$ and $\|\alpha_i|_{\mathfrak{g}_1\cdot\mathfrak{g}_2\cdot\mathfrak{g}_3}\|^j \subseteq \{\perp^{1-i}\}$.

We observe next that $\Upsilon_{1-i}^{\alpha_i \rightsquigarrow \alpha_{1-i}}(\mathfrak{g}) = \emptyset$ and $\Upsilon_{1-i}^{\alpha_i \rightsquigarrow \alpha_i}(\mathfrak{g}) = \{\mathfrak{g}\}$. This holds true as otherwise it would be impossible to have $S(\alpha_i)|_{\mathfrak{g}} = \perp^{1-i}$ and $S(\alpha_{1-i})|_{\mathfrak{g}} \neq \perp^{1-i}$. As all the assumptions are fulfilled, we can apply Lemma 6.4 and obtain a substitution $\hat{S}$ (that avoids constants) and a position $\mathfrak{g}_4'$ such that $\hat{S} \models E$ ((1)), $\hat{S}(\alpha_0)|_{\mathfrak{g}_1\cdot\mathfrak{g}_3\cdot\mathfrak{g}_4'} = \perp^i$ and $\mathfrak{g}_1\cdot\mathfrak{g}_3\cdot\mathfrak{g}_4'$ is accessible in $\hat{S}(\alpha_1)$ with $\blacktriangle\hat{S}(\alpha_1)|_{\mathfrak{g}_1\cdot\mathfrak{g}_3\cdot\mathfrak{g}_4'} = \circledcirc$ ((6)), and $|\mathfrak{g}_4'| \leq |\mathfrak{g}_1 \cdot \mathfrak{g}_3|$. The same lemma implies that $\Upsilon_i^{\alpha_i \rightsquigarrow \alpha_i}(\mathfrak{g}_1 \cdot \mathfrak{g}_3 \cdot \mathfrak{g}_4') = \{\mathfrak{g}_1 \cdot \mathfrak{g}_3 \cdot \mathfrak{g}_4'\}$ and $\Upsilon_i^{\alpha_i \rightsquigarrow \alpha_{1-1}}(\mathfrak{g}_1 \cdot \mathfrak{g}_3 \cdot \mathfrak{g}_4') = \emptyset$ ((3)). Note, in addition, that $\|\alpha_i|_{\mathfrak{g}_1\cdot\mathfrak{g}_3}\|^j \subseteq \{\perp^{1-i}\}$, as decorations at $\mathfrak{g}_1$ are the same as at $\mathfrak{g}_1 \cdot \mathfrak{g}_2$ ((4)).

We can repeat the decomposition in the previous paragraph as long as (*) and (**) can be fulfilled. In particular, we can do this until there are no repeating pairs of decorations on positions of length greater than or equal to n_E and before smallest position $\mathfrak{h}$ such that $\|\alpha_i|_{\mathfrak{h}}\|^j \subseteq \{\perp^{1-i}\}$. By means of such repetition we can reach the situation such that $|\mathfrak{g}_1| = n_E$ and $|\mathfrak{g}_3| \leq 2^{2(|\mathrm{FV}(E)|+2)}$ and consequently $|\mathfrak{g}_1 \cdot \mathfrak{g}_3 \cdot \mathfrak{g}_4'| \leq |\mathfrak{g}_1 \cdot \mathfrak{g}_3| + |\mathfrak{g}_4'| \leq 2|\mathfrak{g}_1 \cdot \mathfrak{g}_3| \leq 2n_E + 2^{2(|\mathrm{FV}(E)|+2)+1}$ ((5)).

We can now reach the conclusion of the current lemma for $S' = \hat{S}$ as well as $\mathfrak{g}' = \mathfrak{g}_1 \cdot \mathfrak{g}_3 \cdot \mathfrak{g}_4'$ and $\mathfrak{g}'' = \mathfrak{g}_1 \cdot \mathfrak{g}_3$ ((2)). The places where we obtain particular statements of the conclusion are indicated by the numbers in double parentheses above. □

In case of infinite type expressions the estimation is similar, but slightly bigger.

Lemma 6.6 `(short critical position witness, infinite)`
Let $\alpha_0, \alpha_1 \in \mathrm{FV}(E)$ and $\alpha_0 \leq \alpha_1 \notin E$.

Assume that S avoids constants and $\mathfrak{g}$ is a position such that $S(\alpha_i)|_{\mathfrak{g}} = \perp^{1-i}$ and $\mathfrak{g}$ is accessible in $S(\alpha_{1-i})$ with $S(\alpha_{1-i})|_{\mathfrak{g}} \neq \perp^{1-i}$.

Then there is a substitution S' that avoids constants and positions $\mathfrak{g}', \mathfrak{g}''$ such that

1. *$S' \models E$;*
2. *$\mathfrak{g}'' \preceq \mathfrak{g}'$;*
3. *$\Upsilon_i^{\alpha_0 \rightsquigarrow \alpha_0}(\mathfrak{g}') = \{\mathfrak{g}'\}$ and $\Upsilon_i^{\alpha_0 \rightsquigarrow \alpha_1}(\mathfrak{g}') = \emptyset$;*
4. *$\|\alpha_i|_{\mathfrak{g}''}\|^{1-i} \subseteq \{\perp^{1-i}\}$;*
5. *$|\mathfrak{g}'| \leq 2n_E + 2^{4(|\mathrm{FV}(E)|+2)+1}$ and $|\mathfrak{g}''| \leq 2n_E + 2^{4(|\mathrm{FV}(E)|+2)+1}$;*
6. *$S'(\alpha_i)|_{\mathfrak{g}'} = \perp^{1-i}$ and $S'(\alpha_{1-i})|_{\mathfrak{g}'} \neq \perp^{1-i}$.*

Proof:
The proof goes along the same lines as the proof of Lemma 6.5, but we cannot use the assumption that two coordinates in static decorations are empty. Therefore, repeating of static decorations is guaranteed to hold only when the $\mathfrak{g}_2$ is of length greater than $(2^{(|\mathrm{FV}(E)|+2)})^4$. The rest of the construction and estimations remains the same and thus the final estimation on the length of $\mathfrak{g}'$ is $2n_E + 2^{4(|\mathrm{FV}(E)|+2)+1}$. □

Theorem 6.7 `(inentailment)`
The inentailment problem is decidable in PSPACE, both for finite and infinite type expressions.

Proof:
Assume we are given a set of inequations E and an inequation $\alpha_0 \leq \alpha_1$. Due to Theorem 3.3 we may assume that E is a saturated set of small inequations.

The algorithm works as follows:

1. If E is not solvable then fail.

2. If E is solvable, guess a substitution S_0 such that $S_0(\alpha) \in \{\perp, \top\}$ for all $\alpha \in \mathrm{dom}(S_0)$, to obtain $E' = S_0(E) \in \mathcal{E}$.

3. Guess a number n such that $n \leq 2n_E + 2^{2(|\mathrm{FV}(E)|+2)+1}$ (or $2n_E + 2^{4(|\mathrm{FV}(E)|+2)+1}$ for infinite type expressions) and choose $i \in \{0, 1\}$.

4. Starting with $\mathfrak{h}_0 = \varepsilon$ at each step j choose $\mathfrak{h}_j$ of length $\leq n$ such that $\mathfrak{h}_{j-1} \prec \mathfrak{h}_j$ and $|\mathfrak{h}_{j-1}| + 1 = |\mathfrak{h}_j|$, generate $\langle \|\alpha_i|_{\mathfrak{h}_j}\|^0, \|\alpha_i|_{\mathfrak{h}_j}\|^1 \rangle$, $\langle \|\alpha_{1-i}|_{\mathfrak{h}_j}\|^0, \|\alpha_{1-i}|_{\mathfrak{h}_j}\|^1 \rangle$ and at each step check the conditions

   (a) $\|\alpha_i|_{\mathfrak{h}_j}\|^{1-i} \subseteq \{\perp^{1-i}\}$,

   (b) $\|\alpha_i|_{\mathfrak{h}_j}\|^{1-i} \cap \|\alpha_{1-i}|_{\mathfrak{h}_j}\|^{i} = \emptyset$.

5. If the condition (4b) above fails for any $\mathfrak{h}_j$ then fail. If j exceeds n then fail. If the condition (4a) succeeds then succeed and return $\mathfrak{h} = \mathfrak{h}_j$ and i.

This algorithm is sound as once it succeeds we can construct a solution S and find a position $\mathfrak{g}$ such that $S(\alpha_i)|_{\mathfrak{g}} = \perp^{1-i}$, $\blacktriangle S(\alpha_{1-i})|_{\mathfrak{g}} \neq \perp^{1-i}$. To this end we take a minimal solution S_0 for E (i.e. one for which the sum of sizes of expressions assigned to variables is minimal). If necessary, we can extend S_0 towards $\mathfrak{h}$ (see Prop. 5.6) and obtain a solution S_1 such that it meets the assumptions of Lemma 6.3 (properties checked in step 4 of the algorithm are necessary here). This lemma gives a substitution S and a position $\mathfrak{g} \succeq \mathfrak{h}$ such that $S \models E$, $S(\alpha_i)|_{\mathfrak{g}} = \perp^{1-i}$ and $\blacktriangle S(\alpha_{1-i})|_{\mathfrak{g}} = \circledcirc$, which was required.

This algorithm is also complete as when there is a solution S such that $S \models E$ and $S \not\models \alpha_0 \leq \alpha_1$ then for all $\beta \notin \{\alpha_0, \alpha_1\}$ such that $S(\beta) \in \{\perp, \top\}$ we define $T(\beta) = S(\beta)$ and consider $E' = T(E) \in \mathcal{E}$ (see Prop. 3.4). Note that $S' = S \backslash T$ is a solution of E'. Additionally, S' avoids constants. We observe that there is a critical position, i.e. for some i and $\mathfrak{g}$ we have $S'(\alpha_i)|_{\mathfrak{g}} = \perp^{1-i}$ and $S'(\alpha_{1-i})|_{\mathfrak{g}} \neq \perp^{1-i}$. By Lemma 6.5 there is a substitution S'' and a critical position of bounded depth, i.e. $\mathfrak{g}'$ with $|\mathfrak{g}'| \leq 2n_E + 2^{2(|\mathrm{FV}(E)|+2)+1}$ (by Lemma 6.6 we obtain bound $|\mathfrak{g}'| \leq 2n_E + 2^{4(|\mathrm{FV}(E)|+2)+1}$ in case of infinite type expressions) such that $S'' \models E'$ and $S''(\alpha_i)|_{\mathfrak{g}'} = \perp^{1-i} \neq S''(\alpha_{1-i})|_{\mathfrak{g}}$. Moreover, there is $\mathfrak{g}''$ such that $\mathfrak{g}'' \preceq \mathfrak{g}'$ and (*) $\|\alpha_i|_{\mathfrak{g}''}\|^{1-i} \subseteq \{\perp^{1-i}\}$, as well as $\curlyvee_i^{\alpha_i \rightsquigarrow \alpha_{1-i}}(\mathfrak{g}') = \emptyset$. Observe that the latter condition implies that for each $\mathfrak{h} \preceq \mathfrak{g}''$ we have (**) $\|\alpha_i|_{\mathfrak{h}}\|^{1-i} \cap \|\alpha_{1-i}|_{\mathfrak{h}}\|^{i} = \emptyset$. We can now let the algorithm to choose $\mathfrak{h}_j$ in step (4) as subsequent prefixes of $\mathfrak{g}''$. The position $\mathfrak{g}''$ is reached by the algorithm since the counter j does not exceed $n \geq |\mathfrak{g}'| \geq |\mathfrak{g}''|$ due to the condition (*) and because the algorithm cannot fail due to the fact that (4b) is satisfied at all previous steps because (**) holds.

This algorithm is in PSPACE as one needs polynomial amount of space to maintain a counter j that counts up to n. One needs also polynomial space to maintain pairs $\langle \|\alpha_i|_{\mathfrak{h}_j}\|^0, \|\alpha_i|_{\mathfrak{h}_j}\|^1 \rangle$, $\langle \|\alpha_{1-i}|_{\mathfrak{h}_j}\|^0, \|\alpha_{1-i}|_{\mathfrak{h}_j}\|^1 \rangle$. □

Corollary 6.8 (`entailment`)
The entailment problem is decidable in PSPACE both for finite and infinite type expressions.

Proof:
The mentioned above algorithm can be easily turned to algorithm for the entailment problem at the cost of change in the complexity class to the co-counterpart, in this case the same class. □

7 Final remarks

The worst-case exponential size of the critical position probably cannot be significantly reduced. It is also difficult to imagine that the size of positions in semi-shortenings can be significantly reduced.

Although the main result of this paper is discouraging, one can also see it as an inspiration to study for which classes of type constraints the type entailment problem can be solved quickly. It is especially worth pursuing since type entailment plays important role in backwards compatibility checks for reimplemented functions.

The construction in this paper, as the one of Niehren and Priesnitz [NP03], can be reworked at the cost of additional significant notational overhead to deal with a contravariant symbol. One will have to turn uniform comparison by $\leq^i$ used in this paper to a version $\leq^{i\star\mathfrak{g}}$ that is parameterised by a position $\mathfrak{g}$ at which the comparison is done, i.e. one that compares covariantly $\leq^i$ on positive positions and contravariantly $\geq^i$ on negative ones. This will also require change in dynamic decorations (see Def. 9) where one has to propagate $\perp^i$ in different directions depending on the polarity. After the operations on substitutions are done properly, the pumping arguments will be the same for the contravariant symbol. Since the arguments in this paper are complicated, it was decided to make the presentation in the simpler format with the covariant symbol.

References

[AC91] Roberto M. Amadio and Luca Cardelli. Subtyping recursive types. In *Proceedings of the 18th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pages 104–118, New York, NY, USA, 1991. Association for Computing Machinery.

[AC93] Roberto M. Amadio and Luca Cardelli. Subtyping recursive types. *ACM Transactions on Programming Languages and Systems*, 15(4):575–631, 1993.

[Car84] Luca Cardelli. A semantics of multiple inheritance. In Gordon Plotkin Gilles Kahn, David B. MacQueen, editor, *Semantics of Data Types*, pages 51–67, Berlin, Heidelberg, 1984. Springer Berlin Heidelberg.

[Car88] Luca Cardelli. A semantics of multiple inheritance. *Information and Computation*, 76(2):138–164, 1988.

[CD78] Mario Coppo and Mariangiola Dezani-Ciancaglini. A new type assignment for λ-terms. *Arch. Math. Log.*, 19(1):139–156, 1978.

[CLPS19] Giuseppe Castagna, Victor Lanvin, Tommaso Petrucciani, and Jeremy G. Siek. Gradual typing: a new perspective. *Proc. ACM Program. Lang.*, 3(POPL):16:1–16:32, 2019.

[ESTZ94] Jonathan Eifrig, Scott F. Smith, Valery Trifonov, and Amy E. Zwarico. Application of OOP type theory: State, decidability, integragtion. In Jeff McKenna, J. Eliot B. Moss, and Richard L. Wexelblat, editors, *OOPSLA'94, Proceedings of the Ninth Annual Conference on Object-Oriented Programming Systems, Languages, and Applications, Portland, Oregon, USA, October 23-27, 1994*, pages 16–30. ACM, 1994.

[FM90] You-Chin Fuh and Prateek Mishra. Type inference with subtypes. *Theoretical Computer Science*, 73(2):155–175, 1990.

[Gom90] Carsten K. Gomard. Partial type inference for untyped functional programs. In *Proceedings of the 1990 ACM Conference on LISP and Functional Programming*, page 282–287, New York, NY, USA, 1990. ACM.

[HR97] Fritz Henglein and Jakob Rehof. The complexity of subtype entailment for simple types. In *Proceedings of the 12th Annual IEEE Symposium on Logic in Computer Science*. IEEE Computer Society, 1997.

[HR98] Fritz Henglein and Jakob Rehof. Constraint automata and the complexity of recursive subtype entailment. In Kim G. Larsen, Sven Skyum, and Glynn Winskel, editors, *Automata, Languages and Programming*, pages 616–627, Berlin, Heidelberg, 1998. Springer Berlin Heidelberg.

[Jen91] Thomas P. Jensen. Strictness analysis in logical form. In John Hughes, editor, *Functional Programming Languages and Computer Architecture*, pages 352–366, Berlin, Heidelberg, 1991. Springer Berlin Heidelberg.

[KPS94] Dexter Kozen, Jens Palsberg, and Michael I. Schwartzbach. Efficient inference of partial types. *Journal of Computer and System Sciences*, 49(2):306–324, 1994.

[KPS95] Dexter Kozen, Jens Palsberg, and Michael I. Schwartzbach. Efficient recursive subtyping. *Mathematical Structures in Computer Science*, 5(1):113–125, 1995.

[Mit84] John C. Mitchell. *Lambda Calculus Models of Typed Programming Languages*. PhD thesis, Massachusetts Institute of Technology, 1984.

[Mit91] John C. Mitchell. Type inference with simple subtypes. *Journal of Functional Programming*, 1(3):245–285, 1991.

[NP03] Joachim Niehren and Tim Priesnitz. Non-structural subtype entailment in automata theory. *Information and Computation*, 186(2):319–354, 2003. Theoretical Aspects of Computer Software (TACS 2001).

[PO95a] Jens Palsberg and Patrick O'Keefe. A type system equivalent to flow analysis. *ACM Trans. Program. Lang. Syst.*, 17(4):576–599, July 1995.

[PO95b] Jens Palsberg and Patrick O'Keefe. A type system equivalent to flow analysis. In *Proceedings of the 22nd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pages 367—378, New York, NY, USA, 1995. Association for Computing Machinery.

[Pot96] François Pottier. Simplifying subtyping constraints. In *Proceedings of the First ACM SIGPLAN International Conference on Functional Programming*, pages 122–133, New York, NY, USA, 1996. Association for Computing Machinery.

[Pot98a] François Pottier. A framework for type inference with subtyping. In Matthias Felleisen, Paul Hudak, and Christian Queinnec, editors, *Proceedings of the third ACM SIGPLAN International Conference on Functional Programming (ICFP '98), Baltimore, Maryland, USA, September 27-29, 1998*, pages 228–238. ACM, 1998.

[Pot98b] François Pottier. *Type inference in the presence of subtyping: from theory to practice*. PhD thesis, Université Paris 7, 1998. English version released as the Research Report no 3483 by INRIA.

[Pot01] François Pottier. Simplifying subtyping constraints: A theory. *Information and Computation*, 170(2):153–183, 2001.

[PWO97] Jens Palsberg, Mitchel Wand, and Patrick O'Keefe. Type inference with non-structural subtyping. *Formal Aspects of Computing*, (9):49–67, 1997.

[Reh98] Jakob Rehof. *The complexity of simple subtyping systems*. PhD thesis, DIKU, Copenhagen, 1998.

[SAN+02] Zhendong Su, Alexander Aiken, Joachim Niehren, Tim Priesnitz, and Ralf Treinen. The first-order theory of subtyping constraints. In *Proceedings of the 29th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pages 203–216. ACM, 2002.

[ST06] Jeremy G. Siek and Walid Taha. Gradual typing for functional languages. In *Proceedings of the 2006 Scheme and Functional Programming Workshop*, 2006.

[ST07] Jeremy G. Siek and Walid Taha. Gradual typing for objects. In Erik Ernst, editor, *ECOOP 2007 – Object-Oriented Programming, 21st European Conference, Berlin, Germany, July 30 - August 3, 2007, Proceedings*, volume 4609 of *LNCS*, pages 2–27. Springer, 2007.

[Tha88] Satish Thatte. Type inference with partial types. In Timo Lepistö and Arto Salomaa, editors, *Automata, Languages and Programming*, pages 615–629, Berlin, Heidelberg, 1988. Springer Berlin Heidelberg.

[Tre00] Ralf Treinen. Predicate logic and tree automata with tests. In Jerzy Tiuryn, editor, *Foundations of Software Science and Computation Structures: Third International Conference, FOSSACS 2000. Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2000, Berlin, Germany, March/April 2000. Proceedings*, volume 1784 of *LNCS*, pages 329–343. Springer-Verlag, 2000.

[Wan84] Mitchell Wand. A semantic prototyping system. In *Proceedings of the 1984 SIGPLAN Symposium on Compiler Construction*, page 213–221, New York, NY, USA, 1984. ACM.

[WOP95] Mitchell Wand, Patrick O'Keefe, and Jens Palsberg. Strong normalization with nonstructural subtyping. *Math. Struct. Comput. Sci.*, 5(3):419–429, 1995.