A *DPLL* PROCEDURE FOR THE PROPOSITIONAL GÖDEL LOGIC*

Dušan Guller

Department of Applied Informatics, Comenius University, Mlynská dolina, 842 15 Bratislava, Slovakia

Keywords: The DPLL procedure, Automated deduction, Gödel logic, Many-valued logics.

Abstract: In the paper, we investigate the satisfiability and validity problems of a formula in the propositional Gödel logic. Our approach is based on the translation of a formula to an equivalent *CNF* one which contains literals of the augmented form: either a or $a \rightarrow b$ or $(a \rightarrow b) \rightarrow b$, where a, b are propositional atoms or the propositional constants *0, 1*. A *CNF* formula is further translated to an equisatisfiable finite order clausal theory which consists of order clauses, finite sets of order literals of the forms $a \eqcirc b$ or $a \prec b$. $\eqcirc$ and $\prec$ are interpreted by the equality and strict linear order on $[0,1]$, respectively. A variant of the *DPLL* procedure for deciding the satisfiability of a finite order clausal theory is proposed. The *DPLL* procedure is proved to be refutation sound and complete. Finally, we reduce the validity problem of a formula (tautology checking) to the unsatisfiability of a finite order clausal theory.

SCIENCE AND TECHNOLOGY PUBLICATIONS

1 INTRODUCTION

A noticeable effort has been made in the development of *SAT* solvers (called *SAT* solvers for the Boolean satisfiability problem), especially in the last decade. Roughly speaking, *SAT* solvers exploit either complete solution methods, called complete or systematic *SAT* solvers, or incomplete or hybrid ones. Complete *SAT* solvers are mostly based on the Davis-Putnam-Logemann-Loveland procedure (*DPLL*) (Davis, 1960; Davis, 1962) improved by various features. Some of the latest overviews of the development of *SAT* solvers, with the underlying complexity theory, may be found in (Dixon, 2004; Dixon, 2004; Kautz, 2007; Gomes, 2007; Biere, 2009). The research in many-valued logics mainly concerns finitely-valued ones. Thank to the finiteness of truth value sets of these logics, almost straightforward extensions of results achieved in classical logic are feasible. The *DPLL* procedure has been firstly generalised for regular clauses over a linearly ordered truth value set (Hähnle, 1996). In ((anyà, 1998), it is described an implementation of this regular *DPLL* procedure with the extended two-sided Jeroslow-Wang literal selection rule defined in (Hähnle, 1996). A signed *DPLL* procedure over a finite truth value set is introduced in (Beckert, 2000). It is based on a branching rule forming branches for every truth value. So, the branching factor equals the cardinality of the truth value set. The branching factor can be decreased by a quotient of the truth value set wrt. a suitable equivalence. A slight modification of that equivalence enables a generalisation to an infinite truth value set as well (Guller, 2009). Another signed variant of the *DPLL* procedure for a countable clausal theory over an arbitrary truth value set is proposed in (Guller, 2009). In some sense, the *DPLL* procedure may be viewed like ”anti-resolution”. Thus, its branching rule, with a finite branching factor, may be considered as if a ”signed anti-hyperresolution rule”. The procedure is refutation complete if the finitary disjunction condition for the set of signs occurring in the input countable clausal theory is satisfied. Infinitely-valued logics have not yet been explored so widely as finitely-valued ones. It is not known any general approach as signed logic one in the finitely-valued case. The solution of the *SAT* and *VAL* problems strongly varies on a chosen infinitely-valued logic. The same holds for the translation of a formula to clause form, the existence of which is not guaranteed in general. The results in this area have been achieved in several ways, since infinite truth value sets form distinct algebraic structures. One approach may be based on the reduction from the infinitely-valued case to the finitely-valued one, as it has been done e.g. for the *VAL* problem in the propositional infinitely-valued Łukasiewicz logic in (Mundici, 1987; Aguz-

*Partially supported by the grants VEGA 1/0688/10 and VEGA 1/0726/09.

Guller D..
A DPLL PROCEDURE FOR THE PROPOSITIONAL GÖDEL LOGIC.
DOI: 10.5220/0003061700310042
In *Proceedings of the International Conference on Fuzzy Computation and 2nd International Conference on Neural Computation* (ICFC-2010), pages 31-42
ISBN: 978-989-8425-32-4
Copyright © 2010 SCITEPRESS (Science and Technology Publications, Lda.)

zoli, 2000). Another approach exploits the reduction of the *SAT* problem to mixed integer programming (*MIP*) (Hähnle, 1994a; Hähnle, 1997). (Baaz, 2001) investigates the *VAL* problem in the prenex fragment of the first-order Gödel logic enriched by the relativisation operator Δ, denoted as the prenex G_∞^Δ. At first, a variant of Herbrand's Theorem for the prenex G_∞^Δ is proved, which reduces the *VAL* problem of a formula in the prenex G_∞^Δ to the *VAL* problem of an open formula in G_∞^Δ. Then a chain normal form is defined using the formulae $\phi \lessdot \psi$, as an abbreviation for $\neg\Delta(\psi \rightarrow \phi)$, and $\phi \equiv_\Delta \psi$, as an abbreviation for $\Delta(\phi \rightarrow \psi) \wedge \Delta(\psi \rightarrow \phi)$. These formulae express the strict dense linear order with endpoints and equality on $[0,1]$, which is not possible without Δ in G_∞. Further, a meta-level logic of order clauses is defined, which is a fragment of classical one. An order clause is a finite set of inequalities of the form either $A < B$ or $A \leq B$ where $<$, $\leq$ are meta-level predicate symbols and A, B are atoms of G_∞^Δ considered as meta-level terms. The semantics of the meta-level logic of order clauses is given by classical interpretations on $[0,1]$, varying on assigned (truth) values to atoms of G_∞^Δ handled as meta-level terms, which are the strict dense linear order with endpoints on $[0,1]$; $<$ is interpreted as the strict dense linear order with endpoints and $\leq$ as its reflexive closure on $[0,1]$. A formula in the prenex G_∞^Δ is valid if and only if a translation of it to the order clause form is unsatisfiable with respect to the semantics of the meta-level logic. The chaining calculi in (Bachmair, 1994; Bachmair, 1998) may be used for efficient deduction over order clauses.

In the paper, we investigate *SAT* and *VAL* problems of a formula in the propositional Gödel logic. Our approach is based on the translation of a formula to an equivalent *CNF* one, Lemma 3.1, Section 3, which contains literals of the augmented form: either a or $a \rightarrow b$ or $(a \rightarrow b) \rightarrow b$, where a, b are propositional atoms or the propositional constants *0*, *1*. At this stage, unlike the chain normal form in (Baaz, 2001), we do not need to express the linear order of truth values by any formulae. We consider a ground fragment of the first-order two-valued logic with equality and strict order. The syntax is given by a class of order clausal theories. An order clause is a finite set of order literals of the form either $a \eqcirc b$ or $a \prec b$. The semantics is given by a class of order interpretations. An order interpretation is a first-order two-valued interpretation such that its universum is $[0,1]$, $\eqcirc$ is interpreted as $=_{[0,1]}$, and $\prec$ as $<_{[0,1]}$. For the purpose of solving the *SAT* problem, a *CNF* formula is translated to an equisatisfiable finite order clausal theory, Lemma 3.3, Section 3. The basis is the translation of a literal to an order clause: e.g. $a \rightarrow b$ is translated to $a \prec b \vee a \eqcirc b \vee b \eqcirc 1$ or $(a \rightarrow b) \rightarrow b$ to $b \prec a \vee b \eqcirc 1$. The trichotomy on order literals: either $a \prec b$ or $a \eqcirc b$ or $b \prec a$, naturally invokes proposing a variant of the *DPLL* procedure with a trichotomy branching rule as an algorithm for deciding the satisfiability of a finite order clausal theory. The *DPLL* procedure is proved to be refutation sound and complete, Theorem 4.1, Section 4. The set of basic Rules (37), (38), (39) may be augmented by the admissible ones (50), (51), (52), (53), (54), (55), which are suitable for practical computing and considerably shorten *DPLL* trees. In case of solving the *VAL* problem, we exploit the fact that a formula ϕ is a tautology (valid) if and only if the order formula $\phi \prec 1$ is unsatisfiable, Theorem 5.1, Section 5. At first, ϕ is translated to an equivalent *CNF* formula $\psi = \bigwedge_{i \leq n} \bigvee_{j \leq m_i} l_j^i$, l_j^i are literals. Hence, ϕ is a tautology if and only if the order formula $\psi \prec 1 \equiv \bigvee_{i \leq n} \bigwedge_{j \leq m_i} l_j^i \prec 1$ is unsatisfiable. Further, every order formula $l_j^i \prec 1$ is translated to an equisatisfiable conjunction of disjunctions of order literals: e.g. $(a \rightarrow b) \prec 1$ is translated to $b \prec a \wedge b \prec 1$ or $((a \rightarrow b) \rightarrow b) \prec 1$ to $(a \prec b \vee a \eqcirc b) \wedge b \prec 1$. This yields an equisatisfiable finite order clausal theory T_ϕ to $\psi \prec 1$ and $\phi \prec 1$. So, ϕ is a tautology if and only if T_ϕ is unsatisfiable.

The paper is organised as follows. Section 2 gives the basic notions, notation, and useful properties concerning the propositional Gödel logic. Section 3 deals with clause form translation. In Section 4, we propose a variant of the *DPLL* procedure with a trichotomy branching rule and prove its refutational soundness, completeness. Section 5 solves the *VAL* problem (tautology checking).

2 PROPOSITIONAL GÖDEL LOGIC

Throughout the paper, we shall use the common notions of propositional many-valued logics. The set of propositional atoms of Gödel logic will be denoted as *PropAtom*. By *PropForm* we designate the set of all propositional formulae of Gödel logic built up from *PropAtom* using the propositional constants *0*, the false, *1*, the true, and the connectives $\neg$, negation, $\wedge$, conjunction, $\vee$, disjunction, $\rightarrow$, implication. We shall assume that Gödel logic is interpreted by the standard G-algebra

$$G = ([0,1], \leq, \vee, \wedge, \Rightarrow_G, {}^{-G}, 0, 1)$$

where $\vee$ and $\wedge$ denote the respective supremum and infimum operators on $[0,1]$,

$$a \Rightarrow_G b = \begin{cases} 1\, if\, a \leq b, \\ b\, else, \end{cases}$$

$$\overline{a}^G = \begin{cases} 1\, if\, a = 0, \\ 0\, else. \end{cases}$$

We recall that G is a complete linearly ordered lattice algebra; the supremum operator $\vee$ is commutative, associative, idempotent, monotone, 0 is its neutral element; the infimum operator $\wedge$ is commutative, associative, idempotent, monotone, 1 is its neutral element;[2] the residuum operator $\Rightarrow_G$ of $\wedge$ satisfies the condition of residuation:

$$\text{for all } a,b,c \in G,\ a \wedge b \leq c \Longleftrightarrow a \leq b \Rightarrow_G c; \quad (1)$$

the Gödel negation $\overline{}^G$ satisfies the condition:

$$\text{for all } a \in G,\ \overline{a}^G = a \Rightarrow_G 0; \quad (2)$$

and the following properties, which will be exploited later, hold:[3]

For all $a,b,c \in G$,

$$a \vee b \wedge c = (a \vee b) \wedge (a \vee c), \quad (3) \text{ (distributivity of } \vee \text{ over } \wedge)$$

$$a \wedge (b \vee c) = a \wedge b \vee a \wedge c, \quad (4) \text{ (distributivity of } \wedge \text{ over } \vee)$$

$$a \Rightarrow_G (b \vee c) = a \Rightarrow_G b \vee a \Rightarrow_G b, \quad (5)$$

$$a \Rightarrow_G b \wedge c = (a \Rightarrow_G b) \wedge (a \Rightarrow_G b), \quad (6)$$

$$(a \vee b) \Rightarrow_G c = (a \Rightarrow_G c) \wedge (b \Rightarrow_G c), \quad (7)$$

$$a \wedge b \Rightarrow_G c = a \Rightarrow_G c \vee b \Rightarrow_G c, \quad (8)$$

$$a \Rightarrow_G (b \Rightarrow_G c) = a \wedge b \Rightarrow_G c, \quad (9)$$

$$((a \Rightarrow_G b) \Rightarrow_G b) \Rightarrow_G b = a \Rightarrow_G b, \quad (10)$$

$$(a \Rightarrow_G b) \Rightarrow_G c = ((a \Rightarrow_G b) \Rightarrow_G b) \wedge (b \Rightarrow_G c) \vee c, \quad (11)$$

$$(a \Rightarrow_G b) \Rightarrow_G 0 = ((a \Rightarrow_G 0) \Rightarrow_G 0) \wedge b \Rightarrow_G 0. \quad (12)$$

A valuation $\mathcal{V}$ of propositional atoms is a mapping $\mathcal{V} : PropAtom \longrightarrow [0,1]$. A partial valuation $\mathcal{V}$ of propositional atoms with the domain $dom(\mathcal{V}) \subseteq PropAtom$ is a mapping $\mathcal{V} : dom(\mathcal{V}) \longrightarrow [0,1]$. Let $atoms(\phi), atoms(T) \subseteq dom(\mathcal{V})$ in case of $\mathcal{V}$ being a partial valuation. The truth value ϕ in $\mathcal{V}$, in symbols $\|\phi\|^{\mathcal{V}}$, is defined by the standard way; the propositional constants *0*, *1* are interpreted by 0, 1, respectively, and the connectives by the respective operators on G. $\mathcal{V}$ is a (partial) propositional model of ϕ, in symbols $\mathcal{V} \models \phi$, iff $\|\phi\|^{\mathcal{V}} = 1$. $\mathcal{V}$ is a (partial) propositional model of T, in symbols $\mathcal{V} \models T$, iff for all $\phi \in T$, $\mathcal{V} \models \phi$. ϕ is a propositional consequence of T, in symbols $T \models_P \phi$, iff for every propositional model $\mathcal{V}$ of T, $\mathcal{V} \models \phi$. ϕ is equivalent to ϕ', in symbols $\phi \equiv \phi'$, iff for every valuation $\mathcal{V}$, $\|\phi\|^{\mathcal{V}} = \|\phi'\|^{\mathcal{V}}$. $\phi \mid T$ is satisfiable iff there exists a propositional model of $\phi \mid T$. $\phi \mid T$ is equisatisfiable to $\phi' \mid T'$ iff $\phi \mid T$ is satisfiable if and only if $\phi' \mid T'$ is satisfiable.

Let X, Y, Z be sets, $Z \subseteq X$, and $f : X \longrightarrow Y$ a mapping. By $X \subseteq_{\mathcal{F}} Y$ we denote X is a finite subset of Y. We designate $\mathcal{P}(X) = \{x \,|\, x \subseteq X\}$, $\mathcal{P}(X)$ is the power set of X; $\mathcal{P}_{\mathcal{F}}(X) = \{x \,|\, x \subseteq_{\mathcal{F}} X\}$, $\mathcal{P}_{\mathcal{F}}(X)$ is the set of all finite subsets of X; $f[Z] = \{f(z) \,|\, z \in Z\}$, $f[Z]$ is called the image of Z with respect to f; and $f|_Z = \{(z, f(z)) \,|\, z \in Z\}$, $f|_Z$ is the restriction of f onto Z. $f : \omega \longrightarrow Y$ is a sequence of Y iff f is a bijection.

3 TRANSLATION TO CLAUSAL FORM

We propose translation of a formula to an equivalent *CNF* formula, Lemma 3.1. In contrast to two-valued logic, we have to consider an augmented set of literals appearing in *CNF* formulae. Let $l, \phi \in PropForm$. l is a literal iff either $l = a$ or $l = a \rightarrow b$ or $l = (a \rightarrow b) \rightarrow b$ where $a \in PropAtom$ and $b \in PropAtom \cup \{0\}$. ϕ is a conjunctive | disjunctive normal form, in symbols *CNF* | *DNF*, iff either $\phi = 0$ or $\phi = 1$ or $\phi = \bigwedge_{i \leq n} \bigvee_{j \leq m_i} l^i_j \mid \phi = \bigvee_{i \leq n} \bigwedge_{j \leq m_i} l^i_j$ where l^i_j are literals.[4]

Lemma 3.1. *Let $\phi \in PropForm$. There exists a CNF $\psi \equiv \phi$.*

Proof. It is straightforward to prove that there exists $\vartheta \equiv \phi$ without any occurrence of $\neg$. The proof is by induction on the structure of ϕ using (2); every subformula of the form $\neg\varphi$ of ϕ is replaced with $\varphi \rightarrow 0 \equiv \neg\varphi$. We further prove the statement:

$$\text{There exists a } CNF\ \psi \equiv \vartheta. \quad (13)$$

The proof is by induction on the structure of ϑ; all the occurrences of $\rightarrow$ in ϑ are pushed down and the resulting *CNF* ψ is recursively built up. The obvious cases are $\vartheta \in PropAtom \cup \{0, 1\}$ and $\vartheta = \vartheta_1 \wedge \vartheta_2$. In the case $\vartheta = \vartheta_1 \vee \vartheta_2$, the distributivity of $\vee$ over $\wedge$, (3), is exploited.

[2] Using the commutativity, associativity, idempotence, monotonicity, neutral elements of $\vee$ and $\wedge$ will not be explicitly referred to.

[3] We assume the decreasing operator priority sequence $\overline{}^G$, $\wedge$, $\Rightarrow_G$, $\vee$, which enables writing order clauses without parentheses.

[4] Associativity of $\wedge$, $\vee$ will not be explicitly referred to, and hence, $\bigwedge_{i \leq n} \phi_i, \bigvee_{i \leq n} \phi_i \in PropForm$ are written without parentheses.

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

Let $\vartheta = \vartheta_1 \rightarrow \vartheta_2$. Then, by induction hypothesis, there exist *CNF*'s $\psi_1 \equiv \vartheta_1$, $\psi_2 \equiv \vartheta_2$, and we distinguish three cases for ψ_1, ψ_2. Case 1: either $\psi_1 = 0$ or $\psi_2 = 1$ is obvious; $\psi_1 \rightarrow \psi_2 \equiv 1$. Case 2: $\psi_1 = 1$ is also obvious; $\psi_1 \rightarrow \psi_2 \equiv \psi_2$. Case 3: neither $\psi_1 = 0$ nor $\psi_2 = 1$ nor $\psi_1 = 1$. Then $\psi_1 = \bigwedge_{i \le n} \bigvee_{j \le m_i} l^i_j$, l^i_j are literals, and we get two cases for ψ_2: either $\psi_2 = \bigwedge_{r \le v} \bigvee_{s \le u_r} k^r_s$, k^r_s are literals, or $\psi_2 = 0$. Using (6), (5), (8), (7), (3), in both the cases, there exists

$$\bigwedge_{\theta \le \Theta} \bigvee_{\xi \le \Xi_\theta} \lambda^\theta_\xi \rightarrow \kappa^\theta_\xi \equiv \psi_1 \rightarrow \psi_2 \overset{(IH)}{\equiv} \vartheta_1 \rightarrow \vartheta_2 = \vartheta, \quad (14)$$

λ^θ_ξ are literals, either κ^θ_ξ are literals or $\kappa^\theta_\xi = 0$. We show that

$$\text{for all } \theta \le \Theta \text{ and } \xi \le \Xi_\theta, \quad (15)$$
$$\text{there exists a } DNF\ \delta^\theta_\xi \equiv \lambda^\theta_\xi \rightarrow \kappa^\theta_\xi.$$

Let $\theta \le \Theta$ and $\xi \le \Xi_\theta$. We then distinguish nine cases for λ^θ_ξ and κ^θ_ξ. Case 3.1: $\lambda^\theta_\xi = a$ and $\kappa^\theta_\xi = b$, $a \in PropAtom$, $b \in PropAtom \cup \{0\}$. Hence, $\delta^\theta_\xi = a \rightarrow b = \lambda^\theta_\xi \rightarrow \kappa^\theta_\xi$ is a *DNF*. Case 3.2: $\lambda^\theta_\xi = a \rightarrow b$ and $\kappa^\theta_\xi = c$, $a \in PropAtom$, $b, c \in PropAtom \cup \{0\}$. Hence,

$$\delta^\theta_\xi = ((a \rightarrow b) \rightarrow b) \wedge (b \rightarrow c) \vee c \overset{(11)}{\equiv} (a \rightarrow b) \rightarrow c$$
$$= \lambda^\theta_\xi \rightarrow \kappa^\theta_\xi$$

is a *DNF*. Case 3.3: $\lambda^\theta_\xi = (a \rightarrow b) \rightarrow b$ and $\kappa^\theta_\xi = c$, $a \in PropAtom$, $b, c \in PropAtom \cup \{0\}$. Hence,

$$\delta^\theta_\xi = (a \rightarrow b) \wedge (b \rightarrow c) \vee c$$
$$\overset{(10)}{\equiv} (((a \rightarrow b) \rightarrow b) \rightarrow b) \wedge (b \rightarrow c) \vee c$$
$$\overset{(11)}{\equiv} ((a \rightarrow b) \rightarrow b) \rightarrow c = \lambda^\theta_\xi \rightarrow \kappa^\theta_\xi$$

is a *DNF*. Cases $3.4 - 3.9$: either $\lambda^\theta_\xi = a$ or $\lambda^\theta_\xi = a \rightarrow b$ or $\lambda^\theta_\xi = (a \rightarrow b) \rightarrow b$, and $\kappa^\theta_\xi = \varphi \rightarrow d$ where either $\varphi = c$ or $\varphi = c \rightarrow d$, $a, c \in PropAtom$, $b, d \in PropAtom \cup \{0\}$. By Cases $3.1 - 3.3$, there exists a *DNF* $\lambda^\theta_\xi \equiv \lambda^\theta_\xi \rightarrow d$, and

$$\delta^\theta_\xi = \lambda^\theta_\xi \vee \varphi \rightarrow d \equiv \lambda^\theta_\xi \rightarrow d \vee \varphi \rightarrow d \overset{(8)}{\equiv} \lambda^\theta_\xi \wedge \varphi \rightarrow d$$
$$\overset{(9)}{\equiv} \lambda^\theta_\xi \rightarrow (\varphi \rightarrow d) = \lambda^\theta_\xi \rightarrow \kappa^\theta_\xi$$

is a *DNF*. So, the claim (15) holds. We get that there exists a *CNF*

$$\psi \overset{((3))}{\equiv} \bigwedge_{\theta \le \Theta} \bigvee_{\xi \le \Xi_\theta} \delta^\theta_\xi \overset{(15)}{\equiv} \bigwedge_{\theta \le \Theta} \bigvee_{\xi \le \Xi_\theta} \lambda^\theta_\xi \rightarrow \kappa^\theta_\xi \overset{(14)}{\equiv} \vartheta.$$

Thus, the claim (13) holds. The induction is completed. We conclude that there exists a *CNF* $\psi \overset{(13)}{\equiv} \vartheta \equiv \phi$.

□

Using Lemma 3.1, we translate $(a \rightarrow b) \rightarrow ((b \rightarrow c) \rightarrow (a \rightarrow c)) \in PropForm$, $a, b, c \in PropAtom$, to an equivalent *CNF*:

$$(a \rightarrow b) \rightarrow ((b \rightarrow c) \rightarrow (a \rightarrow c)) \quad \overset{(9)}{\equiv}$$

$$(a \rightarrow b) \rightarrow (((b \rightarrow c) \wedge a) \rightarrow c) \quad \underset{(5)}{\overset{(8)}{\equiv}}$$

$$(a \rightarrow b) \rightarrow ((b \rightarrow c) \rightarrow c) \vee (a \rightarrow b) \rightarrow (a \rightarrow c) \overset{(9)}{\equiv}$$

$$((a \rightarrow b) \wedge (b \rightarrow c)) \rightarrow c \vee ((a \rightarrow b) \wedge a) \rightarrow c \quad \overset{(8)}{\equiv}$$

$$(a \rightarrow b) \rightarrow c \vee (b \rightarrow c) \rightarrow c \vee$$
$$(a \rightarrow b) \rightarrow c \vee a \rightarrow c \quad \equiv$$

$$(a \rightarrow b) \rightarrow c \vee (b \rightarrow c) \rightarrow c \vee a \rightarrow c \quad \overset{(11)}{\equiv}$$

$$(((a \rightarrow b) \rightarrow b) \wedge (b \rightarrow c)) \vee$$
$$c \vee (b \rightarrow c) \rightarrow c \vee a \rightarrow c \quad \overset{(3)}{\equiv}$$

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

In Lemma 3.1, we have laid no restrictions on the use of the distributivity law, (3), during translation to conjunctive normal form. Therefore the size of the output *CNF* may be exponential in the size of an input formula. To avoid this disadvantage, we propose translation to *CNF* via interpolation using new atoms, which produces *CNF* formulae in linear size. A similar approach exploiting the renaming subformulae technique can be found in (Plaisted, 1986; Boy, 1992; Hähnle, 1994b; Nonnengart, 1998; Sheridan, 2004). By $p^i_j \in PropAtom$ we denote atoms not yet occurring in the set of formulae in question. The empty sequence of symbols is denoted as ε. Let $\phi \in PropForm$. We define the size of ϕ by recursion on the structure of ϕ:

$$|\phi| = \begin{cases} 1 & \text{if } \phi \in PropAtom \cup \{0, 1\}, \\ |\phi_1| + 1 & \text{if } \phi = \neg\phi_1, \\ |\phi_1| + |\phi_2| + 1 & \text{if } \phi = \phi_1 \diamond \phi_2 \\ & \text{where } \diamond \in \{\wedge, \vee, \rightarrow\}. \end{cases}$$

Let $\phi_j \in PropForm$ and $p^i_j \in PropAtom$. We denote

$$\varphi^i_j = \begin{cases} \phi_j & \text{if } \phi_j \in PropAtom, \\ p^i_j & \text{if } \phi_j \notin PropAtom; \end{cases}$$

$${}^{+}\pi^i_j = \begin{cases} \varepsilon & \text{if } \phi_j \in PropAtom, \\ p^i_j \rightarrow \phi_j & \text{if } \phi_j \notin PropAtom; \end{cases}$$

$${}^{-}\pi^i_j = \begin{cases} \varepsilon & \text{if } \phi_j \in PropAtom, \\ \phi_j \rightarrow p^i_j & \text{if } \phi_j \notin PropAtom. \end{cases}$$

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

Table 1: Interpolation rules.

Case:	Positive interpolation Negative interpolation	Laws	$\frac{\text{Size of antecedent}}{\text{Maximum size of consequent}}$	
$\phi_1 \wedge \phi_2$	$\frac{p_0^i \to \phi_1 \wedge \phi_2}{(p_0^i \to \phi_1) \wedge (p_0^i \to \phi_2)}$	(6)	$\frac{\vert\phi_1\vert + \vert\phi_2\vert + 3}{\vert\phi_1\vert + \vert\phi_2\vert + 5}$	(16)
	$\frac{\phi_1 \wedge \phi_2 \to p_0^i}{(\varphi_1^i \to p_0^i \vee \varphi_2^i \to p_0^i) \wedge {}^-\pi_1^i \wedge {}^-\pi_2^i}$	(8)	$\frac{\vert\phi_1\vert + \vert\phi_2\vert + 3}{\vert\phi_1\vert + \vert\phi_2\vert + 13}$	(17)
$\phi_1 \vee \phi_2$	$\frac{p_0^i \to (\phi_1 \vee \phi_2)}{(p_0^i \to \varphi_1^i \vee p_0^i \to \varphi_2^i) \wedge {}^+\pi_1^i \wedge {}^+\pi_2^i}$	(5)	$\frac{\vert\phi_1\vert + \vert\phi_2\vert + 3}{\vert\phi_1\vert + \vert\phi_2\vert + 13}$	(18)
	$\frac{(\phi_1 \vee \phi_2) \to p_0^i}{\phi_1 \to p_0^i \wedge \phi_2 \to p_0^i}$	(7)	$\frac{\vert\phi_1\vert + \vert\phi_2\vert + 3}{\vert\phi_1\vert + \vert\phi_2\vert + 5}$	(19)
$\phi_1 \wedge \phi_2 \to 0$	$\frac{p_0^i \to (\phi_1 \wedge \phi_2 \to 0)}{(p_0^i \to 0 \vee \varphi_1^i \to 0 \vee \varphi_2^i \to 0) \wedge {}^-\pi_1^i \wedge {}^-\pi_2^i}$	$(9),(8)$	$\frac{\vert\phi_1\vert + \vert\phi_2\vert + 5}{\vert\phi_1\vert + \vert\phi_2\vert + 17}$	(20)
	$\frac{(\phi_1 \wedge \phi_2 \to 0) \to p_0^i}{((\phi_1 \to 0) \to p_0^i) \wedge ((\phi_2 \to 0) \to p_0^i)}$	$(8),(7)$	$\frac{\vert\phi_1\vert + \vert\phi_2\vert + 5}{\vert\phi_1\vert + \vert\phi_2\vert + 9}$	(21)
$(\phi_1 \vee \phi_2) \to 0$	$\frac{p_0^i \to ((\phi_1 \vee \phi_2) \to 0)}{(p_0^i \to (\phi_1 \to 0)) \wedge (p_0^i \to (\phi_2 \to 0))}$	$(7),(6)$	$\frac{\vert\phi_1\vert + \vert\phi_2\vert + 5}{\vert\phi_1\vert + \vert\phi_2\vert + 9}$	(22)
	$\frac{((\phi_1 \vee \phi_2) \to 0) \to p_0^i}{((\varphi_1^i \to 0) \to 0 \vee (\varphi_2^i \to 0) \to 0 \vee p_0^i) \wedge {}^+\pi_1^i \wedge {}^+\pi_2^i}$	$(11),(7),(8)$	$\frac{\vert\phi_1\vert + \vert\phi_2\vert + 5}{\vert\phi_1\vert + \vert\phi_2\vert + 19}$	(23)
$(\phi_1 \wedge \phi_2 \to 0) \to 0$	$\frac{p_0^i \to ((\phi_1 \wedge \phi_2 \to 0) \to 0)}{(p_0^i \to ((\phi_1 \to 0) \to 0)) \wedge (p_0^i \to ((\phi_2 \to 0) \to 0))}$	$(8),(7),(6)$	$\frac{\vert\phi_1\vert + \vert\phi_2\vert + 7}{\vert\phi_1\vert + \vert\phi_2\vert + 13}$	(24)
	$\frac{((\phi_1 \wedge \phi_2 \to 0) \to 0) \to p_0^i}{(\varphi_1^i \to 0 \vee \varphi_2^i \to 0 \vee p_0^i) \wedge {}^-\pi_1^i \wedge {}^-\pi_2^i}$	$(11),(10),(8)$	$\frac{\vert\phi_1\vert + \vert\phi_2\vert + 7}{\vert\phi_1\vert + \vert\phi_2\vert + 15}$	(25)
$((\phi_1 \vee \phi_2) \to 0) \to 0$	$\frac{p_0^i \to (((\phi_1 \vee \phi_2) \to 0) \to 0)}{(p_0^i \to 0 \vee (\varphi_1^i \to 0) \to 0 \vee (\varphi_2^i \to 0) \to 0) \wedge {}^+\pi_1^i \wedge {}^+\pi_2^i}$	$(9),(8),(7),(8)$	$\frac{\vert\phi_1\vert + \vert\phi_2\vert + 7}{\vert\phi_1\vert + \vert\phi_2\vert + 21}$	(26)
	$\frac{(((\phi_1 \vee \phi_2) \to 0) \to 0) \to p_0^i}{(((\phi_1 \to 0) \to 0) \to p_0^i) \wedge (((\phi_2 \to 0) \to 0) \to p_0^i)}$	$(7),(8),(7)$	$\frac{\vert\phi_1\vert + \vert\phi_2\vert + 7}{\vert\phi_1\vert + \vert\phi_2\vert + 13}$	(27)
$((\phi_1 \to 0) \to 0) \to 0$	$\frac{p_0^i \to (((\phi_1 \to 0) \to 0) \to 0)}{p_0^i \to (\phi_1 \to 0)}$	(10)	$\frac{\vert\phi_1\vert + 8}{\vert\phi_1\vert + 4}$	(28)
	$\frac{(((\phi_1 \to 0) \to 0) \to 0) \to p_0^i}{(\phi_1 \to 0) \to p_0^i}$	(10)	$\frac{\vert\phi_1\vert + 8}{\vert\phi_1\vert + 4}$	(29)
$((\phi_1 \to \phi_2) \to 0) \to 0, \phi_2 \neq 0$	$\frac{p_0^i \to (((\phi_1 \to \phi_2) \to 0) \to 0)}{(p_0^i \to 0 \vee \varphi_1^i \to 0 \vee (\varphi_2^i \to 0) \to 0) \wedge {}^-\pi_1^i \wedge {}^+\pi_2^i}$	$(9),(8),(12),(8),(10)$	$\frac{\vert\phi_1\vert + \vert\phi_2\vert + 7}{\vert\phi_1\vert + \vert\phi_2\vert + 19}$	(30)
	$\frac{(((\phi_1 \to \phi_2) \to 0) \to 0) \to p_0^i}{((\phi_1 \to 0) \to p_0^i) \wedge (((\phi_2 \to 0) \to 0) \to p_0^i)}$	$(12),(8),(10),(7)$	$\frac{\vert\phi_1\vert + \vert\phi_2\vert + 7}{\vert\phi_1\vert + \vert\phi_2\vert + 11}$	(31)
$(\phi_1 \to \phi_2) \to 0, \phi_2 \neq 0$	$\frac{p_0^i \to ((\phi_1 \to \phi_2) \to 0)}{(p_0^i \to ((\phi_1 \to 0) \to 0)) \wedge (p_0^i \to (\phi_2 \to 0))}$	$(12),(6)$	$\frac{\vert\phi_1\vert + \vert\phi_2\vert + 5}{\vert\phi_1\vert + \vert\phi_2\vert + 11}$	(32)
	$\frac{((\phi_1 \to \phi_2) \to 0) \to p_0^i}{(\varphi_1^i \to 0 \vee (\varphi_2^i \to 0) \to 0 \vee p_0^i) \wedge {}^-\pi_1^i \wedge {}^+\pi_2^i}$	$(11),(12),(8),(10)$	$\frac{\vert\phi_1\vert + \vert\phi_2\vert + 5}{\vert\phi_1\vert + \vert\phi_2\vert + 17}$	(33)
$\phi_1 \to \phi_2, \phi_2 \neq 0$	$\frac{p_0^i \to (\phi_1 \to \phi_2)}{(p_0^i \to \varphi_2^i \vee \varphi_1^i \to \varphi_2^i) \wedge {}^-\pi_1^i \wedge {}^+\pi_2^i}$	$(9),(8)$	$\frac{\vert\phi_1\vert + \vert\phi_2\vert + 3}{\vert\phi_1\vert + \vert\phi_2\vert + 13}$	(34)
	$\frac{(\phi_1 \to \phi_2) \to p_0^i}{((\varphi_1^i \to \varphi_2^i) \to \varphi_2^i \vee p_0^i) \wedge (\varphi_2^i \to p_0^i) \wedge {}^+\pi_1^i \wedge {}^-\pi_2^i}$	$(11),(3)$	$\frac{\vert\phi_1\vert + \vert\phi_2\vert + 3}{\vert\phi_1\vert + \vert\phi_2\vert + 17}$	(35)

SCITEPRESS
SCIENCE AND TECHNOLOGY PUBLICATIONS

Let $\phi_1, \phi_2 \in \mathit{PropForm}$ and $p_j^i \in \mathit{PropAtom}$. In Table 1, we introduce interpolation rules. Let $\phi \in \mathit{PropForm}$. ψ is a *CNF* of ϕ iff ψ is a *CNF* obtained from $p^i \wedge (p^i \rightarrow \phi)$ for some i by a finite derivation using the interpolation rules. We denote the set of all *CNF*'s of ϕ as $\mathit{CNF}(\phi)$. Let $f, g : M \longrightarrow \mathbb{N}$. $f \in O(g)$ iff there exists k such that for all $m \in M$, $f(m) \leq k.g(m)$.

Lemma 3.2. *Let $\phi \in \mathit{PropForm}$. $\mathit{CNF}(\phi) \neq \emptyset$, and for all $\psi \in \mathit{CNF}(\phi)$, ψ is equisatisfiable to ϕ, $|\psi| \in O(|\phi|)$.*

Proof. The proof of $\mathit{CNF}(\phi) \neq \emptyset$ is by induction on the structure of ϕ. It is straightforward to prove that $p^i \wedge (p^i \rightarrow \phi)$ is equisatisfiable to ϕ; for every interpolation rule, its antecedent is equisatisfiable to its consequent; if for every i, ψ_i is equisatisfiable to ϕ_i, then so is $\bigwedge_i \psi_i$ to $\bigwedge_i \phi_i$; there exists k such that for every interpolation rule, the size of its consequent is less than or equal to k times the size of its antecedent. Let $\psi \in \mathit{CNF}(\phi)$. Then there exist i, n, a finite derivation $\zeta_0 = p^i \wedge (p^i \rightarrow \phi), \dots, \zeta_n = \psi$, and k such that for all $j \leq n$, ζ_j is equisatisfiable to ϕ and $|\zeta_j| \leq k.|\phi|$. The proof is by induction on n using the previous statements. $\square$

Using Lemma 3.2, we translate $(a \rightarrow b) \rightarrow ((b \rightarrow c) \rightarrow (a \rightarrow c)) \in \mathit{PropForm}$, $a, b, c \in \mathit{PropAtom}$, to an equisatisfiable *CNF*:

$$p_0^0 \wedge (p_0^0 \rightarrow \underbrace{(a \rightarrow b)}_{p_1^0} \rightarrow \underbrace{((b \rightarrow c) \rightarrow (a \rightarrow c))}_{p_2^0}), \qquad (34)$$

$$\begin{aligned} & p_0^0 \wedge (p_0^0 \rightarrow p_2^0 \vee p_1^0 \rightarrow p_2^0) \wedge \\ & \quad ((a \rightarrow b) \rightarrow p_1^0) \wedge \\ & \qquad (p_2^0 \rightarrow (\underbrace{(b \rightarrow c)}_{p_1^1} \rightarrow \underbrace{(a \rightarrow c)}_{p_2^1})), \quad (35), (34) \end{aligned}$$

$$\begin{aligned} & p_0^0 \wedge (p_0^0 \rightarrow p_2^0 \vee p_1^0 \rightarrow p_2^0) \wedge \\ & \quad ((a \rightarrow b) \rightarrow b \vee p_1^0) \wedge (b \rightarrow p_1^0) \wedge \\ & \qquad (p_2^0 \rightarrow p_2^1 \vee p_1^1 \rightarrow p_2^1) \wedge \\ & \qquad\quad ((b \rightarrow c) \rightarrow p_1^1) \wedge \\ & \qquad\quad (p_2^1 \rightarrow (a \rightarrow c)), \qquad (35), (34) \end{aligned}$$

$$\begin{aligned} & p_0^0 \wedge (p_0^0 \rightarrow p_2^0 \vee p_1^0 \rightarrow p_2^0) \wedge \\ & \quad ((a \rightarrow b) \rightarrow b \vee p_1^0) \wedge (b \rightarrow p_1^0) \wedge \\ & \qquad (p_2^0 \rightarrow p_2^1 \vee p_1^1 \rightarrow p_2^1) \wedge \\ & \qquad\quad ((b \rightarrow c) \rightarrow c \vee p_1^1) \wedge (c \rightarrow p_1^1) \wedge \\ & \qquad\quad (p_2^1 \rightarrow c \vee a \rightarrow c). \end{aligned}$$

We further introduce a ground fragment of the first-order two-valued logic with equality and strict order. The syntax is given by a class of order clausal theories. We form order literals and clauses from $\mathit{PropAtom} \cup \{\mathit{0}, \mathit{1}\}$, regarded as constants, using binary predicates $\eqsim$, equality, and $\prec$, strict order. l is an order literal iff either $l = a \eqsim b = b \eqsim a$; since equality is commutative by definition, we identify $a \eqsim b$ and $b \eqsim a$; or $l = a \prec b$ where $a, b \in \mathit{PropAtom} \cup \{\mathit{0}, \mathit{1}\}$. An order clause is a finite set of order literals. An order clause $\{l_1, \dots, l_n\}$ is written in the form $l_1 \vee \cdots \vee l_n$. The order clause $\emptyset$ is called the empty clause and denoted as $\square$. An order clause $\{l\}$ is called a unit order clause and denoted as l if it does not cause the ambiguity with the denotation of the single literal l in a given context. We designate the set of order clauses as *OrdCl*. Let $l, l_1, \dots, l_n$ be order literals and $C, C' \in \mathit{OrdCl}$. By $l \vee C$ we denote $\{l\} \cup C$ where $l \notin C$. Analogously, by $\bigvee_{i=1}^n l_i \vee C$ we denote $\{l_1\} \cup \cdots \cup \{l_n\} \cup C$ where for all $1 \leq i \neq i' \leq n$, $l_i \notin C$ and $l_i \neq l_{i'}$. By $C \vee C'$ we denote $C \cup C'$. C is a subclause of C', in symbols $C \sqsubseteq C'$, iff $C \subseteq C'$. An order clausal theory is a set of order clauses. A unit order clausal theory is a set of unit order clauses. Let $T, T' \subseteq \mathit{OrdCl}$. By $\mathit{atoms}(C) \mid \mathit{atoms}(T) \subseteq \mathit{PropAtom}$ we denote the set of all the propositional atoms occurring in $C \mid T$. The semantics is given by a class of order interpretations. An order interpretation $\mathcal{I}$ with the domain $\mathit{dom}(\mathcal{I}) = \mathit{PropAtom}$ is a first-order two-valued interpretation such that $\mathcal{U}_\mathcal{I} = [0,1]$, for all $a \in \mathit{PropAtom}$, $a^\mathcal{I} \in [0,1]$, $\mathit{0}^\mathcal{I} = 0$, $\mathit{1}^\mathcal{I} = 1$, and $\eqsim^\mathcal{I} = =_{[0,1]}$, $\prec^\mathcal{I} = <_{[0,1]}$. A partial order interpretation $\mathcal{I}$ with the domain $\mathit{dom}(\mathcal{I}) \subseteq \mathit{PropAtom}$ is an order interpretation such that for all $a \in \mathit{dom}(\mathcal{I})$, $a^\mathcal{I} \in [0,1]$. An (partial) order interpretation $\mathcal{I}$ is identified with the (partial) valuation $\mathcal{V}_\mathcal{I} : \mathit{dom}(\mathcal{V}_\mathcal{I}) \longrightarrow [0,1]$, $\mathcal{V}_\mathcal{I}(a) = a^\mathcal{I}$. Let $\mathit{atoms}(l)$, $\mathit{atoms}(C)$, $\mathit{atoms}(C')$, $\mathit{atoms}(T)$, $\mathit{atoms}(T') \subseteq \mathit{dom}(\mathcal{I})$. $\mathcal{I}$ is a (partial) model of l, in symbols $\mathcal{I} \models l$, iff either for $l = a \eqsim b$, $a^\mathcal{I} =_{[0,1]} b^\mathcal{I}$, or for $l = a \prec b$, $a^\mathcal{I} <_{[0,1]} b^\mathcal{I}$. $\mathcal{I}$ is a (partial) model of C, in symbols $\mathcal{I} \models C$, iff there exists $l \in C$ such that $\mathcal{I} \models l$. $\mathcal{I}$ is a (partial) model of T, in symbols $\mathcal{I} \models T$, iff for all $C \in T$, $\mathcal{I} \models C$. Note that $\square$ and T such that $\square \in T$ are unsatisfiable by definition. C' is an order consequence of C, in symbols $C \models_O C'$, iff for every model $\mathcal{I}$ of C, $\mathcal{I} \models C'$. C is an order consequence of T, in symbols $T \models_O C$, iff for every model $\mathcal{I}$ of T, $\mathcal{I} \models C$. T' is an order consequence of T, in symbols $T \models_O T'$, iff for every model $\mathcal{I}$ of T, $\mathcal{I} \models T'$. $C \mid T$ is satisfiable iff there exists a model of $C \mid T$. $C' \mid T'$ is equisatisfiable to $C \mid T$ iff $C' \mid T'$ is satisfiable if and only if $C \mid T$ is satisfiable. By *OrdPropForm* we designate the augmented set of all order propositional formulae built up from *PropAtom* using 0, 1, $\neg$, $\wedge$, $\vee$, $\rightarrow$, and $\prec$, $\eqcirc$. Note that $OrdPropForm \supseteq PropForm$ by definition, and all the notions and notation concerned with *PropForm* are straightforwardly extended to *OrdPropForm*.

Lemma 3.3. *Let* ϕ *be a conjunctive normal form. There exists* $T_\phi \subseteq_{\mathcal{F}} OrdCl$ *such that* T_ϕ *is equisatisfiable to* ϕ.

Proof. By the definition of *CNF*, we distinguish three cases for ϕ. Case 1: $\phi = 0$. Then ϕ is unsatisfiable and $T_\phi = \{\square\} \subseteq_{\mathcal{F}} OrdCl$ is unsatisfiable as well. So, the claim holds. Case 2: $\phi = 1$. Then ϕ is satisfiable and $T_\phi = \emptyset \subseteq_{\mathcal{F}} OrdCl$ is satisfiable as well. So, the claim holds. Case 3: $\phi = \bigwedge_{i \le n} \bigvee_{j \le m_i} l^i_j$, l^i_j are literals.

$$\text{For all } i \le n \text{ and } j \le m_i \text{, there exists } C^i_j \in OrdCl \text{ such that } C^i_j \text{ is equisatisfiable to } l^i_j. \tag{36}$$

The proof is by definition. We get five cases for l^i_j. Case 3.1: $l^i_j = a$, $a \in PropAtom$. Then $C^i_j = a \eqcirc 1$. Case 3.2: $l^i_j = a \rightarrow 0$, $a \in PropAtom$. Then $C^i_j = a \eqcirc 0$. Case 3.3: $l^i_j = a \rightarrow b$, $a \in PropAtom$, $b \in PropAtom$. Then $C^i_j = a \prec b \vee a \eqcirc b \vee b \eqcirc 1$. Case 3.4: $l^i_j = (a \rightarrow 0) \rightarrow 0$, $a \in PropAtom$. Then $C^i_j = 0 \prec a$. Case 3.5: $l^i_j = (a \rightarrow b) \rightarrow b$, $a \in PropAtom$, $b \in PropAtom$. Then $C^i_j = b \prec a \vee b \eqcirc 1$. So, the claim (36) holds. By (36), there exists $T_\phi \subseteq_{\mathcal{F}} OrdCl$ such that $T_\phi = \{\bigvee_{j \le m_i} C^i_j \mid i \le n\}$ is equisatisfiable to ϕ.

$\square$

Using Lemma 3.3, we translate the *CNF* $((a \rightarrow b) \rightarrow b \vee c \vee (b \rightarrow c) \rightarrow c \vee a \rightarrow c) \wedge (b \rightarrow c \vee c \vee (b \rightarrow c) \rightarrow c \vee a \rightarrow c)$, $a, b, c \in PropAtom$, to an equisatisfiable $T \subseteq_{\mathcal{F}} OrdCl$ where

$$\begin{aligned} T = \{ & b \prec a \vee b \eqcirc 1 \vee c \eqcirc 1 \vee \\ & c \prec b \vee c \eqcirc 1 \vee a \prec c \vee a \eqcirc c \vee c \eqcirc 1, \\ & b \prec c \vee b \eqcirc c \vee c \eqcirc 1 \vee c \eqcirc 1 \vee \\ & c \prec b \vee c \eqcirc 1 \vee a \prec c \vee a \eqcirc c \vee c \eqcirc 1 \}. \end{aligned}$$

4 *DPLL* PROCEDURE

We devise a variant of the *DPLL* procedure over finite order clausal theories. At first, a minimal set of basic rules is introduced. Let l, l_1, l_2, l_3 be order literals. l is a contradiction iff either $l = 0 \eqcirc 1$ or $l = 0 \prec 0$ or $l = 1 \prec 1$ or $l = a \prec 0$ or $l = 1 \prec a$ or $l = a \prec a$ where $a \in PropAtom$. l is a tautology iff either $l = 0 \eqcirc 0$ or $l = 1 \eqcirc 1$ or $l = 0 \prec 1$ or $l = a \eqcirc a$ where $a \in PropAtom$. $l_1 \vee l_2 \vee l_3$ is a general trichotomy iff $l_1 = a \prec b$, $l_2 = a \eqcirc b$, $l_3 = b \prec a$ where $a, b \in PropAtom \cup \{0, 1\}$. Let $T \subseteq OrdCl$. The basic rules are as follows:

(37) (*One literal contradiction simplification rule*)

$$\frac{T}{T \cup \{\square\}}$$

if T *is a unit order clausal theory*, $l \in T$, *and* l *is a contradiction*;

(38) (*One literal transitivity rule of* $\eqcirc$ *and* $\prec$)

$$\frac{T}{T \cup \{a \diamond c\}} \quad \textit{where } \diamond = \begin{cases} \eqcirc \textit{ if } \diamond_1 = \diamond_2 = \eqcirc, \\ \prec \textit{ else}, \end{cases}$$

if T *is a unit order clausal theory*, $a \diamond_1 b, b \diamond_2 c \in T$, *and* $\diamond_1, \diamond_2 \in \{\eqcirc, \prec\}$;

(39) (*General trichotomy branching rule*)

$$\frac{T}{T - \{l_1 \vee C\} \cup \{l_1\} \mid T - \{l_1 \vee C\} \cup \{C\} \cup \{l_2\} \mid T - \{l_1 \vee C\} \cup \{C\} \cup \{l_3\}}$$

if $l_1 \vee C \in T$, $C \neq \square$, *and* $l_1 \vee l_2 \vee l_3$ *is a general trichotomy.*

Rule (39) reflects the linearity of $<_{[0,1]}$ in terms of general trichotomy. Rule (37) formalises its additional properties: the endpoints $0 <_{[0,1]} 1$ and strictness via contradictions. Rule (38) expresses the mutual transitivity of $=_{[0,1]}$ together with $<_{[0,1]}$. Rules (37), (38), (39) are sound in view of satisfiability:

$$T \text{ and } T \cup \{\square\} \text{ in the consequent of Rule (37) are both unsatisfiable.} \tag{40}$$

$$T \text{ is equisatisfiable to } T \cup \{a \diamond c\} \text{ in the consequent of Rule (38).} \tag{41}$$

$$\begin{aligned} & \text{Let } \mathcal{I} \text{ be a partial order interpretation, } dom(\mathcal{I}) \supseteq atoms(T). \\ & \mathcal{I} \models T \text{ if and only if } \mathcal{I} \models T - \{l_1 \vee C\} \cup \{l_1\} \text{ or} \\ & \quad \mathcal{I} \models T - \{l_1 \vee C\} \cup \{C\} \cup \{l_2\} \text{ or} \\ & \quad \mathcal{I} \models T - \{l_1 \vee C\} \cup \{C\} \cup \{l_3\} \\ & \quad \text{in the consequent of Rule (39).} \end{aligned} \tag{42}$$

$$\begin{aligned} & T \text{ is satisfiable if and only if} \\ & \quad T - \{l_1 \vee C\} \cup \{l_1\} \text{ or} \\ & \quad T - \{l_1 \vee C\} \cup \{C\} \cup \{l_2\} \text{ or} \\ & \quad T - \{l_1 \vee C\} \cup \{C\} \cup \{l_3\} \\ & \quad \text{in the consequent of Rule (39) is satisfiable.} \end{aligned} \tag{43}$$

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

The proof is by assumption and definition. The refutational completeness argument of the basic rules, Theorem 4.1(ii), may be provided using the excess literal technique (Anderson, 1970). From this point of view, Rules (37) and (38) handle the base case: T is a unit order clausal theory, while Rule (39) the induction one: it subtracts the excess literal measure of T at least by 1 for every clausal theory in a branch of its consequent.

T is closed under Rules (37) and (38) iff for every application of Rules (37) and (38) of the form $\frac{T}{T'}$, $T' = T$. By $trans(T) \subseteq OrdCl$ we denote the least set such that $trans(T) \supseteq T$ and $trans(T)$ is closed under Rules (37), (38).

Using the basic rules, one can construct a finitely generated tree with the input theory as the root in the usual manner, so as the classical *DPLL* procedure does; for every parent vertex, there exists an application of Rule (37) or (38) or (39) such that the parent vertex is the theory in its antecedent and the children vertices are the theories in its consequent. A branch of a tree is closed iff it contains a vertex T' such that $\square \in T'$. A branch of a tree is open iff it is not closed. A tree is closed iff every its branch is finite and closed. Note that a closed tree is finite by König's Lemma. A tree is open iff it is not closed. A tree is linear iff it consists of only one branch, beginning from its root and ending in its only leaf.

Lemma 4.1. *Let $T \subseteq OrdCl$.*

(i) *If $T \subseteq_{\mathcal{F}} OrdCl$, then $trans(T) \subseteq_{\mathcal{F}} OrdCl$.*
(ii) *If T is a unit order clausal theory and $\square \notin trans(T)$, then $trans(T)$ is a unit order clausal theory.*
(iii) *$atoms(trans(T)) = atoms(T)$.*
(iv) *$T \models_O trans(T)$.*
(v) *If $T \subseteq_{\mathcal{F}} OrdCl$, then there exists a finite linear tree with the root T and the leaf $trans(T)$ constructed using Rules (37) and (38).*

Proof. By assumption and definition. $\square$

The following lemma shows that Rules (37) and (38) are refutation complete for a (countable) unit order clausal theory, which will be exploited in the base case of Theorem 4.1(ii).

Lemma 4.2. *Let $T = trans(T) \subseteq OrdCl$ be a countable unit order clausal theory. There exists a partial model $\mathfrak{A}$ of T, $dom(\mathfrak{A}) = atoms(T)$.*

Proof. By the theorem assumption that T is a unit order clausal theory, $\square \notin T = trans(T)$. In addition, by the theorem assumption that T is a countable set, there exists a sequence δ of $atoms(T)$. At first, we define partial order interpretations Mod_α by recursion on $\alpha \leq \omega$:

$$Mod_0 = \emptyset;$$

$$Mod_\alpha = Mod_{\alpha-1} \cup \{(\delta(\alpha-1), \nu_{\alpha-1})\} \ (0 < \alpha < \omega),$$

$$\mathbb{M}_{\alpha-1} = \{\|a\|^{Mod_{\alpha-1}} \mid a \eqcirc \delta(\alpha-1) \in T, \ a \in dom(Mod_{\alpha-1}) \cup \{0, 1\}\},$$

$$\mathbb{S}_{\alpha-1} = \{Mod_{\alpha-1}(a) \mid a \prec \delta(\alpha-1) \in T, \ a \in dom(Mod_{\alpha-1})\},$$

$$\mathbb{I}_{\alpha-1} = \{Mod_{\alpha-1}(a) \mid \delta(\alpha-1) \prec a \in T, \ a \in dom(Mod_{\alpha-1})\},$$

$$\nu_{\alpha-1} = \begin{cases} \frac{\bigvee \mathbb{S}_{\alpha-1} + \bigwedge \mathbb{I}_{\alpha-1}}{2}, & \mathbb{M}_{\alpha-1} = \emptyset, \\ \bigvee \mathbb{M}_{\alpha-1}, & \mathbb{M}_{\alpha-1} \neq \emptyset; \end{cases}$$

$$Mod_\omega = \bigcup_{\alpha < \omega} Mod_\alpha.$$

It is straightforward to prove the following statements:

$$\text{For all } \alpha \leq \omega,\ Mod_\alpha \text{ is a partial order interpretation, } dom(Mod_\alpha) = \delta[\alpha], \text{ and for all } \beta \leq \alpha,\ Mod_\beta \subseteq Mod_\alpha. \quad (44)$$

$$\text{For all } \alpha \leq \omega \text{ and } l \in T \text{ such that } atoms(l) \subseteq dom(Mod_\alpha),\ Mod_\alpha \models l. \quad (45)$$

$$\text{For all } \alpha \leq \omega \text{ and } a \in dom(Mod_\alpha), \text{ if } Mod_\alpha(a) = 0, \text{ then } a \eqcirc 0 \in T. \quad (46)$$

$$\text{For all } \alpha \leq \omega \text{ and } a \in dom(Mod_\alpha), \text{ if } Mod_\alpha(a) = 1, \text{ then } a \eqcirc 1 \in T. \quad (47)$$

The proofs are by induction on $\alpha \leq \omega$. We put $\mathfrak{A} = Mod_\omega$. By (44), $\mathfrak{A} = Mod_\omega$ is a partial order interpretation, $dom(\mathfrak{A}) = dom(Mod_\omega) \stackrel{(44)}{=\!=} \delta[\omega] = atoms(T)$. Let $l \in T$. Then $atoms(l) \subseteq atoms(T) = dom(Mod_\omega) = dom(\mathfrak{A})$ and $\mathfrak{A} = Mod_\omega \stackrel{(45)}{\models\!=\!=} l$. So, $\mathfrak{A} \models T$. We conclude that $\mathfrak{A}$ is a partial model of T, $dom(\mathfrak{A}) = atoms(T)$. $\square$

The *DPLL* procedure is refutation sound and complete.

Theorem 4.1 (Refutational Soundness and Completeness of the *DPLL* Procedure)**.** *Let $T \subseteq_{\mathcal{F}} OrdCl$.*

(i) *If there exists a closed tree Tree with the root T constructed using Rules (37), (38), (39), then T is unsatisfiable.*

(ii) *There exists a finite tree Tree with the root T constructed using Rules* (37), (38), (39) *with the following properties:*

$$\textit{If } T \textit{ is unsatisfiable, then Tree is closed.} \quad (48)$$

$$\textit{If } T \textit{ is satisfiable, then Tree is open and there exists a partial model } \mathfrak{A} \textit{ of } T, \; dom(\mathfrak{A}) = atoms(T), \textit{ related to Tree.} \quad (49)$$

Proof. (i) The proof is by induction on the structure of *Tree* using (40), (41), (42).

(ii) We exploit the excess literal technique to construct a finite tree *Tree* with the root T using Rules (37), (38), (39). Let $T^F \subseteq_{\mathcal{F}} OrdCl$. We define $elmeasure(T^F) = (\sum_{C \in T^F} |C|) - |T^F|$. We proceed by induction on $elmeasure(T)$.

Let $elmeasure(T) = 0$. We distinguish two cases:

$$\text{either } \Box \in T \text{ or } \Box \notin T.$$

Case 1: $\Box \in T$. Then T is unsatisfiable and $Tree = T$ is a closed tree with the root T. So, (48) holds and (49) holds trivially.

Case 2: $\Box \notin T$. Then, by the denotation of $elmeasure(T)$, T is a unit order clausal theory. By Lemma 4.1(v), there exists a finite linear tree *Tree* with the root T and the leaf $trans(T)$ constructed using Rules (37) and (38). We get two cases:

$$\text{either } \Box \in trans(T) \text{ or } \Box \notin trans(T).$$

Case 2.1: $\Box \in trans(T)$. Then *Tree* is a closed tree with the root T; its only branch from T to $trans(T)$ is closed. Hence, by (i), T is unsatisfiable. So, (48) holds and (49) holds trivially.

Case 2.2: $\Box \notin trans(T)$. Then *Tree* is an open tree with the root T; its only branch from T to $trans(T)$ is open. Since T is a unit order clausal theory, by Lemma 4.1(ii), we get $trans(T)$ is a unit order clausal theory, and by Lemma 4.2 for $trans(T)$, there exists a partial model $\mathfrak{A}$ of $trans(T)$, $dom(\mathfrak{A}) = atoms(trans(T))$. Hence, $\mathfrak{A}$ is a partial model of $T \subseteq trans(T)$, $dom(\mathfrak{A}) = atoms(trans(T)) \overset{\text{Lemma 4.1(iii)}}{=\!=\!=} atoms(T)$, related to *Tree* and T is satisfiable. So, (49) holds and (48) holds trivially.

Let $elmeasure(T) = n > 0$ and the statement hold for all $T^F \subseteq_{\mathcal{F}} OrdCl$ such that $elmeasure(T^F) < n$. Since $elmeasure(T) > 0$, by the denotation of $elmeasure(T)$, there exists $l_1 \vee C \in T$ such that $C \neq \Box$. Let l_2, l_3 be order literals such that $l_1 \vee l_2 \vee l_3$ is a general trichotomy. This yields the application of Rule (39)

$$\frac{T}{(T - \{l_1 \vee C\}) \cup \{l_1\} \mid (T - \{l_1 \vee C\}) \cup \{C\} \cup \{l_2\} \mid (T - \{l_1 \vee C\}) \cup \{C\} \cup \{l_3\}}.$$

We denote $T_1 = (T - \{l_1 \vee C\}) \cup \{l_1\}$, $T_2 = (T - \{l_1 \vee C\}) \cup \{C\} \cup \{l_2\}$, $T_3 = (T - \{l_1 \vee C\}) \cup \{C\} \cup \{l_2\}$. Then $elmeasure(T_1) < elmeasure(T)$, $elmeasure(T_2) < elmeasure(T)$, $elmeasure(T_3) < elmeasure(T)$, and by induction hypothesis, there exist finite trees $Tree_1$ with the root T_1, $Tree_2$ with the root T_2, $Tree_3$ with the root T_3 constructed using Rules (37), (38), (39) such that (48) and (49) hold for $Tree_1$, $Tree_2$, $Tree_3$. This yields

$$Tree = \frac{T}{Tree_1 \mid Tree_2 \mid Tree_3}$$

is a finite tree with the root T constructed using Rules (37), (38), (39). We get two cases:

$$\text{either } T \text{ is unsatisfiable or } T \text{ is satisfiable.}$$

Case 4: T is unsatisfiable. Then, by (42), T_1, T_2, T_3 are unsatisfiable, and by (48) for $Tree_1$, $Tree_2$, $Tree_3$, $Tree_1$, $Tree_2$, $Tree_3$ are closed trees. Hence, *Tree* is a closed tree. So, (48) holds and (49) holds trivially for *Tree*.

Case 5: T is satisfiable. Then, by (42), there exists $1 \leq i \leq 3$ such that T_i is satisfiable. Hence, by (49) for $Tree_i$, $Tree_i$ is an open tree and there exists a partial model $\mathfrak{A}_i$ of T_i, $dom(\mathfrak{A}_i) = atoms(T_i)$, related to $Tree_i$. By the definition of T_i, $T_i \models_O T$. As $\{l_1, l_2, l_3\}$ is a trichotomy, $atoms(l_1) = atoms(l_2) = atoms(l_3)$ and $atoms(T_i) \subseteq atoms(T)$. We get *Tree* is an open tree and $\mathfrak{A} = \mathfrak{A}_i \cup \{(p, 0) \mid p \in atoms(T) - atoms(T_i)\}$, $dom(\mathfrak{A}) = atoms(T)$, is a partial model of T related to *Tree*. So, (49) holds and (48) holds trivially for *Tree*. The induction is completed.

□

The set of basic rules has been proposed as a minimal one, which is suitable for theoretical purposes; e.g. not to have a too complicated completeness argument. For practical computing, it may be augmented by additional rules. Let l, l_1, l_2, l_3 be order literals and $C \in OrdCl$. $l_1 \vee l_2$ is a dichotomy iff either $l_1 = 0 \eqcirc a$ and $l_2 = 0 \prec a$ or $l_1 = a \prec 1$ and $l_2 = a \eqcirc 1$ where $a \in PropAtom$. $l_1 \vee l_2 \vee l_3$ is a trichotomy iff $l_1 = a \prec b$, $l_2 = a \eqcirc b$, $l_3 = b \prec a$ where $a, b \in PropAtom$. C is a tautology iff there exists $C' \in OrdCl$ such that $C' \sqsubseteq C$ and either $C' = \{l\}$ where l is a tautology or C' is a dichotomy or C' is a trichotomy.

(50) (*Contradiction simplification rule*)

$$\frac{T}{(T - \{l \vee C\}) \cup \{C\}}$$

if $l \vee C \in T$ *and* l *is a contradiction*;

(51) (*Tautology simplification rule*)

$$\frac{T}{T - \{C\}} \tag{52}$$

if $C \in T$ and C is a tautology;

(53) (*One literal positive propagation rule*)

$$\frac{T}{T - \{C\}}$$

if $l, C \in T$, $l \in C$, and l is a literal;

(54) (*One literal negative propagation rule*)

$$\frac{T}{(T - \{l_2 \vee C\}) \cup \{C\}}$$

if $l_1, l_2 \vee C \in T$ and there exists l_3 such that $l_1 \vee l_2 \vee l_3$ is a general trichotomy;

(55) (*Dichotomy branching rule*)

$$\frac{T}{(T - \{l_1 \vee C\}) \cup \{l_1\} \mid (T - \{l_1 \vee C\}) \cup \{C\} \cup \{l_2\}}$$

if $l_1 \vee C \in T$, $C \neq \square$, and $l_1 \vee l_2$ is a dichotomy;

(56) (*Trichotomy branching rule*)

$$\frac{T}{\begin{array}{l}(T - \{l_1 \vee C\}) \cup \{l_1\} \mid \\ (T - \{l_1 \vee C\}) \cup \{C\} \cup \{l_2\} \mid \\ (T - \{l_1 \vee C\}) \cup \{C\} \cup \{l_3\}\end{array}}$$

if $l_1 \vee C \in T$, $C \neq \square$, and $l_1 \vee l_2 \vee l_3$ is a trichotomy.

Rules (50), (51), (53), (54), (55), (56) are obviously sound and helpful for constructing more compact *DPLL* trees in many cases, however, superfluous for the completeness argument. Concerning the *SAT* problem of a formula, we conclude.

Corollary 4.1. *Let $\phi \in PropForm$. There exist an equisatisfiable $T_\phi \subseteq_{\mathcal{F}} OrdCl$ to ϕ and a finite tree $Tree_\phi$ with the root T_ϕ constructed using Rules* (37), (38), (39) *with the following properties:*

$$\text{If } \phi \text{ is unsatisfiable, then } Tree_\phi \text{ is closed.} \tag{57}$$

$$\text{If } \phi \text{ is satisfiable, then } Tree_\phi \text{ is open and there exists a partial model } \mathfrak{A}_\phi \text{ of } \phi, \; dom(\mathfrak{A}_\phi) = atoms(\phi). \tag{58}$$

Proof. An immediate consequence of Lemma 3.3 and Theorem 4.1.

□

Note that the *SAT* problem of a finite theory can be reduced to the *SAT* one of a formula in the usual manner. Let $T = \{\phi_i \mid i \leq n\} \subseteq_{\mathcal{F}} PropForm$. Then $\phi = \bigwedge_{i \leq n} \phi_i \in PropForm$ is equisatisfiable to T.

5 TAUTOLOGY CHECKING

One application of the *DPLL* procedure may be to tautology checking. Let $\phi \in PropForm$. ϕ is a tautology (valid) iff for every valuation $\mathcal{V}$, $\mathcal{V} \models \phi$. As explained in Introduction, the *VAL* problem of a formula ϕ can be reduced to the unsatisfiability of the order formula $\phi \prec 1$ consequently translated to an equisatisfiable finite order clausal theory T_ϕ. Then the unsatisfiability of T_ϕ is decided by the *DPLL* procedure. This section provides technical details of the reduction, Theorem 5.1. In addition to the properties stated in Section 2, the following ones hold:

For all $\phi_1, \phi_2 \in PropForm$ and $\psi_1, \psi_2, \psi_3 \in OrdPropForm$,

$$(\phi_1 \wedge \phi_2) \prec 1 \equiv \phi_1 \prec 1 \vee \phi_2 \prec 1, \tag{59}$$

$$(\phi_1 \vee \phi_2) \prec 1 \equiv \phi_1 \prec 1 \wedge \phi_2 \prec 1, \tag{60}$$

$$\psi_1 \vee \psi_2 \wedge \psi_3 = (\psi_1 \vee \psi_2) \wedge (\psi_1 \vee \psi_3). \tag{61}$$

Theorem 5.1 (Reduction Theorem)**.** *Let $\phi \in PropForm$. There exists $T_\phi \subseteq_{\mathcal{F}} OrdCl$ such that T_ϕ is unsatisfiable if and only if ϕ is a tautology.*

Proof. By Lemma 3.1, there exists a conjunctive normal form ψ such that $\psi \equiv \phi$ and we distinguish tree cases:

either $\psi = 0$ or $\psi = 1$ or $\psi = \bigwedge_{i \leq n} \bigvee_{j \leq m_i} l^i_j$, l^i_j are literals.

Case 1: $\phi \equiv \psi = 0$. Then ϕ is not a tautology and $T_\phi = \emptyset \subseteq_{\mathcal{F}} OrdCl$ is satisfiable. So, the claim holds.

Case 2: $\phi \equiv \psi = 1$. Then ϕ is a tautology and $T_\phi = \{\square\} \subseteq_{\mathcal{F}} OrdCl$ is unsatisfiable. So, the claim holds.

Case 3: $\phi \equiv \psi = \bigwedge_{i \leq n} \bigvee_{j \leq m_i} l^i_j$, l^i_j are literals. Then

$$\phi \text{ is a tautology if and only if } \phi \prec 1 \in OrdPropForm \text{ is unsatisfiable;} \tag{62}$$

$$\phi \prec 1 \equiv \psi \prec 1 = (\bigwedge_{i \leq n} \bigvee_{j \leq m_i} l^i_j) \prec 1 \overset{(59)}{\underset{(60)}{\equiv}} \bigvee_{i \leq n} \bigwedge_{j \leq m_i} l^i_j \prec 1. \tag{63}$$

$$\text{For all } i \leq n \text{ and } j \leq m_i \text{, there exists a conjunction of disjunctions of order literals } \delta^i_j \in OrdPropForm \text{ such that } \delta^i_j \text{ is equisatisfiable to } l^i_j \prec 1. \tag{64}$$

The proof is by definition. We get five cases for l^i_j:
Case 3.1: $l^i_j = a$, $a \in PropAtom$. Then $\delta^i_j = a \prec 1$.
Case 3.2: $l^i_j = a \rightarrow 0$, $a \in PropAtom$. Then $\delta^i_j = 0 \prec a$.

Case 3.3: $l^i_j = a \rightarrow b$, $a \in PropAtom$, $b \in PropAtom$. Then $\delta^i_j = b \prec a \wedge b \prec 1$. Case 3.4: $l^i_j = (a \rightarrow 0) \rightarrow 0$, $a \in PropAtom$. Then $\delta^i_j = a \eqcirc 0$. Case 3.5: $l^i_j = (a \rightarrow b) \rightarrow b$, $a \in PropAtom$, $b \in PropAtom$. Then $\delta^i_j = (a \prec b \vee a \eqcirc b) \wedge b \prec 1$. So, the claim (64) holds. By (64) and (63),

$$\bigvee_{i \leq n} \bigwedge_{j \leq m_i} \delta^i_j \text{ is equisatisfiable to } \bigvee_{i \leq n} \bigwedge_{j \leq m_i} l^i_j \prec 1 \text{ and } \phi \prec 1. \tag{65}$$

Hence, there exists $\varphi \in OrdPropForm$ such that

$$\varphi = \bigwedge_{r \leq v} \bigvee_{s \leq u_r} \kappa^r_s \stackrel{((61))}{\equiv} \bigvee_{i \leq n} \bigwedge_{j \leq m_i} \delta^i_j \tag{66}$$

where κ^i_j are order literals. By (66) and (65), there exists $T_\phi \subseteq_{\mathcal{F}} OrdCl$ such that

$$T_\phi = \{\bigvee_{s \leq u_r} \kappa^r_s \,|\, r \leq v\} \text{ is equisatisfiable to } \varphi, \quad \bigvee_{i \leq n} \bigwedge_{j \leq m_i} \delta^i_j, \text{ and } \phi \prec 1. \tag{67}$$

We close that T_ϕ is unsatisfiable $\stackrel{(67)}{\Longleftrightarrow}$ $\phi \prec 1$ is unsatisfiable $\stackrel{(62)}{\Longleftrightarrow}$ ϕ is a tautology.

□

Let $\phi = (a \rightarrow b) \rightarrow ((b \rightarrow c) \rightarrow (a \rightarrow c)) \in PropForm$, $a, b, c \in PropAtom$. Using Theorem 5.1, we show that ϕ is a tautology. At first, using Lemma 3.1, we translate ϕ to an equivalent *CNF*

$$\psi = ((a \rightarrow b) \rightarrow b \vee c \vee (b \rightarrow c) \rightarrow c \vee a \rightarrow c) \wedge (b \rightarrow c \vee c \vee (b \rightarrow c) \rightarrow c \vee a \rightarrow c),$$

cf. the example after Lemma 3.1. Then, using (59) and (60), $\psi \prec 1$ is equivalent to

$$\begin{aligned} \xi = ((a \rightarrow b) \rightarrow b) \prec 1 \wedge c \prec 1 \wedge \\ ((b \rightarrow c) \rightarrow c) \prec 1 \wedge (a \rightarrow c) \prec 1 \vee \\ (b \rightarrow c) \prec 1 \wedge c \prec 1 \wedge \\ ((b \rightarrow c) \rightarrow c) \prec 1 \wedge (a \rightarrow c) \prec 1. \end{aligned}$$

Hence, using (64) and (67), ξ is equisatisfiable to $T_\phi \subseteq_{\mathcal{F}} OrdCl$ where

$$\begin{aligned} T_\phi = \{ & a \prec b \vee a \eqcirc b \vee c \prec b, && [1] \\ & a \prec b \vee a \eqcirc b \vee c \prec 1, && [2] \\ & a \prec b \vee a \eqcirc b \vee b \prec c \vee b \eqcirc c, && [3] \\ & a \prec b \vee a \eqcirc b \vee c \prec a, && [4] \\ & b \prec 1 \vee c \prec b, && [5] \\ & b \prec 1 \vee c \prec 1, && [6] \\ & b \prec 1 \vee b \eqcirc c, && [7] \\ & b \prec 1 \vee c \prec a, && [8] \\ & c \prec 1 \vee c \prec b, && [9] \\ & c \prec 1, && [10] \\ & c \prec 1 \vee b \eqcirc c, && [11] \\ & c \prec 1 \vee c \prec a, && [12] \\ & b \prec c \vee b \eqcirc c \vee c \prec b, && [13] \\ & b \prec c \vee b \eqcirc c \vee c \prec 1, && [14] \\ & b \prec c \vee b \eqcirc c, && [15] \\ & b \prec c \vee b \eqcirc c \vee c \prec a, && [16] \\ & c \prec a \vee c \prec b, && [17] \\ & c \prec a \vee c \prec 1, && [18] \\ & c \prec a \vee b \eqcirc c, && [19] \\ & c \prec a \}. && [20] \end{aligned}$$

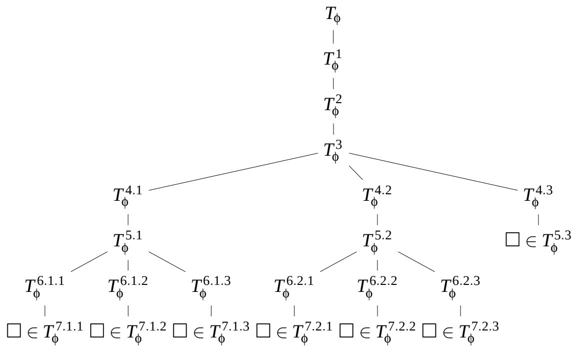

Figure 1: Closed tree $Tree_\phi$.

Finally, using the *DPLL* procedure rules, we can construct a closed tree $Tree_\phi$ with the root T_ϕ, outlined in Figure 1.

We close that T_ϕ is unsatisfiable, and by Theorem 5.1, ϕ is a tautology.

6 CONCLUSIONS

We have investigated the satisfiability and validity problems of a formula in the propositional Gödel logic. The satisfiability problem has been solved via the translation of a formula to an equivalent *CNF* one, containing literals of the forms a, $a \rightarrow b$, or $(a \rightarrow b) \rightarrow b$. A *CNF* formula has further been translated to an equisatisfiable finite order clausal theory, which consists of order clauses with order literals of the forms $a \eqcirc b$ or $a \prec b$. $\eqcirc$ and $\prec$ are interpreted by the equality and strict linear order on $[0,1]$, respectively. The trichotomy on order literals: either $a \prec b$

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

or $a = b$ or $b \prec a$, has naturally led to a variant of the *DPLL* procedure with a trichotomy branching rule, which is refutation sound and complete. We have reduced the validity problem of a formula to the unsatisfiability of a finite order clausal theory.

REFERENCES

Aguzzoli, S. and Ciabattoni, A. Finiteness of infinite-valued Łukasiewicz logic. *Journal of Logic, Language and Information*, 9:5-29, 2000.

Anderson, R. and Bledsoe, W. W. A linear format for resolution with merging and a new technique for establishing completeness. *Journal of the ACM*, 17(3):525-534, 1970.

Baaz, M., Fermüller, C. G. and Ciabattoni, A. Herbrand's theorem for prenex Gödel logic and its consequences for theorem proving. *Proceedings of the LPAR conference, LNCS vol. 2250, Springer-Verlag*, 201-215, 2001.

Bachmair, L. and Ganzinger, H. Rewrite-based equational theorem proving with selection and simplification. *Journal of Logic and Computation*, 4(3):217-247, 1994.

Bachmair, L. and Ganzinger, H. Ordered chaining calculi for first-order theories of transitive relations. *Journal of the ACM*, 45(6):1007-1049, 1998.

Beckert, B., Hähnle, R. and Manyà, F. The SAT problem of signed CNF formulas. In *Labelled Deduction*, Basin, D., D'Agostino, M., Gabbay, D., Matthews, S. and Viganó, L., eds., *Applied Logic Series, vol. 17, Kluwer Academic Publishers*, 61-82, 2000.

Biere, A., Heule, M., van Maaren, H. and Walsh, T., eds. Handbook of Satisfiability. *Frontiers in Artificial Intelligence and Applications Series, vol. 185, IOS Press*, 2009.

Boy de la Tour, T. An optimality result for clause form translation. *Journal of Symbolic Computation*, 14(4):283-301, 1992.

Davis, M. and Putnam, H. A computing procedure for quantification theory. *Communications of the ACM*, 7:201-215, 1960.

Davis, M., Logemann, G. and Loveland, D. A machine program for theorem-proving. *Communications of the ACM*, 5:394-397, 1962.

Dixon, H. E., Ginsberg, M. L., Luks, E. M. and Parkes, A. J. Generalizing Boolean satisfiability II: Theory. *Journal of Artificial Intelligence Research*, 22:481-534, 2004.

Dixon, H. E., Ginsberg, M. L. and Parkes, A. J. Generalizing Boolean satisfiability I: Background and survey of existing work. *Journal of Artificial Intelligence Research*, 21:193-243, 2004.

Gomes, C. P., Kautz, H., Sabharwal, A. and Selman, B. Satisfiability solvers. In *Handbook of Knowledge Representation*, Harmelen, F. v., Lifschitz, V. and Porter, B., eds., *Elsevier Science Publishers*, Part I, Chap. 3, 2007.

Guller, D. On the refutational completeness of signed binary resolution and hyperresolution. *Fuzzy Sets and Systems*, 160(8):1162-1176, 2009.

Hähnle, R. Many-valued logic and mixed integer programming. *Annals of Mathematics and Artificial Intelligence*, 12(3,4):231-264, 1994.

Hähnle, R. Short conjunctive normal forms in finitely-valued logics. *Journal of Logic and Computation*, 4(6):905-927, 1994.

Hähnle, R. Exploiting data dependencies in many-valued logics. *Journal of Applied Non-Classical Logics*, 6(1):49-69, 1996.

Hähnle, R. Proof theory of many-valued logic - linear optimization - logic design: Connections and interactions. *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, 1(3):107-119, 1997.

Kautz, H. and Selman, B. The state of SAT. *Discrete Applied Mathematics*, 155(12):1514-1524, 2007.

Manyà, F., Béjar, R. and Escalada-Imaz, G. The satisfiability problem in regular CNF-formulas. *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, 2(3):116-123, 1998.

Mundici, D. Satisfiability in many-valued sentential logic is NP-complete. *Theoretical Computer Science*, 52:145-153, 1987.

Nonnengart, A., Rock, G. and Weidenbach, Ch. On generating small clause normal forms. *Proceedings of the CADE conference, LNAI vol. 1421, Springer-Verlag*, 397-411, 1998.

Plaisted, D. A. and Greenbaum, S. A structure-preserving clause form translation. *Journal of Symbolic Computation*, 2(3):293-304, 1986.

Sheridan, D. The optimality of a fast CNF conversion and its use with SAT. *Online Proceedings of International Conference on the Theory and Applications of Satisfiability Testing, www.satisfiability.org/SAT04/programme/114.pdf*, 2004.