

# Implementation of decision procedures for non-Fregean logic SCI

Adrian Siwiec

May 1, 2024

Test: ażółółóń

## 1 Sentential logic and the Fregean axiom

## 2 SCI

### 2.1 ...

### 2.2 Axiomatization of SCI

## 3 Satisfiability in SCI

### 3.1 SCI is decidable

### 3.2 Decision procedures

#### 3.2.1 $T$ decision procedure

#### 3.2.2 $T^*$ decision procedure

TODO: differences compared to “Dedukcyjne Dylematy” (DD):

- Rule  $sym$  is replaced by  $sym_1$  and  $sym_2$ .
- Make a clear distinction of two phases: decomposition-reduction phase and closure phase. The decomposition-reduction phase only uses decomposition, reduction and equality rules, the closure phase only the closure rules and the reduction rules.
- Make it clear that when applying rule to  $\Phi \cup \{\varphi\}$  the  $\varphi$  is removed from the conclusion, unless otherwise noted (i.e. in closure rules).
- No need to say: “w wyniku zastosowania reguły co najmniej jedna z konkluzji zawiera formułę, która nie występuje w  $X \cup \{\varphi\}$ ”. In decomposition-reduction phase simply apply all the rules as long as you can. In closure phase apply closure rule only once to any given formula.

#### 3.2.3 Correctness and completeness of $T^*$

**Theorem 1.**  $T^*$  is correct.

*Proof.* We want to show that if  $T^*$  says the formula is closed, it is indeed closed. TODO:

1. All the rules of  $T^*$  are strongly correct. TODO at least the non-obvious ones.
2. We're given a closed tree.
3. Going up the tree, on every step we have a closed set of formulas, because every rule is strongly correct.
4. The root is closed.

□

**Theorem 2.**  $T^*$  is complete.

We want to show, that if the formula is closed, the tree produced by  $T^*$  is closed, or by contraposition, that if the tree produced by  $T^*$  is open, the given formula is open.

First, let us show that first the decomposition-reduction phase finishes and then that the closure phase finishes.

**Theorem 3.** *Decomposition-reduction phase of  $T^*$  finishes and produces finite trees.*

*Proof.* First, to show that the decomposition-reduction phase finishes, let us define:

**Definition 3.1** (Size of formula). Given a formula  $\varphi$ , let us define:

$$s(\varphi) = \begin{cases} 1, & \text{if } \varphi = p, \\ c(\psi) + 1, & \text{if } \varphi = \neg\psi, \\ c(\psi) + c(\vartheta) + 1, & \text{if } \varphi = \psi \equiv \vartheta, \text{ or } \varphi = \psi \rightarrow \vartheta. \end{cases}$$

**Definition 3.2** (Complexity of a set of formulas). Given a set of formulas  $\Phi$ , let us define:

$$c(\Phi) = \sum_{\phi \in \Phi} 3^{s(\phi)}$$

**Lemma 4.** *If rule  $R$  is a decomposition rule or an identity rule,  $\Phi$  is its premise and  $\Phi_1, \dots, \Phi_n$  are its conclusions, then  $c(\Phi) > c(\Phi_i)$ , for  $1 \leq i \leq n$ .*

*Proof.* TODO: Przerachować. Chyba zdaje się działać, ewentualnie trochę podkręcić  $s$  lub  $c$ .  $\square$

**Lemma 5.** *If rule  $R$  is a reduction rule  $\Phi$  is its premise and  $\Phi_1$  is its conclusion, then  $c(\Phi) \geq c(\Phi_1)$ .*

*Proof.* Trivial.  $\square$

**Lemma 6.** *The reduction rules cannot be applied indefinitely without lowering the  $c$  function.*

*Proof.* TODO: proste.  $\square$

As a consequence of lemmas 4, 5 and 6 it is clear that, for each node of the produced tree, after a finite number of rules applied, either a closed node or a node with lower complexity will be reached. Since the complexity of the node is always positive and finite the produced tree will be finite.  $\square$

**Theorem 7.** *All leaves produced by the decomposition-reduction phase are either closed or contain only simple formulas.*

*Proof.* Simple. Assume there is a non-simple formula in an open leaf. One of decomposition, reduction or equality rules would apply (TODO, simple, because some will match and we don't have any additional conditions on applying rules). Contradiction.  $\square$

**Theorem 8.** *Open leaf of a  $T^*$  tree satisfies the following conditions: TODO: rewrite from DD.*

1. X1. Shown by theorem above.
2. X2
3. X3
4. X4
5. X5
6. X6
7. X7
8. X8

## 4 Implementation

### 4.1 Data structures

### 4.2 Optimizations

## 5 Comparison of $T$ and $T^*$ systems

### 5.1 Size of produced proof trees

### 5.2 Time complexity

### 5.3 Real time