CS81 Spring 2012 ANALYTIC TABLEAUX

Smullyan [1] introduced an elegant proof procedure called the **analytic tableaux**. He attributed the idea of this method to earlier works by Beth and Hintikka. With this method, to prove a formula α, we assume $\neg\alpha$ and derive a contradiction from this assumption by constructing a *refutation* tree.

The refutation tree contains vertices which are labeled by logical formulas. The root of the tree is labeled with the formula $\neg\alpha$. Given a vertex u of the tree labeled with a formula β, the children of u is constructed based on the form of β by using the rules given below. A leaf v is called **closed** if, for some formula ψ, both ψ and $\neg\psi$ appear along the unique path from the root to v. The refutation tree is **complete** if all leaves are closed.

Rules for Tableaux Construction

1. *Negation* rule:

$$\begin{array}{c} \neg\neg\alpha \\ | \\ \alpha \end{array}$$

2. *Conjunction* rule:

$$\begin{array}{c} \alpha \wedge \beta \\ | \\ \alpha \\ \beta \end{array} \qquad\qquad \begin{array}{c} \neg(\alpha \wedge \beta) \\ \diagup \quad \diagdown \\ \neg\alpha \quad \neg\beta \end{array}$$

3. *Disjunction* rule:

$$\begin{array}{c} \neg(\alpha \vee \beta) \\ | \\ \neg\alpha \\ \neg\beta \end{array} \qquad\qquad \begin{array}{c} \alpha \vee \beta \\ \diagup \quad \diagdown \\ \alpha \quad \beta \end{array}$$

4. *Implication* rule:

$$\begin{array}{c} \neg(\alpha \to \beta) \\ | \\ \alpha \\ \neg\beta \end{array} \qquad\qquad \begin{array}{c} \alpha \to \beta \\ \diagup \quad \diagdown \\ \neg\alpha \quad \beta \end{array}$$

5. *Universal* $\forall$-rule:

$$\begin{array}{c} (\forall x)\alpha \\ | \\ \alpha[a/x] \end{array} \qquad\qquad \begin{array}{c} \neg(\exists x)\alpha \\ | \\ \neg\alpha[a/x] \end{array}$$

   provided a is *any* parameter[1].

6. *Existential* $\exists$-rule:

$$\begin{array}{c} (\exists x)\alpha \\ | \\ \alpha[a/x] \end{array} \qquad\qquad \begin{array}{c} \neg(\forall x)\alpha \\ | \\ \neg\alpha[a/x] \end{array}$$

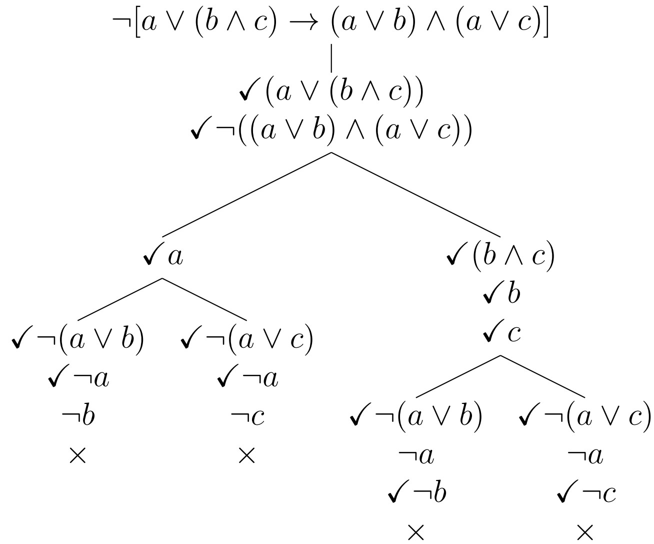
   provided a is a *new* parameter. In [1], Smullyan mentioned that the $\exists$-rule can be *liberalized* by replacing the clause "provided a is a new parameter" by "provided a is new, or else a has not been previously introduced by an $\exists$-rule, and does not occur in $(\exists x)\alpha$, and no parameter of $(\exists x)\alpha$ has been previously introduced by an $\exists$-rule."

[1] In [1], Smullyan used *parameter* to denote what may be called a *constant* (see page 43).

Example 1 Proving the Distributive Law:

$$a \vee (b \wedge c) \vdash (a \vee b) \wedge (a \vee c).$$

The refutation tree is given by:

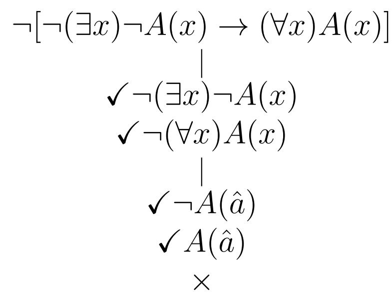


The *checkmarks* denote formulas which are *used* in some step along some branch in the tree.

Example 2 Proving DeMorgan's Law:

$$\neg(\exists x)\neg A(x) \vdash (\forall x)A(x)$$

The refutation tree is given by:



Example 3 (from Smullyan [1]) Proving the sentence $(\exists y)[(\exists x)A(x) \to A(y)]$ using the strict $\exists$-rule.

$$\neg(\exists y)[(\exists x)A(x) \to A(y)]$$
$$|$$
$$\neg[(\exists x)A(x) \to A(a)]$$
$$|$$
$$(\exists x)A(x)$$
$$\neg A(a)$$
$$|$$
$$A(b)$$
$$|$$
$$\neg[(\exists x)A(x) \to A(b)]$$
$$|$$
$$(\exists x)A(x)$$
$$\neg A(b)$$
$$\times$$

Now, using the liberalized $\exists$-rule:

$$\neg(\exists y)[(\exists x)A(x) \to A(y)]$$
$$|$$
$$\neg[(\exists x)A(x) \to A(a)]$$
$$|$$
$$(\exists x)A(x)$$
$$\neg A(a)$$
$$|$$
$$A(a)$$
$$\times$$

References

[1] R. Smullyan, *First-Order Logic*, Dover, 1968.