Contribution Title

First Author[1[0000−1111−2222−3333]], Second Author[2,3[1111−2222−3333−4444]], and Third Author[3[2222−−3333−4444−5555]]

[1] Princeton University, Princeton NJ 08544, USA
[2] Springer Heidelberg, Tiergartenstr. 17, 69121 Heidelberg, Germany
lncs@springer.com
http://www.springer.com/gp/computer-science/lncs
[3] ABC Institute, Rupert-Karls-University Heidelberg, Heidelberg, Germany
{abc,lncs}@uni-heidelberg.de

Abstract. Tableaux proofs are used in automated "proofers" as (TODO), whereas sequent calculus proofs are used in proof assistants as (TODO). This work aims to discuss the translation between them and provide an algorithm for translating first-order, predicate logic tableaux proofs into sequent calculus proofs. It begins with an overview of the definitions in both intuitionistic and classical logic. It then shows a translation process in classical logic, along with its properties. Finally, a potential extension towards translation in intuitionistic logic is explored. (TODO talk about book1, "extention")

Keywords: Tableux proof · sequent calculus · intuitionistic logic.

1 Introduction

1.1 Notation

In this work, sentences will implicitly refer to first-order predicate logic sentences; for intuitionistic logic, their meaning will come from Kripke's semantics [3] . The notation for structures and frames will be (heavily based on ?) / (as in) [1]. To make this document slightly more self-reliant, we will briefly explain:

1.2 Definitions

The definitions will be given for intuitionistic logic; To avoid redundancy, classical logic will be seen as intuitionistic logic with single-framed structures. [TODO cite?]

Definition 1. ***A structure of a language*** *consists of a domain and an assignment from the constant symbols of the language to the domain and from the predicate symbols of the language to predicates in the domain.*

They represent "possible worlds" [TODO cite] in a frame:

Definition 2. ***A Kripke Frame*** *of a Language $\mathcal{L}$, $\mathcal{C} = (R, \{C(p)\}_{p \in R})$ consists of a partially ordered set R, and an $\mathcal{L}$-structure $C(p)$ for all p's in R. Furthermore, in a Kripke Frame: if $p \leq q$, then $C(q)$ extends $C(p)$: all sentences that are true in $C(p)$ are true in $C(q)$, the domain of $C(p)$ is included in the domain of $C(q)$ and the assignments in $C(p)$ are the same as in $C(q)$*

For simplicity, R will always be the set of sequences of integers, and $p \leq q$ if p is in q:

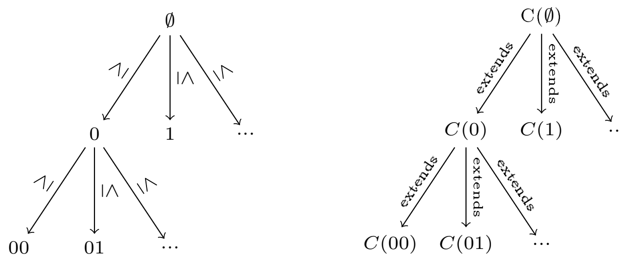

Fig. 1. R and a Kripke frame

Definition 3. ***Forcing***. *When a sentence ϕ of a language $\mathcal{L}$ is* ***forced*** *by a structure $C(p)$ of a frame $\mathcal{C}$, we denote: $p \vDash_{\mathcal{C}} \phi$*
Forcing is defined by induction: [1]

- $p \vDash_{\mathcal{C}} \phi \Leftrightarrow \phi$ *is true in $C(p)$ (if ϕ is an atomic sentence)*

- $p \vDash_{\mathcal{C}} (\phi \rightarrow \psi) \Leftrightarrow$ *for all $q \geq p$, if $q \vDash_{\mathcal{C}} \phi$, then $q \vDash_{\mathcal{C}} \psi$*
- $p \vDash_{\mathcal{C}} \neg\phi \Leftrightarrow$ *for all $q \geq p$, q does not force ϕ*

- $p \vDash_{\mathcal{C}} (\forall x)\phi(x) \Leftrightarrow$ *for all $q \geq p$ and d in $\mathcal{L}_{C(q)}$, $q \vDash_{\mathcal{C}} \phi(d)$*

- $p \vDash_{\mathcal{C}} (\exists x)\phi(x) \Leftrightarrow$ *exists a d in $\mathcal{L}_{C(q)}$, such that $p \vDash_{\mathcal{C}} \phi(d)$*

- $p \vDash_{\mathcal{C}} (\phi \wedge \psi) \Leftrightarrow p \vDash_{\mathcal{C}} \phi$ *and* $p \vDash_{\mathcal{C}} \psi$

- $p \vDash_{\mathcal{C}} (\phi \vee \psi) \Leftrightarrow p \vDash_{\mathcal{C}} \phi$ *or* $p \vDash_{\mathcal{C}} \psi$

Definition 4. ***Truth*** *A sentence is Intuitionistically valid if it is forced in all structures of all Kripke frames of that the sentence's language.*

In classical logic, this definition simplifies to the one of forcing, and it's simplified again by the fact that p = q; in fact, we can define classical validity as: [1]

Definition 5. ***Truth*** *A sentence is classically valid if it is intuitionistically valid in all single-sentenced Kripke frames of that sentence's language.*

1.3 Translation in classical logic

Here We first define a slightly different version of the destructive tableaux proof tree described by [1], where each node is a signed sentence / truth assertion . This will allow for a more implementation-oriented approach and the translation later on.

The correspondence of the destructive tableaux proof tree described in [1] to our new one is shown in figure [TODO].

Generally speaking, a node in the usual definition is replaced by a sequence of all nodes in the path that goes from the root to it. Afterwards, all nodes in this newly formed tableaux corresponding to nodes that are not leafs of the atomic tableaux that introduced it are removed by adjoining its son(s) and its parent. (TODO)

Generally speaking, a node in the usual definition is replaced by a sequence of all nodes in the path that goes from the root to it. Afterwards, some nodes are removed from the newly formed tableaux by adjoining its son(s) and its parent. A node should be removed if its corresponding node in the original tableaux was not a leaf of the atomic tableaux that introduced it.

Definition 6. ***A Signed sentence*** *is a forcing assertion inside of a tableaux proof. It looks like* $T_q\phi$ *or* $F_p\phi$

Definition 7. ***A Signed sentence list*** *is a list forcing assertions inside of a tableau proof. It looks like* $F_{p_1}\phi_1, T_{q_1}, ...F_{p_1}, ...,$*, it is only a construction (a notation) inside the tableau proof. For now, Intuitivelly we can provide the meaning "There exists a frame for witch* $\mathcal{C}(p_1) \vDash \phi_1$ *and* $\mathcal{C}(p_2) \nvDash \phi_2$ *" to it.*

Definition 8. *A tree with the single node* $\{F\phi\}$ *is a tableaux development of* ϕ*.*

if τ *is a tableaux development of* ϕ *then* $\hookleftarrow(\sigma, \tau)$ *is a tableaux development of* ϕ*. We define the function* $\hookleftarrow$*:*

Given a signed sentence σ *and a tableaux development* τ *of* ϕ *:*

(here we denote $l||l' = l_1, l_2..., l_{|l|}, l'_1, l'_2.., l'_{|l'|}$*)*

- $\hookleftarrow(T\neg\alpha, \tau) = \tau$ *with the node* $t||F\alpha$ *added to all leaves* $h||t$ *that contain* $T\neg\alpha$*.*
- $f(F\neg\alpha, \tau) = \tau$ *with* $L : \{T\alpha\}$ *added (adjoined?) to all leaves* L *that contain* $F\neg\alpha$*.*
- $f(T(\alpha \wedge \beta), \tau) = \tau$ *with* $L : \{T\alpha, T\beta\}$ *added to all leaves* L *that contain* $T(\alpha \wedge \beta)$*.*
- $f(F(\alpha \wedge \beta), \tau) = \tau$ *with* $L : \{F\alpha\}$ *and* $L : \{F\beta\}$ *added to all leaves* L *that contain* $F(\alpha \wedge \beta)$*.*
- $f(T(\alpha \vee \beta), \tau) = \tau$ *with* $L : \{T\alpha\}$ *and* $L : \{T\beta\}$ *added to all leaves* L *that contain* $T(\alpha \vee \beta)$*.*
- $f(F(\alpha \vee \beta), \tau) = \tau$ *with* $L : \{F\alpha, F\beta\}$ *added to all leaves* L *that contain* $F(\alpha \vee \beta)$*.*
- $f(T(\alpha \to \beta), \tau) = \tau$ *with* $L : \{F\alpha\}$ *and* $L : \{T\beta\}$ *added to all leaves* L *that contain* $T(\alpha \to \beta)$*.*

- *$f(F(\alpha \to \beta), \tau) = \tau$ with $L : \{T\alpha, F\beta\}$ added to all leaves L that contain $F(\alpha \to \beta)$.*

Theorem 1. *if the elements of the list $\tau' = \hookleftarrow (\sigma_1, \hookleftarrow (\sigma_2(.....(\hookleftarrow (\sigma_n, F\phi)...))))$ are $\{T_1\phi_{p_1}, T_2\phi_{p_2}, ...F_{p_i}\phi_i, F_{p_{i+1}}\phi_{i+1}...\}$, then the following is true:
if there exist a frame that does not force ϕ then there exists a frame ϕ such that : $\mathcal{C}(p_1) \vDash \phi_1$ and $\mathcal{C}(p_2) \vDash \phi_2$ and ... $\mathcal{C}(p_i) \nvDash \phi_i$ and $\mathcal{C}(p_{i+1}) \nvDash \phi_{i+i}$*

Proof.

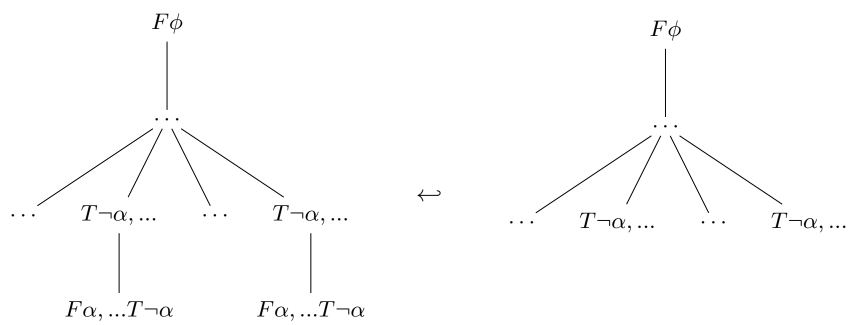

Fig. 2. Example of $\hookleftarrow (T\neg\alpha, \tau)$ and τ

References

1. Nerode, A., Shore, R.A.: Logic for Applications. 2nd edn. Texts in Computer Science. Springer, New York, NY (1997). https://doi.org/10.1007/978-1-4612-0649-1
2. Dummett, M.: Elements of Intuitionism. Oxford Logic Guides. Clarendon Press (2000). https://books.google.fr/books?id=JVFzknbGBVAC
3. Kripke, S.A.: Semantical analysis of intuitionistic logic I. In: Studies in Logic and the Foundations of Mathematics. Elsevier (1965)
4. Author, F.: Article title. Journal **2**(5), 99–110 (2016)
5. Author, F., Author, S.: Title of a proceedings paper. In: Editor, F., Editor, S. (eds.) CONFERENCE 2016, LNCS, vol. 9999, pp. 1–13. Springer, Heidelberg (2016). https://doi.org/10.10007/1234567890
6. Author, F., Author, S., Author, T.: Book title. 2nd edn. Publisher, Location (1999)
7. Author, A.-B.: Contribution title. In: 9th International Proceedings on Proceedings, pp. 1–2. Publisher, Location (2010)
8. LNCS Homepage, http://www.springer.coBm/lncs, last accessed 2023/10/25