Decidability of DPDA equivalence

Colin Stirling
Division of Informatics
University of Edinburgh

email: cps@dcs.ed.ac.uk

Abstract

A proof of decidability of equivalence between deterministic pushdown automata is presented using a mixture of methods developed in concurrency and language theory. The technique appeals to a tableau proof system for equivalence of configurations of strict deterministic grammars.

1 The DPDA problem

Ingredients of pushdown automata with ε-transitions are a finite set of states $\mathcal{P}$, a finite set of stack symbols $\mathcal{S}$, a finite alphabet $\mathcal{A}$ and a finite family of basic transitions, each of the form $pS \xrightarrow{a} q\alpha$ where p, q are states, $a \in \mathcal{A} \cup \{\varepsilon\}$, S is a stack symbol and α is a sequence of stack symbols. A configuration of an automaton is any expression $p\alpha$, $p \in \mathcal{P}$ and $\alpha \in \mathcal{S}^*$ whose behaviour is determined by the basic transitions together with the following prefix rule, where $\beta \in \mathcal{S}^*$:

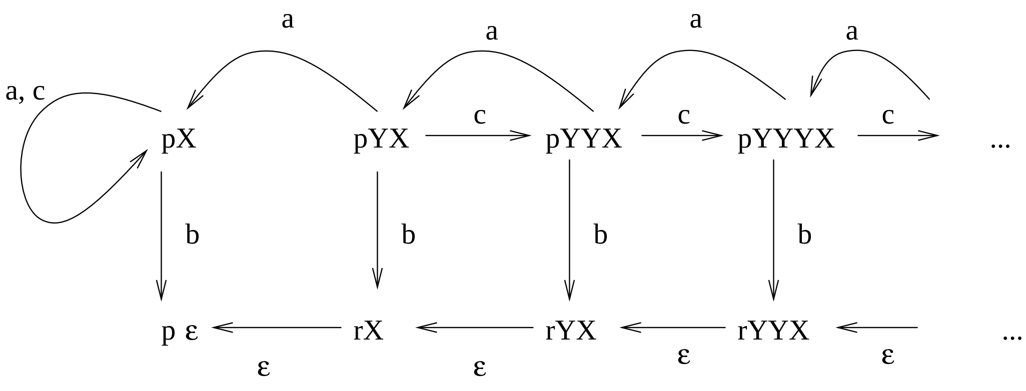
$$\text{if } pS \xrightarrow{a} q\alpha \text{ then } pS\beta \xrightarrow{a} q\alpha\beta$$

The language accepted by a configuration $p\alpha$ is $\{w \in \mathcal{A}^* \ : \ \exists q \in \mathcal{P}.\, p\alpha \xrightarrow{w} q\epsilon\}$ where the extended transitions for words are defined as expected. Note that ε-transitions are swallowed in the usual fashion. Acceptance is by empty stack (and not by final state, see [5]).

A deterministic pushdown automaton, DPDA, has restrictions on its basic transitions (where $pS \xrightarrow{a}$ abbreviates $pS \xrightarrow{a} q\alpha$ for some q and some α)

$$\begin{array}{l} \text{if } pS \xrightarrow{a} q\alpha \text{ and } pS \xrightarrow{a} r\beta \text{ then } q = r \text{ and } \alpha = \beta \\ \text{if } pS \xrightarrow{a} \ \text{ and } a \in \mathcal{A} \text{ then not}(pS \xrightarrow{\varepsilon}) \end{array}$$

Moreover one can assume that in a basic transition $pS \xrightarrow{a} q\alpha$ the length of α is less than 3, and that ε-transitions can only pop the stack: if $pS \xrightarrow{\varepsilon} q\alpha$ then



$$
\begin{array}{llll}
pX \xrightarrow{a} pX & pX \xrightarrow{b} p\epsilon & pX \xrightarrow{c} pX & rX \xrightarrow{\epsilon} p\epsilon \\
pY \xrightarrow{a} p\epsilon & pY \xrightarrow{b} r\epsilon & pY \xrightarrow{c} pYY & rY \xrightarrow{\epsilon} r\epsilon
\end{array}
$$

Figure 1: A DPDA

$\alpha = \epsilon$. One consequence of the restrictions is that the language accepted by a configuration is prefix-free: if w is accepted then no proper prefix of w is accepted. Thus if ϵ is accepted then no other word is. In the following we assume ϵ-free DPDAs[1]. Figure 1 depicts a simple DPDA whose basic transitions are listed under the diagram.

The DPDA decidability problem was first posed in 1966 [2]. Is there an effective procedure for deciding whether or not two configurations of a DPDA accept the same language?[2] Why is the decision question so difficult to answer, despite all the intensive work on the problem over the past 30 years? Because one needs to expose the right structure. It appears that the notation of pushdown configurations, although simple, is not rich enough. Attempts to prove the result (such as Valiant's technique) examine differences between stack lengths and potentially equivalent configurations. This method showed decidability of equivalence for real-time DPDAs which have no ϵ-transitions [7]. But when there are ϵ-transitions it is possible for configurations of arbitrary size to be equivalent. For example configurations pY^nX and pY^mX of Figure 1 are equivalent for all m and n. Finally Sénizergues [8] showed decidability. However his proof is formidable and is over 100 pages long [9]. It exposes structure in DPDAs using power series which leads to quite difficult notation.

Instead we provide a proof using a mixture of techniques developed in concurrency theory for showing decidability of bisimulation equivalence and ideas from

[1] Classical DPDAs have final states, but for any language L recognised by a configuration of such a DPDA there is a configuration of an ϵ-free DPDA with empty stack acceptance for the language $L\$$ where \$ is an endmarker.

[2] As the disjoint union of two DPDAs is a DPDA, we can assume the decision question over configurations of a single DPDA instead of between two different DPDAs.

language theory, especially those developed in [4]. The proof also essentially depends on insights gained from trying to understand Sénizergues's method. The proof consists of two semi-decision procedures. One half of the proof is easy: if two configurations do not accept the same language then there is a smallest word which distinguishes them. Therefore we just need to establish semi-decidability of accepting the same language. The crux of this part of the proof is that there is a finite tableau proof that two configurations are equivalent. The method employs ideas developed in [6, 1, 10]. It relies upon exposure of structure within DPDA, so that we can "decompose" configurations. We shall build a structured language (almost a process algebra) for describing configurations of a DPDA using strict deterministic grammars.

2 Strict deterministic grammars

An ϵ-free context-free grammar in 3-Greibach normal form consists of a finite family $\mathcal{N}$ of nonterminals, a finite alphabet $\mathcal{A}$ and a finite family of basic transitions, each of the form $X \xrightarrow{a} \alpha$ where $X \in \mathcal{N}$, $a \in \mathcal{A}$ and $\alpha \in \mathcal{N}^*$ such that its length, $|\alpha|$, is less than 3. A simple configuration is a sequence of nonterminals whose behaviour is determined by the basic transitions and the prefix rule: if $X \xrightarrow{a} \alpha$ then $X\beta \xrightarrow{a} \alpha\beta$ where $\beta \in \mathcal{N}^*$. The language accepted by a simple configuration is the set of words $\{w \in \mathcal{A}^* : \alpha \xrightarrow{w} \epsilon\}$. However we shall also consider composite configurations which are finite families of simple configurations. We use the (process calculus) notation $\alpha_1 + \ldots + \alpha_n$ for such a configuration. The language accepted by a composite configuration is the union of the languages accepted by its components.

We are interested in a restricted family of context-free grammars, the *strict deterministic grammars* [3, 4]. Assume a context-free grammar (in 3-Greibach normal form). Let $\equiv$ be a partition of its nonterminals $\mathcal{N}$. We extend $\equiv$ to sequences of nonterminals, $\alpha \equiv \beta$ if $\alpha = \beta$ or there is a δ such that $\alpha = \delta X \alpha_1$ and $\beta = \delta Y \beta_1$ and $X \equiv Y$ and $X \neq Y$. The partition $\equiv$ on $\mathcal{N}$ is *strict* if the basic transitions obey the following two conditions:

$$\text{if } X \xrightarrow{a} \alpha \text{ and } Y \xrightarrow{a} \delta \text{ and } X \equiv Y \text{ then } \alpha \equiv \delta$$
$$\text{if } X \xrightarrow{a} \alpha \text{ and } Y \xrightarrow{a} \alpha \text{ and } X \equiv Y \text{ then } X = Y$$

An immediate consequence of these conditions is that if $X \xrightarrow{a} \epsilon$ and $X \equiv Y$ and $X \neq Y$ then not($Y \xrightarrow{a}$). A context-free grammar is strict deterministic if there exists a strict partition of its nonterminals.

We now examine some properties of strict deterministic grammars. First the strictness conditions generalise to words (see [4] for a proof).

Fact 1

i. *If* $X \xrightarrow{w} \alpha$ *and* $Y \xrightarrow{w} \delta$ *and* $X \equiv Y$ *then* $\alpha \equiv \delta$.

ii. *If* $X \xrightarrow{w} \alpha$ *and* $Y \xrightarrow{w} \alpha$ *and* $X \equiv Y$ *then* $X = Y$.

Therefore if $X \equiv Y$ then the languages accepted by X and Y are prefix-disjoint and if $X \neq Y$ then they accept disjoint languages (again see [4] for a proof).

Fact 2 *If* $X \equiv Y$ *and* $X \xrightarrow{w} \epsilon$ *then*
i. $not(Y \xrightarrow{u} \epsilon)$ *for any proper prefix* u *of* w, *and*
ii. $not(Y \xrightarrow{w} \epsilon)$ *when* $X \neq Y$.

Our main concern is with a subset of composite configurations. A composite configuration $\beta_1 + \ldots + \beta_n$ is *admissible* if $\beta_i \equiv \beta_j$ for each pair of components β_i and β_j. For ease of notation we also assume that the empty sum, $\emptyset$, is admissible. In [4] admissible configurations are called "associates". Notice that if $X \equiv Y$ then for any w the sum $\sum\{\beta : X \xrightarrow{w} \beta \text{ or } Y \xrightarrow{w} \beta\}$ is admissible. Moreover "reachability" under any word preserves admissibility (yet again see [4] for a proof).

Fact 3 *If* $\beta_1 + \ldots + \beta_n$ *is admissible then for any* w *the following is admissible*

$$\sum\{\beta'_1 : \beta_1 \xrightarrow{w} \beta'_1\} + \ldots + \sum\{\beta'_n : \beta_n \xrightarrow{w} \beta'_n\}$$

An ϵ-free DPDA can be transformed into an equivalent 3-Greibach normal form strict deterministic grammar (and vice versa) [3]. For every pair of states p, q and stack symbol S introduce a nonterminal $[pSq]$ (whose language will be $\{w \in \mathcal{A}^* : pS \xrightarrow{w} q\epsilon\}$). The basic transitions for $a \in \mathcal{A}$ are also translated: $pS \xrightarrow{a} q\epsilon$ becomes $[pSq] \xrightarrow{a} \epsilon$, $pS \xrightarrow{a} qT$ becomes the family for each r, $[pSr] \xrightarrow{a} [qTr]$, and $pS \xrightarrow{a} qTU$ becomes the family for each r and p', $[pSr] \xrightarrow{a} [qTp'][p'Ur]$. Erase all ϵ-nonterminals (if $pS \xrightarrow{\varepsilon} q\epsilon$ then $[pSq]$ is an ϵ-nonterminal) from the right hand side of any transition. Next delete all transitions involving redundant nonterminals (those which accept no words). It is easy to check that the partition $\equiv$ relating pairs $[pSq]$ and $[pSr]$ is strict. A configuration $pS_1S_2 \ldots S_n$ of a DPDA becomes the following admissible configuration, where the summation is over all p_i, $1 \leq i \leq n$

$$\sum [pS_1p_1][p_1S_2p_2] \ldots [p_{n-1}S_np_n]$$

after all ϵ-nonterminals are erased and all components involving redundant nonterminals are removed.

An example is the translation of the DPDA of Figure 1 into a strict deterministic grammar. First the basic transitions are transformed as follows.

$$\begin{array}{lll}
[pXp] \xrightarrow{a} [pXp] & [pXr] \xrightarrow{a} [pXr] & [pXp] \xrightarrow{b} \epsilon \\
[pXp] \xrightarrow{c} [pXp] & [pXr] \xrightarrow{c} [pXr] & \\
[pYp] \xrightarrow{a} \epsilon & [pYr] \xrightarrow{b} \epsilon & [pYp] \xrightarrow{c} [pYp][pYp] \\
[pYp] \xrightarrow{c} [pYr][rYp] & [pYr] \xrightarrow{c} [pYp][pYr] & [pYr] \xrightarrow{c} [pYr][rYr]
\end{array}$$

There are also two ϵ-nonterminals, $[rXp]$ and $[rYr]$. These are erased from the right hand side of any transition: the transition $[pYr] \stackrel{c}{\longrightarrow} [pYr][rYr]$ is changed to $[pYr] \stackrel{c}{\longrightarrow} [pYr]$. There are also two redundant nonterminals $[pXr]$ and $[rYp]$. All transitions involving these nonterminals are removed. This reduces the transitions to the following

$$
\begin{array}{lll}
[pXp] \stackrel{a}{\longrightarrow} [pXp] & [pXp] \stackrel{b}{\longrightarrow} \epsilon & [pXp] \stackrel{c}{\longrightarrow} [pXp] \\
[pYp] \stackrel{a}{\longrightarrow} \epsilon & [pYr] \stackrel{b}{\longrightarrow} \epsilon & [pYp] \stackrel{c}{\longrightarrow} [pYp][pYp] \\
[pYr] \stackrel{c}{\longrightarrow} [pYp][pYr] & [pYr] \stackrel{c}{\longrightarrow} [pYr] &
\end{array}
$$

The set of nonterminals is $\{[pXp], [pYp], [pYr]\}$ and the partition is into the sets $\{\{[pXp]\}, \{[pYp], [pYr]\}\}$. An example configuration $pYYX$ of the DPDA becomes the following admissible configuration

$$[pYp][pYp][pXp] + [pYp][pYr] + [pYr]$$

The DPDA problem is equivalent to the question of decidability of (language) equivalence between admissible configurations of a strict deterministic grammar. Therefore we now extend the transition relation to admissible configurations. The idea is as in process calculi that one builds transitions from a composite process out of transitions of its components. First the basic transitions are determinized by coalescing all the basic transitions of a nonterminal with the same label: if $X \stackrel{a}{\longrightarrow} \alpha_1$ and $\ldots$ and $X \stackrel{a}{\longrightarrow} \alpha_n$ then form the single transition $X \stackrel{a}{\longrightarrow} \alpha_1 + \ldots + \alpha_n$[3]. We also assume that if not$(X \stackrel{a}{\longrightarrow})$ then $X \stackrel{a}{\longrightarrow} \emptyset$. Consequently for each nonterminal X and each $a \in \mathcal{A}$ there is a single transition rule $X \stackrel{a}{\longrightarrow} \sum \alpha_j$. For instance the rule for $[pYr]$ and c above becomes $[pYr] \stackrel{c}{\longrightarrow} [pYp][pYr] + [pYr]$. The transition rule for admissible configurations, the prefix rule, is then as follows.

$$\text{if } X_i \stackrel{a}{\longrightarrow} \sum \alpha_{ij} \text{ then } \sum X_i \beta_i \stackrel{a}{\longrightarrow} \sum \sum \alpha_{ij} \beta_i$$

For example the admissible configuration corresponding to $pYYX$ above has the following c transition

$$
\begin{array}{c}
[pYp][pYp][pXp] + [pYp][pYr] + [pYr] \\
\downarrow\ c \\
[pYp][pYp][pYp][pXp] + [pYp][pYp][pYr] + [pYp][pYr] + [pYr]
\end{array}
$$

The resulting configuration corresponds to $pYYYX$.

The extended transition relation $\stackrel{w}{\longrightarrow}$, $w \in \mathcal{A}^*$, is defined as usual. Consequently the language accepted by an admissible configuration $\beta_1 + \ldots + \beta_k$ is the set of words $\{w : \beta_1 + \ldots + \beta_k \stackrel{w}{\longrightarrow} \epsilon\}$. The rest of the paper is devoted to the proof of decidability of language equivalence between admissible configurations. In the next section we introduce some useful notation. In section 4 we isolate crucial combinatorial properties of admissible configurations which underpin the tableau proof system for showing decidability in section 5.

[3] By strictness $\alpha_1 + \ldots + \alpha_n$ is admissible.

3 Measures, shapes and recursive nonterminals

Assume a fixed strict deterministic grammar in 3-Greibach normal form without redundant nonterminals. We use α, β, ... to range over sequences of nonterminals and E, F, G, ... to range over admissible configurations. The size of E, written $|E|$, is the length of its longest sequence of nonterminals: if E is $\beta_1 + \ldots + \beta_n$ then $|E|$ is $\max\{|\beta_j| \,:\, 1 \leq j \leq n\}$[4]. Notice that for each n there are only finitely many admissible configurations of size n.

We assume a fixed total ordering on the alphabet $\mathcal{A}$. From this we define a total ordering on words: $u < v$ if $|u| < |v|$ or when $|u| = |v|$ u is lexicographically less than v. If $u < v$ we say that u is shorter than v. For each nonterminal X there is a unique shortest word u such that $X \xrightarrow{u} \epsilon$. We let $w(X)$ denote this word and we let the *norm* of X, written $n(X)$, be its length. An important measure is M which is just larger than the maximum norm.

$$M \quad = \quad 1 + \max\{n(X) \,:\, X \text{ is a nonterminal}\}$$

The notion of norm extends to admissible configurations: $n(E)$ is the length of the shortest word u such that $E \xrightarrow{u} \epsilon$[5]. Infinitely many different admissible configurations can have the same norm (and this is one reason why the decision problem is difficult).

An admissible configuration has a variety of "shapes" as it can be presented in many different ways using the following (obvious) equalities.

$$\begin{array}{llll} E + \emptyset & = E & \emptyset E & = \emptyset \\ E(F + G) & = EF + EG & (E + F)G & = EG + FG \end{array}$$

One basic shape is "head nonterminal form" $X_1G_1 + \ldots + X_kG_k$ where $X_i \neq X_j$, $i \neq j$, and $X_i \equiv X_j$: some of the G_is may be ϵ. In this case the X_is are the "heads" and the G_is are the "tails". Another important head form is $\beta_1G_1 + \ldots + \beta_nG_n + E'$ where $\beta_i \equiv \beta_j$ (and $\beta_i \neq \beta_j$, $i \neq j$) and $|\beta_i| = |\beta_j|$ and G_i is not ϵ, and $|E'| \leq |\beta_i|$. Instead one may focus on "tail" forms. If $X_i \xrightarrow{w} E_i$ for each $i : 1 \leq i \leq k$ (where some E_is may be $\emptyset$) then $X_1G_1 + \ldots + X_kG_k \xrightarrow{w} E_1G_1 + \ldots + E_kG_k$. The shape $E_1G_1 + \ldots + E_kG_k$ highlights the tails. Because the grammar is in 3-Greibach normal form $|E_i| \leq 1 + |w|$ for each i. In this example the "head" $E_1 + \ldots + E_k$ is itself admissible. Tail forms $E_1G_1 + \ldots + E_kG_k$ are only permitted when the "head" is admissible. For instance $[pYp][pYp][pXp] + [pYp][pYr] + [pYr]$ is not permitted to have as tail form $[pYp]G_1 + ([pYp] + \epsilon)G_2$. However it does have the form $[pYp]G_1 + [pYr]G_2$.

Another useful notation is "the result of E after u", written $E \cdot u$. This is the configuration F such that $E \xrightarrow{u} F$, which can be $\emptyset$. An example is

[4] We let $|\emptyset| = 0$.

[5] We assume $n(\emptyset) = \infty$.

$(X_1G_1 + \ldots + X_nG_n) \cdot w(X_i) = G_i$. If $|u| < \min\{n(E_i) : 1 \leq i \leq k\}$ then $(E_1G_1 + \ldots + E_kG_k) \cdot u = (E_1 \cdot u)G_1 + \ldots + (E_k \cdot u)G_k$.

Although the starting point is a fixed strict deterministic grammar we shall extend it with auxilary nonterminals, ranged over by V, each of which has an associated definition $V \stackrel{\text{def}}{=} H$. We say that $(V_1, \ldots, V_n)$ is a family of *recursive nonterminals* if for each $i : 1 \leq i \leq n$ either $V_i \stackrel{\text{def}}{=} H_{i1}V_1 + \ldots + H_{in}V_n$ where $H_{i1} + \ldots + H_{in}$ is admissible and each H_{ij} is distinct and does not contain auxilary nonterminals, or $V_i \stackrel{\text{def}}{=} V_j$ and $j \leq i$ and $V_j \stackrel{\text{def}}{=} V_j$. An auxilary nonterminal can only appear as a final element in a sequence of nonterminals. Admissibility is extended to such families of sequences as follows. A configuration which is a singleton V is admissible, and $\beta_1V_1' + \ldots + \beta_kV_k'$ is admissible if the head $\beta_1 + \ldots + \beta_k$ is admissible and each β_j is distinct, and there is a family of recursive nonterminals $(V_1, \ldots, V_n)$ such that each V_i' is one of the V_js. An admissible configuration can therefore be presented in tail form $E_1V_1 + \ldots + E_nV_n$. We assume that $|V| = 1$ for each recursive nonterminal V.

The transition rules are extended to the wider class of admissible configurations with the following rule

$$\text{if } E \xrightarrow{w} V_i \text{ and } V_i \stackrel{\text{def}}{=} H \text{ then } E \xrightarrow{w} H$$

The definition of $E \cdot u$ is refined so that it is always unique: if $E \xrightarrow{u} V_i$ and $V_i \stackrel{\text{def}}{=} H$ then $E \cdot u = H$. Consequently if $E \cdot u = V_i$ then $V_i \stackrel{\text{def}}{=} V_i$. The language accepted by an extended configuration E is the set $\{w : (E \cdot w) = V_i\}$. We view a recursive nonterminal V_i such that $V_i \stackrel{\text{def}}{=} V_i$ as a terminating nonterminal. The norm of E, $n(E)$, is again the length of the smallest word accepted by E. In the sequel we are only interested in normed configurations. This implies that in a recursive family $(V_1, \ldots, V_n)$ there is at least one terminating nonterminal.

Two configurations E and F are equivalent, written $E \sim F$, if they accept the same language and, when applicable, agree on terminating recursive nonterminals. If E and F are normed then E and F accept the same language iff they "reject" the same words, and because the language accepted by a configuration is prefix-free it follows that $E \sim F$ iff for all words w

$$(E \cdot w) = \emptyset \text{ iff } (F \cdot w) = \emptyset \quad \text{and} \quad (E \cdot w) = V_i \text{ iff } (F \cdot w) = V_i$$

Later we use this consequence as the criterion for equivalence of configurations. Below are some obvious properties of equivalence, including congruence.

Fact 1

i. *If* $E \sim F$ *then for all* $u \in A^*$, $E \cdot u \sim F \cdot u$.

ii. *If* $E \sim E'$ and $F \sim F'$ *then* $E + F \sim E' + F'$.

iii. *If* $EF \sim G$ *and* $F \sim F'$ *then* $EF' \sim G$.

iv. *If* $EF \sim G$ *and* $E \sim E'$ *then* $E'F \sim G$.

v. *If* $E \sim F$ *then* $n(E) = n(F)$.

The family $(V'_1, \ldots, V'_n)$ of recursive nonterminals *refines* the family $(V_1, \ldots, V_n)$ when the following two conditions hold.

$$\begin{array}{l} \text{if } V_i \stackrel{\text{def}}{=} H_1V_1 + \ldots + H_nV_n \text{ then } V'_i \stackrel{\text{def}}{=} H_1V'_1 + \ldots + H_nV'_n \\ \text{if } V_i \stackrel{\text{def}}{=} V_j \text{ and } V'_i \stackrel{\text{def}}{=} H \text{ then } V'_j \stackrel{\text{def}}{=} H \end{array}$$

A refined family agrees on the definitions of nonterminating nonterminals and preserves equality of definitions, but may contain fewer terminating nonterminals. Because equivalence of configurations includes agreement of terminating recursive nonterminals, equivalence is preserved by refinement.

Fact 2 *If* $E_1V_1 + \ldots + E_nV_n \sim F_1V_1 + \ldots + F_nV_n$ *and* $(V'_1, \ldots, V'_n)$ *refines* $(V_1, \ldots, V_n)$ *then* $E_1V'_1 + \ldots + E_nV'_n \sim F_1V'_1 + \ldots + F_nV'_n$.

Equivalence can be "approximated". For $n \geq 0$ we say that E and F are n-equivalent, written $E \sim_n F$, if for all words w whose length $|w| \leq n$

$$(E \cdot w) = \emptyset \text{ iff } (F \cdot w) = \emptyset \quad \text{and} \quad (E \cdot w) = V_i \text{ iff } (F \cdot w) = V_i$$

Note that for each n it is decidable whether $E \sim_n F$. Moreover $E \sim F$ iff $E \sim_n F$ for all $n \geq 0$. Below are some routine properties of the approximants $\sim_n$.

Fact 3

i. *If* $E \sim_n F$ *then for all* $u \in A^*$ *where* $|u| \leq n$, $E \cdot u \sim_{n-|u|} F \cdot u$.

ii. *If* $E \sim_n F$ and $0 \leq m < n$ *then* $E \sim_m F$.

iii. *If* $E \sim_n F$ *and* $F \not\sim_n G$ *then* $E \not\sim_n G$.

iv. *If* $E \sim_n E'$ and $F \sim_n F'$ *then* $E + F \sim_n E' + F'$.

v. *If* $EF \sim_n G$ *and* $E \sim_n E'$ *then* $E'F \sim_n G$.

4 Imbalance and size

Consider trying to show that $E \sim F$. One approach is goal directed. Start with the goal $E = F$ (to be understood as, "is $E \sim F$?") and then reduce it to subgoals. Keep reducing to further subgoals until one reaches either obviously true subgoals (such as $G = G$) or obviously false subgoals (such as $G = H$ when $\mathrm{n}(G) \neq \mathrm{n}(H)$). This naive technique which is described more formally in terms of tableaux in the next section is the approach adopted.

Assume that E has shape $E_1G_1 + \ldots + E_nG_n$ and that F has similar shape $F_1G_1 + \ldots + F_nG_n$. The measure of "imbalance" between E and F with these shapes is $\max\{\ |E_i|, |F_i|\ : 1 \leq i \leq n\}$. An important step in the proof is that one can bound imbalance by reducing goals to balanced subgoals.

Proposition 1 *If* $X_1H_1 + \ldots + X_kH_k \sim F$ *and* $E_1H_1 + \ldots + E_kH_k \sim F'$ *then* $E_1(F \cdot w(X_1)) + \ldots + E_k(F \cdot w(X_k)) \sim F'$.

Proof: Let E be $X_1H_1 + \ldots + X_kH_k$. Assume $E \sim F$ and $E_1H_1 + \ldots + E_kH_k \sim F'$. Since $(E \cdot w(X_i)) = H_i$ it follows that $H_i \sim (F \cdot w(X_i))$. By congruence $E_1H_1 + \ldots + E_kH_k \sim E_1(F \cdot w(X_1)) + \ldots + E_k(F \cdot w(X_k))$, see Fact 1 of the previous section. □

Proposition 1 will be used to reduce imbalance. Let $X_1H_1 + \ldots + X_kH_k = F$ and $E_1H_1 + \ldots + E_kH_k = F'$ be two goals where $X_1 + \ldots + X_k \xrightarrow{w} E_1 + \ldots + E_k$, $F \xrightarrow{w} F'$, $|w| = M$ and $n(F') > 0$. Therefore $|E_i| \leq M + 1$ for each i. If F contains recursive nonterminals then it has head form $\beta_1G_1 + \ldots + \beta_nG_n$ where $|\beta_i| = M + 1$ and $G_i \neq \epsilon$ or $|\beta_i| < M + 1$ and G_i is a recursive nonterminal V_j. If F does not contain recursive nonterminals then it has a similar head form $\beta_1G_1 + \ldots + \beta_{n-1}G_{n-1} + JG_n$ where $|\beta_i| = M + 1$ and $|J| < M + 1$ and $G_n = \epsilon$. Assume that F does not contain recursive nonterminals (the other case is almost similar). Let $(\beta_i \cdot w)$ be F'_i and $(J \cdot w)$ be F'_n and therefore $|F'_i| \leq 2M + 1$ for each i. Because $F' = F \cdot w$ the second goal is $E_1H_1 + \ldots + E_kH_n = F'_1G_1 + \ldots + F'_nG_n$. If the two starting goals are true then Proposition 1 justifies reduction to the subgoal

$$E_1(F \cdot w(X_1)) + \ldots + E_k(F \cdot w(X_k)) = F'_1G_1 + \ldots + F'_nG_n$$

Because $|w(X_i)| < M$ for each i, the left hand side configuration has the following matrix form

$$\begin{array}{ccc} E_1(\beta_1 \cdot w(X_1))G_1 + \ldots + E_1(J \cdot w(X_1))G_n & + \\ \vdots & \vdots \\ E_k(\beta_1 \cdot w(X_k))G_1 + \ldots + E_k(J \cdot w(X_k))G_n & \end{array}$$

where $|E_i(\beta_j \cdot w(X_i))| \leq 3M + 2$, and $|E_i(J \cdot w(X_i))| < 3M + 2$. Collecting terms together the subgoal therefore has the tail form $E'_1G_1 + \ldots + E'_nG_n = F'_1G_1 + \ldots + F'_nG_n$ where the maximum imbalance is $3M + 2$.

Next we need to show that introducing balanced subgoals is "sound" (there is a fuller discussion of soundness in the next section). The next Proposition shows that if the starting goals are not true then falsity is preserved when introducing a balanced subgoal.

Proposition 2 *If* $n \geq m$ *and* $m \geq M - 1$ *and* $E_1H_1 + \ldots + E_kH_k \not\sim_{n-m} F'$ *and* $X_1H_1 + \ldots + X_kH_k \sim_n F$ *then* $E_1(F \cdot w(X_1)) + \ldots + E_k(F \cdot w(X_k)) \not\sim_{n-m} F'$.

Proof: Assume $n \geq m \geq M - 1$ and $X_1H_1 + \ldots + X_kH_k \sim_n F$. Therefore $H_i \sim_{n-|w(X_i)|} (F \cdot w(X_i))$. Because $|w(X_i)| \leq M - 1$, $H_i \sim_{n-m} (F \cdot w(X_i))$. Therefore $E_1H_1 + \ldots + E_kH_k \sim_{n-m} E_1(F \cdot w(X_1)) + \ldots + E_k(F \cdot w(X_k))$. Assume

$E_1H_1 + \ldots + E_kH_k \not\sim_{n-m} F'$. By Fact 3 iii of the previous section the result follows. $\Box$

Bounding imbalance between configurations is not enough for showing decidability. The sizes of subgoals may keep growing. The next and crucial step in the argument is a mechanism for controlling size. It is at this point that we appeal to recursive nonterminals. The balanced goal,

$$(1)\quad E_1G_1 + \ldots + E_nG_n = F_1G_1 + \ldots + F_nG_n$$

where the E_is and F_is do not contain recursive nonterminals, can be reduced to a subgoal of the form

$$(2)\quad E_1V_1 + \ldots + E_nV_n = F_1V_1 + \ldots + F_nV_n$$

where $(V_1, \ldots, V_n)$ is a family of recursive nonterminals. The mechanism for reducing goal (1) to goal (2) involves constructing the recursive family $(V_1, \ldots, V_n)$ from a subsidary family of goals, $E_1^iG_1 + \ldots + E_n^iG_n = F_1^iG_1 + \ldots + F_n^iG_n$ where $i \geq 1$, with the same tails as (1).

Proposition 3 *If $k \geq 1$ and $E_1^iG_1 + \ldots + E_n^iG_n \sim F_1^iG_1 + \ldots + F_n^iG_n$ for each $i : 1 \leq i \leq k$ and every E_j^i and F_j^i does not contain recursive nonterminals then there is a family of recursive nonterminals $(V_1, \ldots, V_n)$ such that*

1. $E_1^iV_1 + \ldots + E_n^iV_n \sim F_1^iV_1 + \ldots + F_n^iV_n$ *for each* $i : 1 \leq i \leq k$,
2. *if* $V_i \stackrel{\text{def}}{=} H_1V_1 + \ldots + H_nV_n$ *then* $G_i \sim H_1G_1 + \ldots + H_nG_n$.
3. *if* $V_i \stackrel{\text{def}}{=} V_j$ *then* $G_i \sim G_j$.

Proof: The proof proceeds by iteratively refining families of recursive nonterminals for each $E_1^iG_1 + \ldots + E_n^iG_n \sim F_1^iG_1 + \ldots + F_n^iG_n$ in order starting with $i = 1$. Let E be $E_1^1G_1 + \ldots + E_n^1G_n$ and let F be $F_1^1G_1 + \ldots + F_n^1G_n$. For the base case $V_i^0 \stackrel{\text{def}}{=} V_i^0$, $1 \leq i \leq n$. Clearly 2 and 3 hold for each V_i^0. Assume that the jth family $(V_1^j, \ldots, V_n^j)$, $j \geq 0$, is given and that 2 and 3 hold for each V_i^j. Let E' be $E_1^1V_1^j + \ldots + E_n^1V_n^j$ and let F' be $F_1^1V_1^j + \ldots + F_n^1V_n^j$. If $E' \sim F'$ then we have dealt with the first equation. Now let E be $E_1^2G_1 + \ldots + E_n^2G_n$ and F be $F_1^2G_1 + \ldots + F_n^2G_n$ and let E' be $E_1^2V_1^j + \ldots + E_n^2V_n^j$ and let F' be $F_1^2V_1^j + \ldots + F_n^2V_n^j$. If $E' \sim F'$ then we have dealt with the second equation too. We keep repeating this until either all the equations are exhausted (and then $(V_1^j, \ldots, V_n^j)$ is the required family of recursive nonterminals) or E is $E_1^lG_1 + \ldots + E_n^lG_n$ and F is $F_1^lG_1 + \ldots + F_n^lG_n$ and E' is $E_1^lV_1^j + \ldots + E_n^lV_n^j$ and and F' is $F_1^lV_1^j + \ldots + F_n^lV_n^j$ and $E' \not\sim_k F'$ for a least k. Let u be the smallest distinguishing word for E' and F'. There are two possibilities. First that one and only one of $(E' \cdot u)$ and $(F' \cdot u)$ is $\emptyset$. Second is that just one of this pair is a particular terminating nonterminal V_i^j. We show below that the first possibility is impossible because $E \sim F$. In the case of the second possibility we refine the family of recursive nonterminals to $(V_1^{j+1},\ldots,V_n^{j+1})$ where each V_i^{j+1} obeys conditions 2 and 3. By Fact 2 of section 3, $E_1^i V_1^{j+1}+\ldots+E_n^i V_n^{j+1} \sim F_1^i V_1^{j+1}+\ldots+F_n^i V_n^{j+1}$ for all $i < l$. Hence we continue the construction for E is $E_1^l G_1+\ldots+E_n^l G_n$ and F be $F_1^l G_1+\ldots+F_n^l G_n$ and E' is $E_1^l V_1^{j+1}+\ldots+E_n^l V_n^{j+1}$ and F' is $F_1^l V_1^{j+1}+\ldots+F_n^l V_n^{j+1}$.

We now examine the case when $E' \not\sim_k F'$ and $u = a_1\ldots a_k$ is the smallest distinguishing word. Consider the following four sequences when Z is E', F', E and F respectively

$$(Z\cdot a_1),\ldots,(Z\cdot a_1\ldots a_i),\ldots,(Z\cdot a_1\ldots a_k)$$

Consider the initial part of the sequence in the case Z is E' up to the first prefix, if there is one, $u_1 = a_1\ldots a_m$ such that $Z\cdot u_1 = E''$ where $E'' = H_1V_1^j+\ldots+H_nV_n^j$ and $(E'\cdot a_1\ldots a_{m-1}) \xrightarrow{a_m} V_i^j$. From 2 we know that $G_i \sim H_1G_1+\ldots+H_nG_n$ because $V_i^j \stackrel{\text{def}}{=} E''$. The initial part of the sequence when Z is E up to $E\cdot a_1\ldots a_{m-1}$ is similar to the initial part for Z is E' in that they have the same "heads". Consequently $E\cdot u_1 = G_i$. Therefore the sequence for Z is E is updated from position m to k. Let $E\cdot a_1\ldots a_s$, for $s \geq m$, be $(H_1G_1+\ldots+H_nG_n)\cdot a_{m+1}\ldots a_s$. This updating restores the same heads in the two sequences Z is E and Z is E' until the next occurrence of a $G_{i'}$ in the updated sequence for Z is E. We repeatedly update the new sequence for Z is E whenever there is a later position $E'\cdot a_1\ldots a_t = H_1'V_1^j+\ldots+H_n'V_n^j$ and $V_{i'}^j \stackrel{\text{def}}{=} H_1'V_1^j+\ldots+H_n'V_n^j$ and $E\cdot a_1\ldots a_t$ in the (updated) sequence is $G_{i'}$ for $t < k$. The same updating construction is applied to the sequences when Z is F' and Z is F. Note that repeated updating of the sequences for E and F does not affect the property that their corresponding positions are equivalent.

The final positions of the sequences for E' and F' are the elements $E'\cdot u$ and $F'\cdot u$. If one of them is $\emptyset$ then one of the final positions of the updated sequences for E and F is also $\emptyset$, which would contradict that $E \sim F$. Therefore one of them is a terminating recursive nonterminal V_i^j. Without loss of generality assume that $E'\cdot u = V_i^j$. Consider the final element of the updated sequence for E. It is either G_i or G_t when $V_t^j \stackrel{\text{def}}{=} V_i^j$ (and $i \leq t$). In the second case by 2, $G_t \sim G_i$. Consider now the final element $F'\cdot u$.

The first case is that $F'\cdot u$ is $H_1'V_1^j+\ldots+H_n'V_n^j$ and in the updated sequence $F\cdot u$ is $H_1'G_1+\ldots+H_n'G_n$ (where no H_i is ϵ). Because $E \sim F$ it follows that $G_i \sim H_1'G_1+\ldots+H_n'G_n$. The family $(V_1^j,\ldots,V_n^j)$ is refined to $(V_1^{j+1},\ldots,V_n^{j+1})$ as follows. First $V_i^{j+1} \stackrel{\text{def}}{=} H_1'V_1^{j+1}+\ldots+H_n'V_n^{j+1}$. Next for any index t such that $V_t^j \stackrel{\text{def}}{=} V_i^j$ let $V_t^{j+1} \stackrel{\text{def}}{=} H_1'V_1^{j+1}+\ldots+H_n'V_n^{j+1}$. For the other entries we merely update the index j to $j+1$ on the V_i^js on both sides of $\stackrel{\text{def}}{=}$. By construction properties 2 and 3 both hold for the new family $(V_1^{j+1},\ldots,V_n^{j+1})$.

The second case is that $F'\cdot u = V_{i'}^j$. Therefore the final element $F\cdot u$ in the updated sequence is either $G_{i'}$ or $G_{t'}$ such that $G_{t'} \sim G_{i'}$ where $V_{t'}^j \stackrel{\text{def}}{=} V_{i'}^j$ and $i' \leq t'$. Because $E'\cdot u = V_i^j$ we know that $i \neq i'$ since u distinguishes

E' and F'. However $G_i \sim G_{i'}$ because $E \sim F$. Consider $\min\{i, i'\}$. Without loss of generality assume it is i'. The refined family of recursive nonterminals $(V_1^{j+1}, \ldots, V_n^{j+1})$ is defined as follows. First $V_i^{j+1} \stackrel{\text{def}}{=} V_{i'}^{j+1}$. Secondly for any index t such that $V_t^j \stackrel{\text{def}}{=} V_i^j$ let $V_t^{j+1} \stackrel{\text{def}}{=} V_{i'}^{j+1}$. For the rest of the entries we just update the index j to $j+1$ as in the first case. By construction, properties 2 and 3 hold for the new family of recursive nonterminals.

The stages of the construction produce a sequence of families of recursive nonterminals $(V_1^0, \ldots, V_n^0), \ldots, (V_1^j, \ldots, V_n^j), \ldots$ where each family refines the previous family. The final step in the proof is that the iteration must terminate by stage $n-1$. At each stage j exactly one terminating nonterminal V_i^j is directly refined. Other elements V_t^j when $V_t^j \stackrel{\text{def}}{=} V_i^j$ and $t > i$ may also be refined. No element V_i^k with index i is directly refined more than once. Therefore by stage $n-1$ the iteration must terminate with the family $(V_1^{n-1}, \ldots, V_n^{n-1})$. Now it is a simple argument that if property 1 does not hold by stage $n-1$ then afterall $E \not\sim F$ for the then current E and F. $\square$

The recursive family $(V_1, \ldots, V_n)$ as constructed in the proof of Proposition 3 is said to be "canonical" for the family $E_1^i G_1 + \ldots + E_n^i G_n = F_1^i G_1 + \ldots + F_n^i G_n$ of true goals. The construction of canonical nonterminals is independent of the tails G_i.

Fact 1 *If $(V_1, \ldots, V_n)$ is canonical for $E_1^i G_1 + \ldots + E_n^i G_n \sim F_1^i G_1 + \ldots + F_n^i G_n$ then it is also canonical for the family $E_1^i J_1 + \ldots + E_n^i J_n \sim F_1^i J_1 + \ldots + F_n^i J_n$, where $i : 1 \leq i \leq k$.*

The assembly of a canonical family proceeds in stages. Each recursive family $(V_1^{j+1}, \ldots, V_n^{j+1})$ refines $(V_1^j, \ldots, V_n^j)$. As the construction must terminate by stage $j = n-1$, at most n of the goals $E_1^i G_1 + \ldots + E_n^i G_n = F_1^i G_1 + \ldots + F_n^i G_n$ are used in the refinement process. The other goals play no role. The building of the V_i^{j+1}s from the V_i^js appeals to the smallest distinguishing word u_{j+1} for $E' \not\sim F'$ (when E' is $E_1^l V_1^j + \ldots + E_n^l V_n^j$ and F' is $F_1^l V_1^j + \ldots + F_n^l V_n^j$). We have no insight as to the upper bound on $|u_{j+1}|$. For instance it is not determined by the maximum norm of the heads E_i^l and F_i^l. Indeed this turns out to be the reason why the procedure for equivalence of the next section is only semi-decidable.

Next we wish to show that introducing canonical recursive nonterminals is "sound" for a family of goals (see the next section for a fuller discussion of soundness). We need to consider how to introduce recursive nonterminals when the family of goals need not all be true. The idea is to approximate canonicity by defining when a recursive family $(V_1, \ldots, V_n)$ is "canonical to depth d" where $d \geq 0$, for a family of goals $E_1^i G_1 + \ldots + E_n^i G_n = F_1^i G_1 + \ldots + F_n^i G_n$ when $E_1^i G_1 + \ldots + E_n^i G_n \sim_m F_1^i G_1 + \ldots + F_n^i G_n$ for $m > d$ for each i. The construction is the same as in Proposition 3, except that we stop at the first stage $j \geq 0$ with $(V_1^j, \ldots, V_n^j)$ as the required family of recursive nonterminals if the sum of the distinguishing words $s_j = |u_1| + \ldots + |u_j|$ is no larger than d, and $E' \sim_{d-s_j} F'$

for the current E' and F'. A goal $E = F$ is "m-true" if $E \sim_m F$. Given m it is decidable whether a finite family of goals are m-true. Furthermore given $d < m$ it is decidable whether family $(V_1, \ldots, V_n)$ is canonical to depth d for a finite family $E_1^i G_1 + \ldots + E_n^i G_n \sim_m F_1^i G_1 + \ldots + F_n^i G_n$. If $(V_1, \ldots, V_n)$ is canonical to depth d for a family of goals which are m-true then it is also canonical to depth d for that family for any $k > m$, provided they are k-true. Being canonical to depth d is independent of the tails: if $(V_1, \ldots, V_n)$ is canonical to depth d for $E_1^i G_1 + \ldots + E_n^i G_n \sim_m F_1^i G_1 + \ldots + F_n^i G_n$ then it is also canonical to depth d for the family $E_1^i J_1 + \ldots + E_n^i J_n \sim_m F_1^i J_1 + \ldots + F_n^i J_n$ with the same heads but different tails. Moreover if $(V_1, \ldots, V_n)$ is canonical for a family of (true) goals then there is a smallest d for which it is canonical to depth d. This motivates the next result, which guarantees that introduction of recursive nonterminals is "sound".

Proposition 4 *If $0 < k < n$ and $0 < d < m$ and $(V_1, \ldots, V_n)$ is canonical to depth d for the family of goals $E_1^i G_1 + \ldots + E_n^i G_n = F_1^i G_1 + \ldots + F_n^i G_n$ where $E_1^i G_1 + \ldots + E_n^i G_n \sim_m F_1^i G_1 + \ldots + F_n^i G_n$ for each $i : 1 \leq i \leq k$, and $E_1 G_1 + \ldots + E_n G_n \not\sim_{m-d} F_1 G_1 + \ldots + F_n G_n$ then $E_1 V_1 + \ldots + E_n V_n \not\sim_{m-d} F_1 V_1 + \ldots + F_n V_n$.*

Proof: Assume $(V_1, \ldots, V_n)$ is canonical to depth d where $d < m$ for the family of goals $E_1^i G_1 + \ldots + E_n^i G_n = F_1^i G_1 + \ldots + F_n^i G_n$, and that $E_1^i G_1 + \ldots + E_n^i G_n \sim_m F_1^i G_1 + \ldots + F_n^i G_n$ for each $i : 1 \leq i \leq k$. By construction of the canonical nonterminals to depth d, it follows that if $V_i \stackrel{\text{def}}{=} H_1 V_1 + \ldots + H_n V_n$ then $G_i \sim_{m-d} H_1 G_1 + \ldots + H_n G_n$ and if $V_i \stackrel{\text{def}}{=} V_j$ then $G_i \sim_{m-d} G_j$. Next assume that $E \not\sim_{m-d} F$ when E is $E_1 G_1 + \ldots + E_n G_n$ and F is $F_1 G_1 + \ldots + F_n G_n$, but $E' \sim_{m-d} F'$ when E' is $E_1 V_1 + \ldots + E_n V_n$ and F' is $F_1 V_1 + \ldots + F_n V_n$. Consider the smallest word $u = a_1 \ldots a_k$ which distinguishes E and F. Note that $E \cdot a_1 \ldots a_i \not\sim_{m-(d+i)} F \cdot a_1 \ldots a_i$ and $E' \cdot a_1 \ldots a_i \sim_{m-(d+i)} F' \cdot a_1 \ldots a_i$. Consider the following four sequences when Z is E, F, E' and F': $(Z \cdot a_1), \ldots (Z \cdot a_1 \ldots a_{k'})$ where either $k' = k$ or $k' < k$ and the final elements for the sequences for Z is E' and F' is a terminating nonterminal V_i. The idea is as in the proof of Proposition 3 to update the sequences for Z is E and Z is F so that they have the same heads as as those for Z is E' and Z is F'. Consider the initial prefix $u_1 = a_1 \ldots a_i$ of Z is E', if there is one, such that $Z \cdot u_1 = E''$ and $E'' = H_1 V_1 + \ldots + H_n V_n$ and $(E' \cdot a_1 \ldots a_{i-1}) \xrightarrow{a_i} V_j$. Hence $E \cdot u_1 = G_j$. Because $V_j \stackrel{\text{def}}{=} E''$ the following hold: $G_j \sim_{m-d} H_1 G_1 + \ldots + H_n G_n$ and $E \cdot u_1 = G_j \not\sim_{m-(d+i)} F \cdot u_1$. Therefore by Fact 3 iii of section 3, $H_1 G_1 + \ldots + H_n G_n \neq_{m-(d+i)} F \cdot u_1$. Therefore the sequence for Z is E is updated from position i to k': $E \cdot a_1 \ldots a_s$, $s > i$, becomes $(H_1 G_1 + \ldots + H_n G_n) \cdot a_{i+1} \ldots a_s$. This updating restores the same heads in the two sequences for Z is E and Z is E' until the next occurrence of a $G_{i'}$ in the first sequence in which case we then again update it. The same updating construction is applied to the sequence Z is F using Z is F'. The repeated updating of the sequences for E and F does not affect the property that their corresponding positions j are inequivalent at $m-(d+j)$. Consider now the final elements in the updated sequences for E and F. The first case is that one and only one of the elements is $\emptyset$, but then one and only one of the corresponding elements in the sequences for E' and F' is also $\emptyset$ which is a contradiction. The second case is that one of the elements, say in the sequence for E, is a terminating recursive nonterminal U_j, which means that some G_i is U_j. But then the corresponding element in the sequence for E' is also a terminating nonterminal V_i (because $G_i \sim_{m-d} H$ iff H is G_i). Therefore the corresponding element in the sequence for F' is also V_i and so the final element in the sequence for F is U_j as well. $\Box$

5 Tableaux

The proof of decidability is completed by presenting a tableau proof system for demonstrating equivalence of admissible configurations. The proof system is goal directed and consists of two kinds of rules, "simple" and "conditional". Simple rules have the form

$$\frac{\text{Goal}}{\text{Subgoal}_1 \ \ldots \ \text{Subgoal}_n}\ \mathcal{C}$$

where Goal is what currently is to be proved and the subgoals are what it reduces to, provided the side condition $\mathcal{C}$ holds. A conditional rule has the form

$$\begin{array}{c} \text{Goal}_1 \\ \vdots \\ \text{Goal}_k \\ \vdots \\ \dfrac{\text{Goal}}{\text{Subgoal}} \end{array}\ \mathcal{C}$$

where Goal is the current goal to be shown and there is a single subgoal to which it reduces provided that the goals $\text{Goal}_1,\ldots,\text{Goal}_k$ occur above Goal on the path between it and the root (starting goal) and provided that the side condition $\mathcal{C}$ holds. Goals and subgoals are all of the form $E = F$ where E and F are (normed) admissible configurations which may contain recursive nonterminals.

There is also the important notion of when a current goal counts as final. Final goals are classified as either successful or unsuccessful. A *tableau proof* for a starting Goal is a finite proof tree, whose root is Goal and all of whose leaves are successful final goals, and all of whose inner subgoals are the result of an application of one of the rules. It is our intention to show that $E \sim F$ iff there is a tableau proof for $E = F$.

There is one simple rule UNF presented in Figure 2. A goal $E = F$ reduces

UNF

$$\frac{E = F}{E \cdot a_1 = F \cdot a_1 \quad \ldots \quad E \cdot a_k = F \cdot a_k} \; \mathcal{A} = \{a_1, \ldots, a_k\}$$

Figure 2: Simple tableau rule

to the subgoals $E \cdot a_i = F \cdot a_i$ for each $a_i \in \mathcal{A}$. UNF obeys local completeness and soundness. Completeness is that if the goal is true then so are all the subgoals. This is clear from Fact 1 i of section 3. Soundness is that if all the subgoals are true then so is the goal, or equivalently if the goal is false then so is at least one of the subgoals. A finer version uses approximants, which provide a measure of how false a goal $E = F$ is. Consider the smallest n such that $E \not\sim_n F$. For UNF if the goal is false at $n+1$, $E \not\sim_{n+1} F$, then at least one of the subgoals is false at n, $E \cdot a \not\sim_n F \cdot a$, see Fact 3 i of section 3.

The conditional rules are given in Figure 3. The BAL rules introduce "balance" between goals, and CUT introduces recursive nonterminals. Completeness for BAL is that if the premise goals (those above the subgoal) are true then so is the subgoal which follows from Proposition 1 of the previous section. The statement of completeness for CUT is that there are correct applications of it. If $(V_1, \ldots, V_n)$ is canonical for the first k premises then there is a depth d for which it is canonical. Moreover $(V_1, \ldots, V_n)$ needs to be a recursive family for the true goal $E_1G_1 + \ldots + E_nG_n = F_1G_1 + \ldots + F_nG_n$, in which case the subgoal follows[6].

For soundness of the conditional rules consider global soundness of the proof system. The overall idea is that if there is a successful tableau whose root is false then there is a path through the tableau within which each subgoal is false. The idea is refined using approximants. If the root is false then there is an offending path (of false goals) through the tableau within which the approximant indices decrease whenever rule UNF has been applied, and hence this would mean that a successful final goal is false (which, as we shall show, is impossible). Soundness of the conditional rules is that if the premises are on an offending path then the subgoal preserves the falsity index of the goal immediately above it. In the case of BAL(R) assume that the offending path passes through the premise goals. There is a least n such that for the initial premise $F \sim_n X_1H_1 + \ldots + X_kH_k$ and $F \not\sim_{n+1} X_1H_1 + \ldots + X_kH_k$. As there are exactly M applications of UNF between the initial and final premise it follows that $F' \sim_{n-M} E_1H_1 + \ldots + E_kH_k$. However, as this is the offending path $F' \not\sim_{(n-M)+1} E_1H_1 + \ldots + E_kH_k$. Therefore by Proposition 2 of the previous section $F' \not\sim_{(n-M)+1} E_1(F \cdot w(X_1)) + \ldots + E_k(F \cdot w(X_k))$. The same argument proves soundness of BAL(L). There is a similar

[6]If $(V_1, \ldots, V_n)$ is canonical for the first k premises and $E_1V_1 + \ldots + E_nV_n \not\sim F_1V_1 + \ldots + F_nV_n$ then $(V_1, \ldots, V_n)$ can be refined to $(V_1', \ldots, V_n')$ so it is canonical for all the premises. Because there can only be at most $n-1$ refinements, eventually CUT will be applicable if goals with the common tails G_i persist: see the later discussion.

BAL(R)

$$\frac{\begin{array}{c} F = X_1H_1 + \ldots + X_kH_k \\ \vdots \\ F' = E_1H_1 + \ldots + E_kH_k \end{array}}{F' = E_1(F \cdot w(X_1)) + \ldots + E_k(F \cdot w(X_k))} \quad \mathcal{C}_1$$

BAL(L)

$$\frac{\begin{array}{c} X_1H_1 + \ldots + X_kH_k = F \\ \vdots \\ E_1H_1 + \ldots + E_kH_k = F' \end{array}}{E_1(F \cdot w(X_1)) + \ldots + E_k(F \cdot w(X_k)) = F'} \quad \mathcal{C}_1$$

where $\mathcal{C}_1$ is the condition

1. There are precisely M applications of UNF between the top goal and the bottom goal, and no application of any other rule.

CUT

$$\frac{\begin{array}{c} E_1^1G_1 + \ldots + E_n^1G_n = F_1^1G_1 + \ldots + F_n^1G_n \\ \vdots \\ E_1^kG_1 + \ldots + E_n^kG_n = F_1^kG_1 + \ldots + F_n^kG_n \\ \vdots \\ E_1G_1 + \ldots + E_nG_n = F_1G_1 + \ldots + F_nG_n \end{array}}{E_1V_1 + \ldots + E_nV_n = F_1V_1 + \ldots + F_nV_n} \quad \mathcal{C}_2$$

where $\mathcal{C}_2$ is the condition

1. No E_j^i or F_j^i contains recursive nonterminals.

2. $(V_1, \ldots, V_n)$ is canonical to depth d for the goals $E_1^iG_1 + \ldots + E_n^iG_n = F_1^iG_1 + \ldots + F_n^iG_n$, $1 \leq i \leq k$ and $k \leq n$.

3. There are at least $d+1$ applications of UNF between the goal $E_1^kG_1 + \ldots + E_n^kG_n = F_1^kG_1 + \ldots + F_n^kG_n$ and $E_1G_1 + \ldots + E_nG_n = F_1G_1 + \ldots + F_nG_n$ (as well as possibly other rules).

Figure 3: Conditional tableau rules

Successful final goals

$$
\begin{array}{ccc}
 & E = F & \\
 & \vdots & \text{UNF} \\
 & \vdots & \text{at least once} \\
1.\ E = E & 2.\ E = F &
\end{array}
$$

Unsuccessful final goals

1. $E = F$ when $\mathrm{n(E)} \neq \mathrm{n(F)}$

2. $V_i = V_j$ when $i \neq j$ and $V_i \stackrel{\text{def}}{=} V_i$, and $V_j \stackrel{\text{def}}{=} V_j$

Figure 4: Final goals

argument for CUT. Consider $m > d$ such that $E_1^i G_1 + \ldots + E_n^i G_n \sim_m F_1^i G_1 + \ldots + F_n^i G_n$ for each $i : 1 \leq i \leq k$ and $E_1^k G_1 + \ldots + E_n^k G_n \not\sim_{m+1} F_1^k G_1 + \ldots + F_n^k G_n$. There are at least $d+1$ applications of UNF between the kth premise and the final premise of the rule, and so as this is an offending path $E_1 G_1 + \ldots + E_n G_n \not\sim_{m-d} F_1 G_1 + \ldots + F_n G_n$. Therefore by Proposition 4, $E_1 V_1 + \ldots + E_n V_n \not\sim_{m-d} F_1 V_1 + \ldots + F_n V_n$.

Final goals are presented in Figure 4. Unsuccessful goals are clearly false. A final goal is successful if it is either an identity or a repeat. An offending path of false goals with decreasing falsity indices cannot include either kind of successful goal. Clearly it is not possible for $E \not\sim_m E$. For the other case, suppose the offending path passes through $E = F$ twice. At the first instance there is an m, $E \sim_m F$ and $E \not\sim_{m+1} F$, but as there is at least one application of UNF between the two occurrences this would imply that $E \not\sim_m F$, which is a contradiction.

The first main result is that a successful tableau for $E = F$ indeed constitutes a proof that $E \sim F$.

Theorem 1 *If there is a successful tableau for $E = F$ then $E \sim F$.*

Proof: Suppose there is a successful tableau for $E = F$ but $E \not\sim F$. Then there is a least n such that $E \not\sim_n F$. We now construct an offending path of false goals through the tableau within which the approximant indices decrease whenever UNF is applied. But this is impossible, for we must reach a successful final goal as the tableau is finite. □

More intricate is the proof of the converse of Theorem 1, that if $E \sim F$ then there is a successful tableau for $E = F$. Given a true goal one applies the rules, preserving truth, according to the strategy described below. It is therefore not possible to reach an unsuccessful final goal. Thus the main issue is how to guarantee that the tableau construction is finite. We show that on any infinite path of goals developed using the strategy there must be infinitely many successful final goals.

We start with a simple observation.

(1) For any $m \geq 0$, there are only finitely many different goals $E = F$ (whose recursive nonterminals belong to $(V_1, \ldots, V_n)$) with $|E| \leq m$ and $|F| \leq m$.

If F has recursive nonterminals in the family $(V_1, \ldots, V_n)$ then we let $\mathrm{rec}(F)$ be the size of the largest definition in the family, $\max\{|H| : V_i \stackrel{\mathrm{def}}{=} H\}$. The next observation tells us how much a configuration can increase in size through an application of UNF.

(2) For any a, $|E \cdot a| \leq \max\{\mathrm{rec(E)}, |\mathrm{E}| + 1\}$.

The size of an application of BAL is the size of the configuration F in the initial goal of the rule (see Figure 3), and the application is said to use the configuration F. The resulting subgoal contains the configuration $E_1(F \cdot w(X_1)) + \ldots + E_k(F \cdot w(X_k))$. E_i is a "head" of an application and $(F \cdot w(X_i))$ is a "tail". The size of a head is bounded, $|E_i| \leq M + 1$ (using (2)). Moreover because $E_1 + \ldots + E_k$ is itself admissible if $E_i(F \cdot w(X_i)) \xrightarrow{u} (F \cdot w(X_i))$ then $(E_j(F \cdot w(X_j)) \cdot u) = \emptyset$ for $j \neq i$. A further observation about BAL (which uses (2)) is as follows.

(3) If $E' = F'$ is the result of an application of BAL of size m then configurations $|E'|$, $|F'| \leq k + 2M$, where $k = \max\{m, \mathrm{rec}(E')\}$.

A configuration F without recursive nonterminals is "small" if $|F| \leq M^2 + 4M+1$, and F with recursive nonterminals is small if $|F| \leq \mathrm{rec}(F)+M^2+4M+1$. The strategy is to apply the BAL rules wherever possible when the sizes of their applications are small, and otherwise to apply UNF. The rule CUT is not applied. Any infinite path of goals containing infinitely many small applications of BAL, and no application of CUT, must therefore contain infinitely many final goals ("repeats") by properties (3) and (1).

Next suppose there is an application of BAL which uses a large F of size m. The strategy is to build a "block". Assume that it is an application of BAL(L).

$$\begin{array}{lrl} & & F \\ & & \vdots \;\; \mathrm{BAL(L)} \\ (*) \qquad & E_1(F \cdot w(X_1)) + \ldots + E_k(F \cdot w(X_k)) = & F' \end{array}$$

F is the "root configuration" of the block and $(*)$ is its "root goal" (which will also be a potential root, the initial premise, of an application of CUT). Once a block is initiated with BAL(L), the strategy is to repeatedly apply BAL(L) wherever possible, and UNF otherwise[7]. However BAL(R) is permitted, once

$$
\begin{array}{rcll}
 & & F'' & \\
 & & \vdots & \text{BAL(L)} \\
E'_1(F'' \cdot w(X'_1)) + \ldots + E'_{k'}(F'' \cdot w(X'_{k'})) & = & H & \\
\vdots & & \vdots & \text{UNFs} \\
(F'' \cdot w(X'_i)) = G_1 & = & H_1 & \\
\vdots & & \vdots & \\
G_k & = & H_k &
\end{array}
$$

Figure 5: A potential switch from BAL(L) to BAL(R)

the "tail" of an application of BAL(L) is exposed, see Figure 5. Assume an application of BAL(L) using F''. Between its result and the goal $G_1 = H_1$ there are no further applications of BAL(L), and G_1 is a tail of the BAL application. BAL(R) is now permitted provided it uses configuration G_i, $i \geq 1$. BAL(R) is not permitted using a configuration from a goal above $G_1 = H_1$. BAL(R) is not enforced, for one can still apply BAL(L). The strategy is always to apply a BAL rule whenever it is permitted. If BAL(R) is applied then the strategy is to repeatedly apply BAL(R), and to use UNF otherwise. BAL(L) is only permitted once a tail $(F''\cdot w(X'_i))$ of an application of BAL(R) is the right hand configuration of a goal. Thus a block consists of alternating sub-blocks of BAL(L)s and UNFs and BAL(R)s and UNFs. If a later application of BAL is smaller than m then either a new block with a smaller root configuration is initiated or the size of the application is small and the earlier strategy applies.

Assume a root configuration F of size m with block root $E_1 = F_1$. Let π be a path of goals $E_1 = F_1, \ldots, E_l = F_l, \ldots$ belonging to the block developed from $E_1 = F_1$ using the strategy. All applications of BAL have at least size m in this path. We show the following crucial property.

(4) For every G which is used in an application of BAL in π there is a word u such that G is $(F \cdot u)$ and $|(F \cdot v)| > m - (M^2 + 3M)$ for all prefixes v of u.

Property (4) holds for initial root configuration F because F is $F \cdot \epsilon$ and $|F| = m$. Assume the block is initiated with a BAL(L). Consider a later application of BAL(L) using F' (where there are no intervening applications of BAL(R)) as depicted in the left derivation of Figure 6. F' arises from F via applications of UNF (and possibly BAL(L)). Consequently there is a word u, constructed from the applications of UNF, with $F' = (F \cdot u)$, and by assumption $|F'| \geq m$. For every prefix v of u, $(F \cdot v)$ is a configuration on the path between F and F'.

[7] If BAL(R) initiates the block then the strategy is to repeatedly apply BAL(R) and UNF.

$$
\begin{array}{rcll}
 & & F & \\
 & & \vdots & \text{BAL(L)} \\
E_1 & = & F_1 & \\
 & & \vdots & \\
 & & F' & \\
 & & \vdots & \text{BAL(L)} \\
E_n & = & F_n & \\
 & & \vdots &
\end{array}
\qquad\qquad
\begin{array}{rcll}
 & & F' & \\
 & & \vdots & \text{BAL(L)} \\
E_j & = & F_j & \\
\vdots & & & \\
(F' \cdot w(X_i)) & & & \\
\vdots & & & \\
E' & & & \\
\vdots & & & \text{BAL(R)} \\
E_l & = & F_l & \\
\vdots & & &
\end{array}
$$

Figure 6: Showing property (4)

Assume that for one of these configurations F'', $|F''| \leq m - (M^2 + 3M)$. There are two cases to examine.

The first case is that F'' occurs between a configuration used for a BAL and its application (between, for example, F and F_1 in Figure 6). The second case is that F'' occurs at or after an application of BAL(L), between F_1 and F' in Figure 6. Consider the first case. There are at most $M - 1$ applications of UNF between F'' and the application of BAL (because F'' cannot be the configuration used in this application). Assume that $E_1 = F_1$ is the result of this BAL(L) which uses F and that F' is the next configuration used in an application of BAL(L). Because $|F''| \leq m - (M^2 + 3M)$ and there are at most $M - 1$ applications of UNF between it and F_1, by (2) it follows that $|F_1| \leq m - (M^2 + 2M + 1)$. Because $|F'| \geq m$, there must be at least $(M^2 + 2M + 1)$ applications of UNF between F_1 and F' which are size increasing: for F_1 must increase its size and become F'. The second case is also covered by these observations: if F'' occurs between F_1 and F' then there must be at least $(M^2 + 3M)$ applications of UNF between F_1 and F' which are size increasing. However within at most $M^2 + M$ applications of UNF from F_1 a tail $(F \cdot w(X_i))$ of the application of BAL(L) must occur as the left hand configuration of a goal. E_1 has the form $E'_1(F \cdot w(X_1)) + \ldots + E'_k(F \cdot w(X_k))$ where $|E'_i| \leq M + 1$. Because BAL(L) does not apply between F_1 and F', each E'_i's size must be declining. If E_1 in head nonterminal form is $Y_1H_1 + \ldots + Y_lH_l$ then within M applications of UNF the left hand configuration must be H_j for some j, and within another M applications of UNF H_j must lose its "head" nonterminals, and so on. Consequently within $M^2 + M$ applications of UNF between F_1 and F' a goal $(F \cdot w(X_i)) = F_k$ has to occur. F_1 may have increased in size in becoming F_k but only by $M^2 + M$, and so $|F_k| \leq m - (M + 1)$. So there are still at least $M + 1$ applications of UNF between F_k and F' which are size increasing. However BAL(R) is now permitted, and clearly it must apply between F_k and F' because there must be a sequence of M UNFs where the right hand configurations are not decreasing in size. But this is a contradiction.

Next we show that property (4) continues to hold when there is a switch from BAL(L) using F' to an application of BAL(R) using E', pictured on the right in Figure 6. BAL(R) is only permitted when a tail $(F' \cdot w(X_i))$ of the application of BAL(L) occurs as a left hand configuration. By assumption there is a word u such that F' is $(F \cdot u)$. Therefore the tail $(F' \cdot w(X_i))$ is $(F \cdot u\, w(X_i))$. There are no applications of BAL between this tail and E', and therefore there is a word v such that E' is $(F \cdot u\, w(X_i)\, v)$. Moreover both F' and E' have size at least m. Assume that for some prefix v' of $w(X_i)\, v$, $|F' \cdot v'| \leq m - (M^2 + 3M)$. There are two cases to consider. First is that v' is a prefix of $w(X_i)$, and secondly that it is a prefix of the form $w(X_i)\, v''$. Because $|w(X_i)| \leq M - 1$ for the first case this means that $|(F' \cdot w(X_i))| \leq m - (M^2 + 2M + 2)$. Therefore there has to be at least $(M^2 + 2M + 2)$ applications of UNF between $(F' \cdot w(X_i))$ and E' which increase size, and for the second case there has to be at least $M^2 + 3M$ applications. However BAL(L) is still permitted between $(F' \cdot w(X_i))$ and E'. Clearly BAL(L) must therefore apply to a configuration belonging to a goal strictly above E' because there must be a sequence of M UNFs where the left hand configurations are not decreasing in size. But this is a contradiction.

The argument for (4) is now repeated for all further applications of BAL within π.

Using (4) we now establish a final property which shows that CUT eventually applies in a block. Assume a root configuration F of size m with block root $E_1 = F_1$, and assume π is a path of goals belonging to this block developed using the strategy. If F does not contain recursive nonterminals then it can be written in head form $\beta_1 G_1 + \ldots + \beta_n G_n$ where $|\beta_i| = (M^2 + 4M + 1)$ or $|\beta_i| < (M^2 + 4M + 1)$ and $G_i = \epsilon$. Notice that there is an upper bound on the "width" n (as it can be no more than the number of sequences of nonterminals whose length is at most $M^2 + 4M + 1$). In fact we can reduce the number of tails by amalgamating them when they are the same: for instance, if $G_i = G_j$ then we can have the expression $\ldots + (\beta_i + \beta_j) G_i + \ldots$. The final property for this case of F is

(5A) The result of every application of BAL within π has the form $E_1 G_1 + \ldots + E_n G_n = F_1 G_1 + \ldots + F_n G_n$ where the G_is are the tails of the root configuration F.

If F does contain recursive nonterminals drawn from $(U_1, \ldots, U_l)$ then it has a similar head form $\beta_1 G_1 + \ldots + \beta_n G_n$ where $|\beta_i| = (M^2 + 4M + 1)$ and $G_i \neq \epsilon$, or $|\beta_i| < (M^2 + 4M + 1)$ and G_i is a recursive nonterminal U_j. This is not quite enough for stating the property, for besides the tails G_i we need a (bounded) number of "supplementary" tails. Consider any β_i with $|\beta_i| < (M^2 + 4M + 1)$ and let x_1 be $\min\{M, (M^2 + 4M + 1) - |\beta_i|\}$. G_i is a recursive nonterminal U_j. If

$U_j \stackrel{\text{def}}{=} U_k$ then we include U_k as an extra tail. Otherwise $U_j \stackrel{\text{def}}{=} H_{j1}U_1 + \ldots + H_{jl}U_l$. We now put the right hand side definition of U_j into head form $\beta_{j1}G_{j1} + \ldots + \beta_{jk}G_{jk}$ where $|\beta_{ji}| = x_1$ and $G_{ji} \neq \epsilon$ or $|\beta_{ji}| < x_1$ and G_{ji} is a recursive nonterminal. We include the G_{ji}s as extra tails. Next we iterate the construction, but for a smaller size. Consider any $|\beta_{ji}| < x_1$ and let $x_2 = x_1 - |\beta_{ji}|$. Assume G_{ji} is $U_{j'}$. If $U_{j'} \stackrel{\text{def}}{=} U_{k'}$ then include $U_{k'}$ as an extra tail. Otherwise $U_{j'} \stackrel{\text{def}}{=} H$. Put H into head form, $\beta'_1 G'_1 + \ldots + \beta'_{k'} G'_{k'}$ where $|\beta'_{i'}| = x_2$ and $G'_{i'} \neq \epsilon$ or $|\beta'_{i'}| < x_2$ and $G'_{i'}$ is a recursive nonterminal. Include these $G'_{i'}$s as supplementary tails. For each $\beta'_{i'}$ such that $|\beta'_{i'}| < x_2$ repeat this construction for size $x_3 = x_2 - |\beta'_{i'}|$. And so on. Note that the sizes of the heads are decreasing $M \geq x_1 > x_2 > \ldots$. Let $G_1, \ldots, G_n$ be the primary tails of F. Here n is bounded by the number of sequences of nonterminals (excluding the recursive nonterminals) whose length is less than or equal to $M^2 + 4M + 1$. Let $G_{n+1}, \ldots, G_{n+k'}$ be the supplementary tails. Here k' is also bounded[8]. Thus the "width" $n + k'$ is bounded. As with the earlier case we can amalgamate equivalent tails. Notice that F has head form $\beta_1 G_1 + \ldots + \beta_n G_n + E_{n+1} G_{n+1} + \ldots + E_{n+k'} G_{n+k'}$ (where each E_{n+i} is $\emptyset$). The property corresponding to (5A) is as follows.

(5B) The result of every application of BAL within π has the form $E_1 G_1 + \ldots + E_{n+k'} G_{n+k'} = F_1 G_1 + \ldots + F_{n+k'} G_{n+k'}$ where the G_is are the primary and supplementary tails of the root configuration F.

Condition (5) in its two versions essentially follows from (4) and admissibility. Assume F does contain recursive nonterminals (the other case is similar but easier). The head of F, $\beta_1 + \ldots + \beta_n$ is admissible. Each β_i has the form $X^i_1 \ldots X^i_t$. Let β^j_i be the jth suffix $X^i_j \ldots X^i_t$ of β_i. By admissibility if $\beta_i \xrightarrow{w} \beta^j_i$ then either $\beta_k \xrightarrow{w} \beta^j_k$ (and $X^i_1 \ldots X^i_{j-1}$ is the same sequence as $X^k_1 \ldots X^k_{j-1}$) or $(\beta_k \cdot w) = \emptyset$. Let G be used in an application of BAL in π. By property (4) G must have the form $E_1 \beta_1^{b_1} G_1 + \ldots + E_n \beta_n^{b_n} G_n$ where the head $E_1 \beta_1^{b_1} + \ldots + E_n \beta_n^{b_n}$ is admissible, and if $|\beta_i| \geq (M^2 + 3M)$ then $b_i = M^2 + 3M$ and if $|\beta_i| < M^2 + 3M$ then $\beta_i^{b_i}$ is ϵ. Note that for at least one $|\beta_i| = (M^2 + 4M + 1)$, $E_i \neq \emptyset$. The result of BAL using G (assume it is BAL(L)) has the following form where $|u| = M$ and $|E'_i| \leq M + 1$.

$$(**) \quad E'_1(G \cdot w(X_1)) + \ldots + E'_k(G \cdot w(X_k)) = (G \cdot u)$$

It is at this point that we may need the supplementary tails to account for $(G \cdot u)$ and $(G \cdot w(X_i))$. The first case is that they only contain primary tails and have the form $E''_1 \beta_1^{b'_1} G_1 + \ldots E''_n \beta_n^{b'_n} G_n$ where if $|\beta_i| \geq (M^2 + 4M)$ then b'_i is $(M^2 + 4M)$ and if $|\beta_i| < (M^2 + 4M)$ then $\beta_i^{b'_i}$ is ϵ. The second case is that they "enter" a tail

[8] k' is at most the maximum of n and $2M$ times the number of sequences of nonterminals (not including recursive ones) whose length is less than or equal to M.

G_i when it is a recursive nonterminal and $\beta_i^{b'_i}$ is ϵ: for instance, $(G \cdot u)$ is $(U_j \cdot u')$ for some suffix u' of u (and so, $|u'| < M$). In which case $(U_j \cdot u')$ has the form $E''_1 G_{n+1} + \ldots + E''_{k'} G_{n+k'}$ (where the G_{n+i}s are supplementary tails and the heads are very small, $|E''_i| \leq 2M + 1$). This concludes the proof of (5).

Property (5) ensures that eventually in a block CUT is applicable, and that in any infinite path containing infinitely many CUTs there are infinitely many final goals. Assume that $E'_1 = F'_1$ is the root goal of a block which is the result of BAL(L) using F. This root goal is (* *) when G is F. Assume F has recursive nonterminals (the other case is similar). Therefore from (5B) the left hand configuration has the form $E_1 G_1 + \ldots + E_{n+k'} G_{n+k'}$ where $|E_i| \leq M^2 + 6M$ for $1 \leq i \leq n$ and $|E_i| \leq 3M + 2$ for $n < i \leq n + k'$, and the right hand configuration $(F \cdot u)$ has the similar form $F_1 G_1 + \ldots + F_{n+k'} G_{n+k'}$ where $|F_i| \leq M^2 + 5M + 1$ for $1 \leq i \leq n$ and $|F_i| \leq M + 1$ for $n < i \leq n + k'$. Many of the $E_i s$ and F_is may be $\emptyset$. Because both the width, $n + k'$, and the sizes of the "heads" are bounded, there can only be finitely many root goals (of a block) with different heads. Let $(V_1, \ldots, V_{n+k'})$ be the canonical family of recursive nonterminals for the root goal with depth d. Consider the first application of a BAL after $d + 1$ applications of UNF. Assume the result is $E'_2 = F'_2$ which has the form of (5B). The size of the heads are bounded[9]. If $(V_1, \ldots, V_{n+k'})$ is a recursive family for this goal then CUT can be applied. Otherwise the family is refined to give a recursive family for $E'_1 = F'_1$ and $E'_2 = F'_2$ with depth d'. The argument is repeated. There can only be at most $n + k' - 1$ refinements to the recursive family of nonterminals, and therefore CUT must eventually apply in a block. Moreover because there can only be finitely many roots of a block with different heads, in any infinite path with infinitely many applications of CUT there must be infinitely many final goals.

Theorem 2 *If $E \sim F$ then there is a successful tableau for $E = F$.*

Proof: Assume that $E \sim F$. Now we keep applying the rules preserving truth using the strategy described above. If goals are small then one keeps applying BAL(L), BAL(R) and UNF. Otherwise one tries to build a block and apply CUT. By preserving truth it is not possible to reach an unsuccessful final goal. Also it is not possible to become stuck, as UNF is always applicable unless a goal is final. Hence the only issue is that the tableau construction goes on forever. Assume that there is a an infinite path through the tableau. If CUT is only applied finitely often on this path then consider the subsequence after its final application. All

[9] The largest head in the root goal has size $M^2 + 6M$. Let G be the resulting goal after $d + 1$ applications of UNF. There can be at most $d + 1/M$ applications of BAL between the root and G. From the argument for (4) if BAL does not apply within $M^2 + M$ applications of UNF after G then both BAL rules are permitted, and if it still does not apply then the size of the configurations in the goals must be diminishing. Thus, as a crude measure, there is an upper bound of $2M^2 + (d + k)M$ where k is a constant in the size of the heads in $E'_2 = F'_2$ when it has the form in (5B).

attempts to build a block are thwarted, and therefore infinitely often there are small goals and so infinitely often there are final goals. Consequently CUT must be applied infinitely often. However by the analysis above any application of CUT is bounded independently of the tails and therefore we must introduce the same family of recursive nonterminals infinitely often to goals with the same heads, and therefore there must be infinitely many final goals. □

6 Conclusion

We have provided a manageable proof of decidability of equivalence between DPDAs. However because the procedure consists of two semi-decision procedures we are unable to provide a complexity bound. More work is needed to see if we can find a useful bound on the depth d for a canonical family of recursive nonterminals.

An intriguing open question is whether there is a more general class of context-free grammars than the strict deterministic for which language equivalence is decidable.

The proof technique developed in this paper can also be applied to decision problems for bisimulation equivalence. Language equivalence and bisimulation equivalence coincide in the deterministic case (provided there is no redundancy). In particular the technique developed here can be used to extend the result in [10] to all pushdown processes.

Acknowledgement: I am deeply indebted to Olaf Burkart and Didier Caucal for numerous discussions about DPDA, and to Geraud Sénizergues for explanations of his result.

References

[1] Christensen, S., Hüttel, H., and Stirling, C. (1995). Bisimulation equivalence is decidable for all context-free processes. *Information and Computation*, **121**, 143-148.

[2] Ginsberg, S., and Greibach, S. (1966). Deterministic context-free languages. *Information and Control*, 620-648.

[3] Harrison, M. (1978). *Introduction to Formal Language Theory*, Addison-Wesley.

[4] Harrison, M., Havel, I., and Yehudai, A. (1979). On equivalence of grammars through transformation trees. *Theoretical Computer Science*, **9**, 173-205.

[5] Hopcroft, J., and Ullman, J. (1979). *Introduction to Automata Theory, Languages, and Computation*, Addison-Wesley.

[6] Hüttel, H., and Stirling, C. (1991). Actions speak louder than words: proving bisimilarity for context free processes. *Proceedings 6th Annual Symposium on Logic in Computer Science*, IEEE Computer Science Press, 376-386.

[7] Oyamaguchi, M., Honda, N., and Inagaki, Y. (1980). The equivalence problem for real-time strict deterministic languages. *Information and Control*, **45**, 90-115.

[8] Sénizergues, G. (1997). The equivalence problem for deterministic pushdown automata is decidable. *Lecture Notes in Computer Science*, **1256**, 671-681.

[9] Sénizergues, G. (1998). L(A) = L(B)? Tech. Report LaBRI, Université Bordeaux I, pp. 1-166. (Submitted to *Theoretical Computer Science.*)

[10] Stirling, C. (1998). Decidability of bisimulation equivalence for normed pushdown processes. *Theoretical Computer Science*, **195**, 113-131.