Tableau Prover Tatzelwurm: Hyper-Links and UR-Resolution

Carsten Bierwald *Thomas Käufl*

Universität Karlsruhe, Institut für Logik, Komplexität und Deduktionssysteme
Postfach 6980, 76128 Karlsruhe, Deutschland
E-Mail: {carsten | kaeufl}@ira.uka.de

Introduction

In this paper, the use of semantic hyper-linking and of UR-resolution in a tableau based theorem prover is discussed. Semantic hyper-linking, see [5] or [6], can be viewed as a systematic search for a model of a formula X. If this search fails, X has been shown to be unsatisfiable.

Also proving the unsatisfiability of a formula using analytic tableaux is a search for a model. So, it seems promising to adapt semantic hyper-linking for the use in tableaux. As hyper-linking generates ground instances of clauses, this approach fits neatly to the tableau prover *Tatzelwurm*. This prover also generates ground instances of γ-formulae. (This allows the use of decision procedures for quantifier theories during a proof. See [8].)

In the recent years implementations of the tableau method used free variables. (See [7] for free variables and [1] or [2] for an implementation of a tableau prover employing this method.) Hyper-linking allows to implement an efficient prover which replaces the variables of γ-formulae by ground terms. So, when the equality appears on a branch, its closure can be determined with a congruence closure algorithm. (See [8].)

Some Terminology

In the tableau calculus a composite formula belongs to one of the following types.

type of the formula	formulae equivalent to
α	$\alpha_1 \wedge \alpha_2, \neg\neg\alpha_1$
β	$\beta_1 \vee \beta_2$
γ	$\forall x\, \gamma(x)$
δ	$\exists x\, \delta(x)$

The variable bound by the outmost quantifier of a γ-formula is said to be *γ-bound*. $\gamma(x)$ ($\delta(x)$) is *contained* in γ (δ), and the α_i (β_i) are contained in α (β). This relation is transitive: If X is contained in Y and Y in Z then X is contained in Z.

A literal is said to be *negative*, if it is a negated atomic formula and *positive* else. A formula is *proper*, if each variable is bound by at most one quantifier. (Rectified formulae are proper.)

A *structure* giving meaning to first order formulae is a pair $A = (U, \mathcal{I})$. U is the universe and $\mathcal{I}$ the interpretation assigning a meaning to the function and predicate symbols. If A is model for a formula X, we write $A \models X$, and $A \not\models X$, if this is not the case. A *Herbrand structure* is a structure where U is the Herbrand universe.

Substitutions are defined as usual. ι is the identity substitution.

Semantic Hyper-Linking and Analytic Tableaux

Assume we are given a structure A and ground literals $L_1, \ldots, L_n$, having A not as model. Then, $A[L_1, \ldots, L_n]$ is a structure with the property

$A[L_1, \ldots, L_n] \models L$ for a ground literal iff

1. $L \in \{L_1, \ldots, L_n\}$ or
2. $L \notin \{L_1, \ldots, L_n\}, \neg L \notin \{L_1, \ldots, L_n\}$ and $A \models L$

The set $\{L_1, \ldots, L_n\}$ is the *explicit part* of $A[L_1, \ldots, L_n]$.

Hyper-Linking selects a clause $C = C_1 \vee C_2$, such that each literal of C appears either in C_1 or C_2. If there is a ground substitution σ, such that each of the literals of $\sigma(C_1)$ has a complement in $\{L_1, \ldots, L_n\}$ and if this is not the case for each of the literals of $\sigma(C_2)$, then $\sigma(C)$ is a *partial hyper-link*. If $\sigma(C_2)$ contains variables, a ground substitution τ must be found, such that $A \not\models \tau(\sigma(C_2))$. (This definition is taken from [6].)

Example. Assume we are given $C = P(x) \vee Q(x)$ and $A[\neg P(a), \neg Q(a), \neg P(b)]$. Using $\sigma(x) = a$, C can be split into $C_1 = P(x) \vee Q(x)$ where C_2 is the empty clause. The substitution $\sigma(x) = b$ allows to split C into $C_1 = P(x)$ and $C_2 = Q(x)$.

We now discuss a small proof performed by CLIN-S and how it can be transformed into a tableau proof. To this end, we assume that there is a function hlu_0 determining which formulae can be used for the tableau expansion. If a formula can be used, then hlu_0 should give the substitutions for obtaining a hyper-link. (In the example we omit that this function depends on the explicit part of the model. (The T_i are sets of ground clauses generared by CLIN-S.)

CLIN-S	Tableau
C1. $\neg P(a)$	1. $\neg P(a)$
C2. $Q(f(z), z)$	2. $\forall z\, Q(f(z), z)$
C3. $\neg Q(f(x), y) \vee P(x)$	3. $\forall x, y\, \neg Q(f(x), y) \vee P(x)$
A_0: all negated atoms are true.	
partial hyper-link: $Q(f(z), z)$	$hlu_0(\forall z\, Q(f(z), z)) = \{\iota\}$.
$T_1 = \{Q(f(a), a)\}$	
$A_1 = A_o[Q(f(a), a)]$	4. $Q(f(a), a)$

partial hyper-link: $\neg Q(f(a), a) \vee P(a)$

$T_2 = \{Q(f(a), a), \neg Q(f(a), a) \vee P(a)\}$

$A_2 = A_o[Q(f(a), a), P(a)]$

partial hyper-link: $\neg P(a)$

T_3

$= \{Q(f(a), a), \neg Q(f(a), a) \vee P(a), \neg P(a)\}$

T_3 is propositional not satisfiable.

$hlu_0(\forall x, y\, \neg Q(f(x), y) \vee P(x))$

$= \{\{x/a, y/a\}\}$

5. $\neg Q(f(a), a) \vee P(a)$

$hlu_0(\neg Q(f(a), a) \vee P(a)) = \{\iota\}$

Now the β-rule is applied to 5. By this, two closed branches are obtained.

Consider the expansion of the tableau with the formula 4. $hlu_0(\forall z\, Q(f(z), z))$ must be $\{\iota\}$, since the partial hyper-link $Q(f(z), z)$ exists. Each ground instance of it must be generated. In the first step, $Q(f(a), a)$ is selected. Since there is no substitution such that $\neg Q(f(x), y) \vee P(x)$ becomes a partial hyper-link, hlu_0 is Ø for formula 3.

We now present the complete definition of *hlu* or more precisely the function $hlu_{S, A}$ which determines the hyper-link unifiers. It depends on a structure A and a set of closed formulae S containing the literals being false in A.

Definition. Let S be a set of closed formulae and F a formula. A variable appearing in S or F is assumed to be bound by at most one quantifier. Let A be a decidable structure with the property: If $L \in S$, then $A \not\models L$. The set $hlu_{S, A}(F)$ of *hyper-link-unifiers* of F is

1. If F is an equation: $hlu_{S, A}(F) = \{\iota\}$

2. F is a literal:
   If $A \not\models F$: $hlu_{S, A}(F) = \{\iota\}$
   If $A \models F$: $hlu_{S, A}(F) = \{\sigma$: $\sigma(F)$ complementary to a $G \in S$, σ most general$\}$

3. $hlu_{S, A}(\alpha) = hlu_{S, A}(\alpha_1) \cup hlu_{S, A}(\alpha_2)$

4. $hlu_{S, A}(\beta) = MG(hlu_{S, A}(\beta_1), hlu_{S, A}(\beta_2))$

5. $hlu_{S, A}(\gamma_x) = gen(x, hlu_{S, A}(\gamma'_x))$

6. $hlu_{S, A}(\delta) = \{\iota\}$

Remark. $\sigma' = gen(x, \sigma)$ is a substitution with $\sigma'(y) = \sigma(y)$ for $x \neq y$ and $\sigma'(x) = x$. (The generalisation of *gen* to sets of substitutions is obvious.) The function *MG* combines to substitutions.

For example $MG(\{\{x/a, y/b\}, \{x/b\}\}, \{\{z/c\}, \iota, \{x/c\}\}) = \{\{x/a, y/b, z/c\}, \{x/a, y/b\}, \{x/b, z/c\}, \{x/b\}\}$ and $MG\{\{x/a\}, \{x/b\}\} = \emptyset$. (The definition can be found in [3] and a forthcoming technical report [4].)

The definition also shows that it is not necessary to transform the formulae into conjunctive normal form. In the implementation the decomposition of composite formulae is done only once. The result is stored as tree together with the formula.

In [3] and [4] it is shown that it is sufficient to determine the hyper-link unifiers in order to obtain a complete proof procedure.

Example. We now demonstrate how the use of hyper-links restricts the application of tableau expansion rules. The proof starts with the formulae 1–3 in the tableau. The structure is a trivial one: Each negated atom is assumed to be true. (Negative hyper-linking)

1. $P(a)$
2. $\forall x, y\ (P(x) \rightarrow Q(f(x), y))$
3. $\forall z\ (Q(f(z), f(z)) \rightarrow \neg Q(f(z), z))$
4. $\forall y\ (P(a) \rightarrow Q(f(a), y))$ $\quad \gamma(2)$
5. $P(a) \rightarrow Q(f(a), a)$ $\quad \gamma(4)$

The hyper-link unifier of 4 is ι. a has been determined by the fair strategy.

<table>
<tr><td rowspan="2">6. $\neg P(a)$</td><td colspan="2">7. $Q(f(a), a)$ $\quad \beta(5)$
8. $P(a) \rightarrow Q(f(a), f(a))$ $\quad \gamma(4)$
2 and 3 cannot be used. $f(a)$ has been determined by the fair strategy.</td></tr>
<tr><td>9. $\neg P(a)$</td><td>10. $Q(f(a), f(a))$ $\quad \beta(5)$</td></tr>
</table>

The branches ending with 6 and 9 are closed. In the branch ending with 10 the set of hyper-link unifiers of formula 3 is $gen(z, \{\{z/a\}\})$. The proof is successfully terminated after expansion of the tableau with $Q(f(a), f(a)) \rightarrow \neg Q(f(a), a))$.

1. How does hyper-linking constrain the application of the γ-rule? If formula 7 has been added to the tableau, the sets of hyper-link unifiers for the quantifications 2 and 3 are still empty. For formula 4 the only hyper-link unifier is ι. A fair strategy, for example that suggested by Smullyan [9] would have selected 2 or 3 for the expansion of the tableau.

2. The sets of hyper-link unifiers generated in the example are $\{\iota\}$ where ι is the identity. In these cases, a fair strategy must select an instantiation from the Herbrand universe. Additionally, heuristics can be used to determine further instantiations. *Tatzelwurm* maintains a list of links between all pairs of literals which can be made complementary. So, there is a link between 1 and $\neg P(x)$ being contained in 5. Assume the fair strategy would select the term $t \neq a$ when applying the γ-rule to 4. Since this link exists, formula 5 would be added to the tableau together with $P(t) \rightarrow Q(f(t), t)$.

3. A β-formula must be used for tableau expansion, iff the set of its hyper-link unifiers is $\{\iota\}$. ι is the only possible hyper-link unifier, since these formulae are closed when appearing on a tableau branch.

Unit Resulting Resolution

In CLIN-S (see [5]) unit resulting resolution has been used for a quick test in order to reject possible models. Since the results obtained were promising, we tried to adopt this method in *Tatzelwurm*.

Units are ground literals or universally closed literals. Unit resulting resolution (UR-resolution) is used by the tableau procedure as lemma generation. Suppose that $\forall y\ (\neg P(y) \lor R(y))$ and $\forall x\ P(x)$ appear on a branch. UR-resolution generates $\forall y\ R(y)$. This unit is not added to the tableau formulae. It is kept in a separate data structure. Only if the branch is tested for closure, *Tatzelwurm* checks whether a unit can be made complementary to a literal appearing on the branch. UR-resolution is applied when a clause-like γ-formula is selected for tableau expansion. ($P \rightarrow Q \land R$ is not clause-like, $P \rightarrow Q \lor R$ and $P \land Q \rightarrow R$ are.) During generation of new units it is checked whether a pair of complementary units is obtained or if the existing units allow to derive the empty clause from the selected γ-formula. In these cases the branch is closed.

By this, UR-resolution helps to find out closures of branches in early steps of the proof.

Conclusion and Future Work

Adding semantic hyper-linking and UR-resolution to *Tatzelwurm* has increased its deductive power considerably. Both methods can be used when the prover uses decision procedures for theories too. So, when the equality occurs in a proof, the system uses a congruence closure algorithm.

Tatzelwurm offers an interface allowing the user to define structures. At present, the only "built in" structures are trivial ones. (All not negated literals are true, false resp.) It is planned to study also other structures for use as built-in ones.

UR-resolution does not apply the built in decision procedures for theories. It seems to be promising to investigate the combination of these procedures with UR-resolution.

References

1. Beckert, B., Possega, J. leanTAP. Lean Tableau Based Theorem Proving. In: Proc. 13th Int. Conf. on Automated Deduction (CADE 13). Springer 1996

2. Beckert, B., Hähnle, R., Oel P., Sulzmann R. The Tableau-Based Theorem Prover 3TAP, version 4.0. In: Proc. 12th Int. Conf. on Automated Deduction (CADE 12). Springer 1994

3. Bierwald, C.: Formel-Links für einen Tableaubeweiser. Diploma Thesis. Institut für Logik, Komplexität und Deduktionssysteme. Universität Karlsruhe, 1996

4. Bierwald, C., Käufl, Th.: Adding Semantic Hyper-Links and UR-Resolution to the Tableau Prover Tatzelwurm. Technical Report 30. Institut für Logik, Komplexität und Deduktionssysteme. Universität Karlsruhe, in prepararation.

5. Chu, H.: Semantically Guided First-Order Theorem Proving Using Hyper-Linking. PhD thesis. University of North Carolina at Chapel Hill. 1994

6. Chu, H., Plaisted, D. A.: Semantically Guided First-Order Theorem Proving Using Hyper-Linking. In: Proc. 12th Int. Conf. on Automated Deduction (CADE 12). Springer 1994

7. Fitting M.: First-Order Logic and Automated Theorem Proving. Springer, 1990

8. Käufl, Th.; Zabel, N.: Cooperation of Decision Procedures in a Tableau-Based Theorem Prover. Revue d'Intelligence Artificielle, Vol. 4, no. 3: 1990, pp. 99 - 126

9. Smullyan, R. First-Order Logic, Springer 1968