Refutation of resolution-based decision procedure for two variables with equality

Abstract: We evaluate six equations and three conjectures for the decision procedure. None is tautologous. This refutes the procedure for two variables with equality, and forms a *non* tautologous fragment of the universal logic VŁ4.

We assume the method and apparatus of Meth8/VŁ4 with `T`autology as the designated proof value, **`F`** as contradiction, `N` as truthity (non-contingency), and `C` as falsity (contingency). The 16-valued truth table is row-major and horizontal, or repeating fragments of 128-tables, sometimes with table counts, for more variables. (See ersatz-systems.com.)

LET ~ Not, ¬ ; + Or, ∨, ∪, ⊔ ; - Not Or; & And, ∧, ∩, ⊓ , · , ⊗ ; \ Not And;
> Imply, greater than, →, ⇒ , ↦, ≻, ⊃, ↠ ; < Not Imply, less than, ∈, ≺, ⊂, ⊬, ⊭, ↞, ≲ ;
= Equivalent, ≡, :=, ⇔, ↔, ≜, ≈, ≃ ; @ Not Equivalent, ≠, ⊕;
% possibility, for one or some, ∃, ◊, M; # necessity, for every or all, ∀, □, L;
(z=z) `T` as tautology, ⊤, ordinal 3; (z@z) **`F`** as contradiction, Ø, Null, ⊥ , zero;
(%z>#z) `N` as non-contingency, Δ, ordinal 1; (%z<#z) `C` as contingency, ∇, ordinal 2;
~(y < x) (x ≤ y), (x ⊆ y), (x ⊑ y); (A=B) (A~B).
Note for clarity, we usually distribute quantifiers onto each designated variable.

From: de Nivelle, H.; Pratt-Hartmann, I. (2001). A resolution-based decision procedure for the two variable fragment with equality. nivelle@mpi.mpg-sb.de, ipratt@cs.man.ac.uk www.cs.man.ac.uk/~ipratt/papers/logic/ijcar01.pdf

1 Introduction

The two-variable-fragment L2 ≈ is the set of formulas that do not contain function symbols, that possibly contain equality (≈), and that use only two variables. The two-variable fragment without equality L2 is the subset of L2 ≈ not involving the predicate ≈. For example, the formula

∀x∃y[r(x, y) ∧ ∀x(r(y, x) → x≈y)], (1.1.1)

LET p, q, r, s: a, x, r, y.

(r&(#q&%s))&((r&(%s&#q))>(#q=%s)) ;
`FFFF FFFF FFFF FF``NN` (1.1.2)

stating that every element is r-related to some element whose only r-successor is itself, is in L2 ≈ (but not in L2). Note in particular the 're-use' of the variable x by nested quantifiers in this example. In the same way, it is possible to translate modal formulas into L2, (without equality) by reusing variables. For example, the modal formula

□◊□a (1.2.1)

#%#p = (s=s) ; **`F`**`N`**`F`**`N` **`F`**`N`**`F`**`N` **`F`**`N`**`F`**`N` **`F`**`N`**`F`**`N` (1.2.2)

can be translated into

∀y(r(x, y) → ∃x(r(y, x) ∧ ∀y(r(x, y) → a(y)))). (1.3.1) ((r&(#q&%s))&(r&(%s&#q)))>(p&#s) ;

```
TTTT TTTT TTTT TTCT
```
(1.3.2)

No equality is needed for translating modal formulas.

Remark 1.3.2: Eqs. 1.2.2 is not equivalent to 1.3.2 as claimed.

2 Motivation

A logic is said to have the *finite-model property* if any satisfiable formula in that logic is satisfiable in a finite structure. It is easy to see that any fragment of first order logic having the finite model property is decidable; and indeed, most of the known decidable fragments of first-order logic have the finite model property. … One such fragment of particular interest here is the so-called Gödel class: the set of first-order formulas *without equality* which, when put in prenex form, have quantifier prefixes matching the pattern $\exists * \forall\forall \exists *$. Gödel .. showed that the Gödel class has the finite model property, and is thus decidable. In the same paper, Gödel claimed that allowing ≈ in formulas of the Gödel class would not affect the finite model property, a claim which was later shown to be false by Goldfarb.. . Between these two discoveries, Scott .. showed that any formula of the two-variable fragment can be transformed into a formula in the Gödel fragment which is equisatisfiable. Relying on Gödel's incorrect claim, Scott concluded decidability for *L*2 ≈*:* Of course, what Scott actually showed was the decidability for *L*2 only. That the full two-variable fragment does indeed have the finite model property was eventually established by Mortimer .. . The fragment *L*2 ≈ is of particular interest when dealing with natural language input, because many simple natural language sentences translate into *L*2 ≈. To give a somewhat fanciful example, the sentence

Every meta-barber shaves every man who shaves no man who shaves himself.

translates to the two-variable formula

$$\forall x(\text{meta-barber}(x)\rightarrow\forall y((\text{man}(y)\wedge\forall x((\text{man}(x)\wedge\text{shave}(x,x))\rightarrow\text{shave}(y,x)))\rightarrow\text{shave}(x,y))). \quad (2.1.1)$$

LET p, q, s, x, y: meta-barber, man, shave, x, y.

(p&#x)>((((q&#y)&(s&(#x&#x)))>(~s&(%y&#x)))>(s&(#x&%y))) ;

```
TTTT TTTT TTTT TTTT(16)
TCTC TCTC TCTC TCTC(16)
TTTT TTTT TTTT TTTT(16)
TCTC TCTC TTTT TTTT(16)
```
(2.1.2)

Remark 2.1.2: Eq. 2.1.2 as stated is *not* tautologous, meaning the example is not a theorem as presumably it should be.

3 Making equality disappear

In this section, we give give a method for removing equality from a formula in L2 ≈, based on resolution. … Occurrences of the ≈-symbol fall into two groups. Negative occurrences can be 'simulated' without recourse to equality. Positive occurrences can be restricted to those belonging to a ∃! quantifier.

Remark 3: We interpret the quantifier ∃! as ∃, due to the *non* tautologous performances of Eqs. 1… and 2… above.

Lemma 5. Let $\gamma(x)$ be a formula not involving the variable y and let $\delta(y)$ a formula not involving the variable x. Then the formulas

$$\forall x \forall y(\gamma(x) \vee \delta(y) \vee x \approx y) \quad (5.1.1)$$

LET p, q, r, s: γ, δ, x, y.

((p&#r)+(q&#s))+(#r=#s) ; `TTTT CTCT CCTT TTTT` (5.1.2)

and

$$\forall x\ \gamma(x) \vee \forall x\ \delta(x) \vee (\exists! x\ \neg\ \gamma(x) \wedge \forall x(\gamma(x) \leftrightarrow \delta(x))) \quad (5.2.1)$$

((p&#r)+(q&#r))+((%r&~(p&r))&((p&#r)=(q&#r))) ;
`CCCC TNTN CCCC TNTN` (5.2.2)

are logically equivalent. Using this, we can use the splitting rule to decompose the disjunctions of the Type 3 formula. The result is a formula, in all positive occurrences of ≈ belong to a ∃! quantifier. These can be eliminated by introducing new individual constants.

Remark 5.2.2: Eqs. 5.1.2 and 5.2.2 not logically equivalent, thereby refuting Lemma 5.