Notre Dame Journal of Formal Logic
Volume 46, Number 2, 2005

Finite Tree Property for First-Order Logic with Identity and Functions

Merrie Bergmann

Abstract The typical rules for truth-trees for first-order logic without functions can fail to generate finite branches for formulas that have finite models—the rule set fails to have the *finite tree property*. In 1984 Boolos showed that a new rule set proposed by Burgess does have this property. In this paper we address a similar problem with the typical rule set for first-order logic with identity and functions, proposing a new rule set that does have the finite tree property.

1 Introduction

Here is a typical pair of truth-tree rules for quantified formulas:[1]

Rule 1 Universal quantifications

$$\begin{array}{c}(\forall x)P\\ P(a/x)\end{array}$$

Rule 2 Existential quantifications

$$\begin{array}{c}(\exists x)P\\ P(a/x),\end{array}$$

with a being a constant that is new to the branch

where $P(a/x)$ denotes a substitution instance of the quantified formula, that is, the formula that results from uniformly replacing x in P with a. Owing to its interaction with the universal quantification rule, the rule for existential quantifications may lead exclusively to infinite truth-trees for some formulas that nevertheless have finite

Received May 16, 2003; accepted November 14, 2003; printed May 25, 2005
2000 Mathematics Subject Classification: Primary, 03B10, 03F03
Keywords: finite tree property, truth-trees

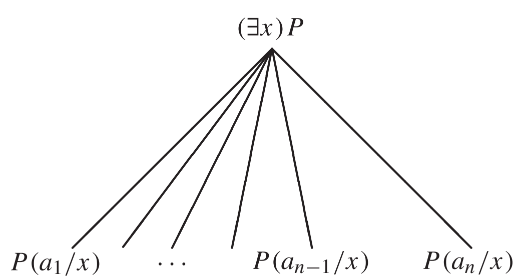
models. Here is the start of a tree for '$(\forall x)(\exists y)Gxy$'.

$$(\forall x)(\exists y)Gxy$$
$$(\exists y)Gay$$
$$Gab$$
$$(\exists y)Gby$$
$$Gbc$$
$$(\exists y)Gcy$$
$$Gcd$$

This tree will never have a finite completed open branch. An open branch is one that contains no pair of contradictory formulas P and $\sim P$, and a completed branch is one on which the relevant rule has been applied to each compound formula and that includes, for each universal quantification $(\forall x)P$ and each constant a occurring on the branch, the substitution instance $P(a/x)$. The problem here is that the new constant introduced by the existential quantification rule must then be used to instantiate the universal quantification, yielding a new existentially quantified formula. Yet not only is the formula '$(\forall x)(\exists y)Gxy$' satisfiable, it is satisfiable in a single-member domain.

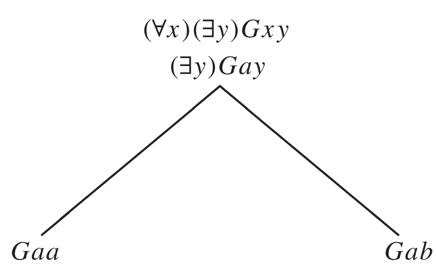
In 1984 Boolos proved that an alternative rule for existential quantifications (proposed by Burgess) guarantees the *finite tree property*, that is, the existence of a finite completed open branch for any finitely satisfiable formula:[2]

Rule 3 Existential quantifications (revised)



where $a_1, \ldots, a_{n-1}$ are the constants that already occur on the branch containing $(\exists x)P$ and a_n is a constant that does not already occur on the branch.

This rule generates the following tree (among others) for the previous formula:



The left branch is complete (although the right branch is not) and open.

It is a nice result that this new rule guarantees the finite tree property while preserving soundness and completeness. But when first-order logic is augmented to include identity and complex terms with function symbols, the obvious tree rules reintroduce the problem. The rule for universal quantifications is generalized to allow instantiation to any closed term, and we add the following rule.

Rule 4 Identity formulas

$$P, t_1 = t_2$$
$$P(t_1 :: t_2)$$

with P being a literal formula.[3]

This rule states that if a branch contains a literal formula P and an identity formula $t_1 = t_2$, then any formula $P(t_1 :: t_2)$, where $P(t_1 :: t_2)$ is the result of replacing one or more occurrences of t_1 in P with t_2 or vice versa, may be added to the branch. Completed open branches are redefined to require that universal quantifications be instantiated to every closed term on their branches and that the identity formula rule be exhaustively applied, and to prohibit formulas of the form $\sim t = t$ (as well as pairs of formulas P and $\sim P$).

Whereas such a system is sound and complete, the finite tree property fails. Consider the tree,

$$(\forall x)Pf(x)$$
$$Pf(a)$$
$$Pf(f(a))$$
$$Pf(f(f(a)))$$
$$\vdots$$

where each substitution instance introduces a new closed term $f(\ldots)$. This branch is bound to be infinite, despite the formula's satisfiability in a single-member domain.

2 A New Rule Set

We can, however, guarantee the finite tree property by making the following four modifications to the existing rule set.

Modification 2.1 *We restrict the rule for universal quantifications so that only constants are used in instantiations (and complete branches require only such instantiations).*

This halts the process exemplified in the previous tree after '$Pf(a)$' has been added. But now consider the following tree for the formula '$(\forall x)Fx \ \& \ \sim Ff(a)$'.

$$(\forall x)Fx \ \& \ \sim Ff(a)$$
$$(\forall x)Fx$$
$$\sim Ff(a)$$
$$Fa$$

The universal quantification has been instantiated with the sole constant occurring on the branch, and the one branch of this tree is open. But the formula is unsatisfiable.

Indeed, '$\sim Ff(a)$' is inconsistent with '$(\forall x)Fx$', but the modified rule for universal quantifications does not allow instantiating '$(\forall x)Fx$' with the complex term '$f(a)$' to produce '$Ff(a)$'.

Modification 2.2 *To compensate for weakening the rule for universal quantifications, therefore, we introduce an additional rule that bears obvious affinities to Burgess's rule:*

Rule 5 Complex terms

$$P_{[f(a_1,\ldots,a_n)]}$$

$$b_1 = f(a_1,\ldots,a_n) \qquad \cdots \qquad b_m = f(a_1,\ldots,a_n) \qquad b_{m+1} = f(a_1,\ldots,a_n)$$

where $P_{[f(a_1,\ldots,a_n)]}$ is a formula containing a closed term $f(a_1,\ldots,a_n)$ whose arguments $a_1,\ldots,a_n$ are constants; $b_1,\ldots,b_m$ are the constants that already occur on the branch on which this formula occurs; and b_{m+1} is a constant that does not already occur on the branch.

This rule ensures that each complex term is asserted to denote the same object as some constant. The above tree is thus extended as follows, with neither branch open:

$$\begin{array}{c} (\forall x)Fx \,\&\, \sim Ff(a) \\ (\forall x)Fx \\ \sim Ff(a) \\ Fa \end{array}$$

$$\begin{array}{ccc} a = f(a) & \qquad\qquad & b = f(a) \\ \sim Fa & & Fb \\ & & \sim Fb \end{array}$$

The last formula on the left is the result of substituting 'a' for '$f(a)$' in '$\sim Ff(a)$', and the last two formulas on the right instantiate '$(\forall x)Fx$' with 'b' and substitute 'b' for '$f(a)$' in '$\sim Ff(a)$'.

The rules' requirement, that the arguments $a_1,\ldots,a_n$ in the term $f(a_1,\ldots,a_n)$ be *constants*, is not unduly restrictive. It just means that sometimes the rule must be applied several times before reaching formulas in which all complex terms have been eliminated.

The rule for identity formulas can also lead to infinite branches for formulas with finite models, as in the following example.

$$a = f(a)$$
$$f(a) = f(f(a))$$
$$f(f(a)) = f(f(f(a)))$$
$$\vdots$$

Here we have repeatedly substituted the term on the right side of a formula in place of the term on the left side in the very same formula.

Modification 2.3 *To prevent such infinite branches, we restrict the substitutions that are permitted by the identity formula rule.*

Rule 6 Identity formulas (revised)

$$P$$
$$a = t$$
$$P(a//t)$$

where P is literal formula, a is a constant, and t is a closed term, and $P(a//t)$ is the result of substituting a for one or more occurrences of t in P.

The restriction prevents the repeated substitutions in the preceding tree. Here is a completed tree for the formula '$a = f(a)$' using the modified identity rule.

$$a = f(a)$$
$$a = a$$

Modification 2.4 *We define a completed branch to be a branch such that*

(a) *each nonliteral formula has had the appropriate rule applied to it,*
(b) *each universal quantification has been instantiated with each constant occurring on that branch, and*
(c) *the complex term rule and identity formula rules have been exhaustively applied.*

The definition of open branches remains the same. The single branch on the preceding tree now counts as a completed open branch, demonstrating the satisfiability of '$a = f(a)$'.

It is easily shown that the new system of tree rules is sound, that is, every tree constructed for a satisfiable formula will have at least one open branch. The system is also complete. To establish a Hintikka completeness proof,[4] we first define a Hintikka set of formulas Γ for our new system as a set with the following properties.

i. There is no formula P such that $\{P, \sim P\} \subseteq \Gamma$.
ii. If $\sim\sim P \in \Gamma$ then $P \in \Gamma$.
iii. If $P \ \& \ Q \in \Gamma$ then $\{P, Q\} \subseteq \Gamma$.
iv. If $\sim(P \ \& \ Q) \in \Gamma$ then $\{\sim P, \sim Q\} \bigcap \Gamma \neq \varnothing$.
v. If $(\forall x)P \in \Gamma$ then for at least one constant a, $P(a/x) \in \Gamma$, and $\{P(b/x) : b$ is a constant that occurs in a formula of $\Gamma\} \subseteq \Gamma$.
vi. If $\sim(\forall x)P \in \Gamma$ then for at least one constant a, $\sim P(a/x) \in \Gamma$.
vii. If $(\exists x)P \in \Gamma$ then for at least one constant a, $P(a/x) \in \Gamma$.
viii. If $\sim(\exists x)P \in \Gamma$ then for at least one constant a, $\sim P(a/x) \in \Gamma$, and $\{\sim P(b/x) : b$ is a constant that occurs in a formula of $\Gamma\} \subseteq \Gamma$.

ix. No formula $\sim t = t$ is a member of Γ.
x. If $a = t \in \Gamma$, where a is a constant, and $P \in \Gamma$, where P is a literal formula, then every formula $P(a//t)$ is also a member of Γ.
xi. If a complex term $f(a_1, \ldots, a_n)$ in which $a_1, \ldots, a_n$ are individual constants occurs in any formula in Γ, then for at least one constant b, $b = f(a_1, \ldots, a_n) \in \Gamma$.

Every completed open branch of a tree in our system is obviously a Hintikka branch.

We now establish that every Hintikka set Γ (in our new sense) has a model $\mathbf{M}$. The model is defined as follows.

Definition 2.5 Let $\mathbf{p}$ be the function that maps the alphabetically $\mathbf{i}$th constant of the language to the integer $\mathbf{i}$. Define a second function $\mathbf{q}$ as follows: $\mathbf{q}(a) = \mathbf{p}(a')$ if a' is the alphabetically earliest constant such that $a' = a \in \Gamma$, and $\mathbf{q}(a) = \mathbf{p}(a)$ otherwise. The domain for $\mathbf{M}$ is the image under $\mathbf{q}$ of the set of individual constants occurring in members of Γ. For each constant a occurring in some sentence in Γ, $\mathbf{M}(a) = \mathbf{q}(a)$. For all other constants a, $\mathbf{M}(a)$ is the smallest positive integer in the domain. For each n-place predicate F (other than the identity predicate), $\mathbf{M}(F) = \{\langle \mathbf{M}(a_1), \ldots, \mathbf{M}(a_n)\rangle : Fa_1, \ldots, a_n \in \Gamma\}$. For each n-place functor f, $\mathbf{M}(f) = \{\langle \mathbf{d}_1, \ldots, \mathbf{d}_n, \mathbf{d}_{n+1}\rangle$: either

(a) there exist constants $a_1, \ldots, a_n, a_{n+1}$ such that $\mathbf{d}_i = \mathbf{M}(a_i)$, $1 \leq i \leq n+1$, and $a_{n+1} = f(a_1, \ldots, a_n) \in \Gamma$, or
(b) there are no such constants and $\mathbf{d}_{n+1}$ is the smallest member of the domain$\}$.

We note that the construction of $\mathbf{M}(f)$ for each functor f assigns to f a function, and not merely a relation, over the domain. This is guaranteed by the way we defined the function $\mathbf{q}$ and by property (x) of Hintikka sets.

Given properties (x) and (xi) of Hintikka sets, it is straightforward to prove that

(∗) If a Hintikka set Γ contains a literal formula P with a closed term $f(t_1, \ldots, t_n)$ then there is some constant a such that $\mathbf{M}(a) = \mathbf{M}(f(t_1, \ldots, t_n))$ and Γ contains each formula $P(a//t)$.

That every member of a Hintikka set Γ is true on the associated model $\mathbf{M}$ can then be established with straightforward induction using the defining properties of Hintikka sets and the following basis. If P is an atomic formula $Ft_1 \ldots t_n$, it follows from (∗) that Γ also contains a formula $Fa_1 \ldots a_n$ such that a_i is a constant and $\mathbf{M}(t_i) = \mathbf{M}(a_i)$, $1 \leq i \leq n$. Since $Fa_1 \ldots a_n \in \Gamma$, $\langle \mathbf{M}(a_1), \ldots, \mathbf{M}(a_n)\rangle \in \mathbf{M}(F)$ by the definition of $\mathbf{M}$, and so both $Fa_1 \ldots a_n$ and $Ft_1 \ldots t_n$ are true on $\mathbf{M}$. If P is an atomic formula $t_1 = t_2$, then it follows from (∗) that Γ also contains a sentence $a_1 = a_2$ such that a_1 and a_2 are constants, $\mathbf{M}(a_1) = \mathbf{q}(a_1) = \mathbf{M}(t_1)$, and $\mathbf{M}(a_2) = \mathbf{q}(a_2) = \mathbf{M}(t_2)$. Let $\mathbf{q}(a_1)$ be $\mathbf{p}(b_1)$ and let $\mathbf{q}(a_2)$ be $\mathbf{p}(b_2)$. It follows by the definition of $\mathbf{q}$ that $b_1 = a_1 \in \Gamma$, with b_1 alphabetically earlier than or identical to a_1, and it thus follows by property (x) that $b_1 = a_2 \in \Gamma$. By the definition of $\mathbf{q}$ also, b_2 is the alphabetically earliest constant such that $b_2 = a_2 \in \Gamma$. Now b_1 cannot be alphabetically earlier than b_2, since $b_1 = a_2 \in \Gamma$. Further, by property (x), $b_2 = b_1 \in \Gamma$ and so $b_2 = a_1 \in \Gamma$; hence b_2 cannot be alphabetically earlier than b_1 since $\mathbf{q}(a_1)$ is $\mathbf{p}(b_1)$. Thus, b_1 and b_2 are the same constant and so $\mathbf{q}(a_1) = \mathbf{q}(a_2)$. Consequently $\mathbf{M}(t_1) = \mathbf{M}(a_1) = \mathbf{M}(a_2) = \mathbf{M}(t_2)$, and $t_1 = t_2$ is true on $\mathbf{M}$.

The new rule system guarantees the existence of at least one tree with a finite completed open branch for any finitely satisfiable formula, with the usual alternation of rules generating such a tree: first apply the rules for truth-functional compounds and existential quantifications, then exhaustively apply the rule for universal quantifications, then the rule for complex terms, then the rule for identity statements, and repeat.

The proof of this guarantee extends Boolos's proof of the same result for first-order logic without functions. Let **M** be a finite model for P with a size **n** domain. A model **N** for P is **M**-*good* if for some $\mathbf{m} \leq \mathbf{n}$, there are **m** constants $a_1, \ldots, a_m$ not occurring in P such that **N** assigns distinct values to $a_1, \ldots, a_m$ and **N** differs from **M** at most in the values assigned to these constants. **M** itself is **M**-good, with $\mathbf{m} = 0$. A *branch* of a tree for P is **M**-good if all of its formulas are true in an **M**-good model for P.

Boolos proved that each rule of the tree subsystem excluding the complex term rule is such that the result of applying the rule to a formula(s) on an **M**-good branch results in at least one extended branch that is **M**-good. But we can show that the complex term rule has this property as well: Assume that the complex term rule is applied on an **M**-good branch whose **M**-good model is **N**. Then the branch contains a formula with a closed complex term $f(a_1, \ldots, a_n)$, where $a_1, \ldots, a_n$ are all constants, such that $b_1 = f(a_1, \ldots, a_n), \ldots, b_m = f(a_1, \ldots, a_n)$, and $b_{m+1} = f(a_1, \ldots, a_n)$ are entered on distinct continuations of that branch. We consider two possibilities.

1. If any one (or more) of $b_1 = f(a_1, \ldots, a_n), \ldots, b_m = f(a_1, \ldots, a_n)$ is true on **N**, then **N** is an **M**-good model for the corresponding continuation of the branch since no new constants have been introduced.
2. If none of $b_1 = f(a_1, \ldots, a_n), \ldots, b_m = f(a_1, \ldots, a_n)$ is true on **N**, then, because b_{m+1} does not already occur on the branch, $b_{m+1} = f(a_1, \ldots, a_n)$ is true on a model $\mathbf{N}'$ that is just like **N** except that $\mathbf{N}'(b_{m+1}) = \mathbf{u}$, where **u** is the member of the domain such that $\langle \mathbf{N}(a_1), \ldots, \mathbf{N}(a_n), \mathbf{u} \rangle$ is a member of $\mathbf{N}(f)$. This member **u** is not assigned to any other constant b_i that occurs on the branch but not in P (else it would follow that $b_i = f(a_1, \ldots, a_n)$ is true on **N**, which contradicts our assumption). Thus $\mathbf{N}'$ is also **M**-good.

Any tree for a formula P with a finite model **M** will always contain at least one **M**-good branch since the initial tree consisting of the formula P is itself an **M**-good branch. And, as Burgess notes, an **M**-good branch contains at most a finite number of formulas. For such a branch can contain no more than $\mathbf{n} + \mathbf{p}$ constants, where **p** is the number of constants occurring in P, and with only finitely many constants at hand the tree rules can generate at most finitely many new formulas. This establishes the finite tree property for first-order logic with identity and functions.

Notes

1. See Bergmann et al. [1] and Smullyan [5] for examples of complete rule sets.

2. The rule and the proof appear in [2].

3. That is, an atomic formula or its negation. The restriction is not necessary but makes the system—as well as its metatheory—simpler.

4. See [1] for such a proof. "Hintikka sets" were first studied by Hintikka in [3] and [4].

References

[1] Bergmann, M., J. Moor, and J. Nelson, *The Logic Book*, McGraw-Hill, New York, 1998. 179

[2] Boolos, G., "Trees and finite satisfiability: Proof of a conjecture of Burgess," *Notre Dame Journal of Formal Logic*, vol. 25 (1984), pp. 193–97. Zbl 0561.03004. MR 85f:03005. 179

[3] Hintikka, J. , "Form and content in quantification theory," *Acta Philosophica Fennica*, vol. 8 (1955), pp. 7–55. Zbl 0067.00103. MR 16,1079b. 179

[4] Hintikka, J., "Notes on the quantification theory," *Societas Scientiarum Fennica. Commentationes Physico-Mathematicae*, vol. 17 (1955), no. 12, 13 pp. Zbl 0067.25002. MR 22:4633. 179

[5] Smullyan, R. M., *First-Order Logic*, Dover Publications, Inc., New York, 1995. Corrected reprint of the 1968 original. Zbl 0172.28901. MR 95j:03001. 179

Department of Computer Science
Smith College
McConnell Hall 214
Northampton MA 01063
bergmann@cs.smith.edu