Noname manuscript No.
(will be inserted by the editor)

Complexity Issues in Basic Logic

S. Aguzzoli[1], B. Gerla[2], Z. Haniková[3]

[1] DSI, University of Milano, Italy, aguzzoli@dsi.unimi.it (corresponding author)
[2] DMI, University of Salerno, Italy, bgerla@unisa.it
[3] ICS, Academy of Sciences of the Czech Republic, zuzana@cs.cas.cz

The date of receipt and acceptance will be inserted by the editor

Abstract We survey complexity results concerning a family of propositional many-valued logics. In particular, we shall address satisfiability and tautologousness problems for Hájek's Basic Logic BL and for several of its schematic extensions. We shall review complexity bounds obtained from functional representation results, as well as techniques for dealing with non-trivial ordinal sums of continuous t-norms.

1 Introduction

In this paper we give an overview of known propositional complexity results for some important logics. In particular, we consider the logics given by each of t-norm algebras; these include BL, Łukasiewicz logic, Gödel logic, Product logic, as well as some logics with the Δ connective.

Research into this area has been undertaken in the last three decades, starting with a pioneering work [30] of Mundici in 1987 regarding NP-completeness of Łukasiewicz logic and flourishing during the nineties, with some problems still left open.

Our aim is to go over main results on the topic and methods which have been developed and used for analysis of propositional complexity of t-norm logics. We do not go into details of proofs, referring the reader to the ample literature available.

In [30] Mundici first introduced a functional representation argument, inspired by McNaughton's theorem [26], to state complexity of Łukasiewicz logic. This technique has been further applied to other many-valued logics, Gödel and Product among them [4]. The functional representation theorems are deeply linked to the study of free algebras.

Results on the coNP-completeness of BL and logics given by single t-norm algebras have been obtained by a technique developed in the paper [7], where an algorithm capable of working with a t-norm algebra which is a non-trivial ordinal sum is first outlined. It has been further developed in [24].

1.1 Basic notions

Unless specified otherwise, our propositional language is that of BL; the alphabet has countably many propositional variables, basic connectives 0, & , →, and defined connectives ∧, ∨, ¬, 1, ≡. Formulas are built up as usual. Connectives are defined from &, →, and 0 as follows:

$$\begin{array}{rll} \neg\varphi & \text{is} & \varphi \to 0 \\ \varphi \wedge \psi & \text{is} & \varphi \,\&\, (\varphi \to \psi) \\ \varphi \vee \psi & \text{is} & ((\varphi \to \psi) \to \psi) \wedge ((\psi \to \varphi) \to \varphi) \\ \varphi \equiv \psi & \text{is} & (\varphi \to \psi) \,\&\, (\psi \to \varphi) \\ 1 & \text{is} & 0 \to 0 \end{array}$$

Let $\mathbf{L} = \langle L, \wedge, \vee, *, \Rightarrow, 0, 1\rangle$ be a BL-algebra. An $\mathbf{L}$-evaluation of propositional variables is a mapping e, assigning to each propositional variable p an element of L. Each evaluation of propositional variables extends uniquely to propositional formulas as follows:

$$\begin{aligned} e(0) &= 0 \\ e(\varphi \,\&\, \psi) &= e(\varphi) * e(\psi) \\ e(\varphi \to \psi) &= e(\varphi) \Rightarrow e(\psi) \end{aligned}$$

The t-algebra given by a continuous t-norm $*$ is the BL-algebra $[0,1]_* = \langle [0,1], *, \Rightarrow, 0\rangle$, where $*$ is a continuous t-norm and $\Rightarrow$ is its residuum. The terms 't-algebra', 'standard algebra' and 't-norm algebra' are all used in literature and have the same meaning.

For Łukasiewicz, Gödel and Product t-norms we shall use the notation $[0,1]_{\text{Ł}}$, $[0,1]_{\text{G}}$ and $[0,1]_{\Pi}$ respectively, to denote the standard algebras. Further, we shall use the symbols BL, Ł, G, Π to denote propositional Basic, Łukasiewicz, Gödel and Product logics.

There is a 1-1 correspondence between formulas of propositional BL and terms in the language of BL-algebras; the term results from the formula by replacing all connectives with the operations which evaluate them (and by replacing propositional variables by individual variables), and is called the *associated term* of the formula. Conversely, if operation symbols in a term are replaced by propositional connectives (and individual variables in the term are replaced by propositional variables), the result is referred to as the *associated formula* of the term.

Definition 1 *(i) A formula φ is a* 1*-tautology of a BL-algebra* $\mathbf{L}$ *(an* $\mathbf{L}$*-tautology) iff* $e(\varphi) = 1$ *for all* $\mathbf{L}$*-evaluations* e *(i. e., iff its associated term always has the value* 1*).*
(ii) A formula is a t-tautology iff it is a 1*-tautology of each t-algebra.*

1.2 SAT and TAUT problems in a many-valued setting

To be able to compare results on classical and many-valued logic, let us suppose that the (basic) connectives of formulas of the classical propositional logic are &, $\rightarrow$ and 0.

Then we define

$$\mathrm{TAUT} = \{\varphi; \forall e(e(\varphi) = 1)\}$$
$$\mathrm{SAT} = \{\varphi; \exists e(e(\varphi) = 1)\}$$

where φ runs over all formulas in the basic language, and e over evaluations in the classical two-valued Boolean algebra on $\{0, 1\}$.

Obviously TAUT $= \{\neg\varphi; \varphi \notin \mathrm{SAT}\}$ and (consequently, considering the semantic equivalence of φ and $\neg\neg\varphi$,) SAT $= \{\neg\varphi; \varphi \notin \mathrm{TAUT}\}$. Here $\neg\varphi$ is as usual, $\varphi \rightarrow 0$. Thus TAUT is reducible to the complement of SAT and vice versa.

It is well known (Cook's theorem) that the set SAT is NP-complete, thus TAUT is coNP-complete.

With a many-valued logic $\mathcal{L}$, or a $\mathcal{L}$-algebra $\mathbf{A}$, one might wonder about the definition of the SAT and TAUT problems, as the classical dichotomy is no longer at hand. For a fixed semantics given by an algebra $\mathbf{A}$, it makes sense to distinguish the following sets of formulas (cf. [21]). In all cases φ stands for propositional formulas in the BL-language and $e_{\mathbf{A}}$ runs over evaluations in $\mathbf{A}$.

$$\mathrm{TAUT}_1^{\mathbf{A}} = \{\varphi : \forall e_{\mathbf{A}}(e_{\mathbf{A}}(\varphi) = 1)\}$$
$$\mathrm{TAUT}_{\mathrm{pos}}^{\mathbf{A}} = \{\varphi : \forall e_{\mathbf{A}}(e_{\mathbf{A}}(\varphi) > 0)\}$$
$$\mathrm{SAT}_1^{\mathbf{A}} = \{\varphi : \exists e_{\mathbf{A}}(e_{\mathbf{A}}(\varphi) = 1)\}$$
$$\mathrm{SAT}_{\mathrm{pos}}^{\mathbf{A}} = \{\varphi : \exists e_{\mathbf{A}}(e_{\mathbf{A}}(\varphi) > 0)\}$$

These sets are referred to as 1-tautologies, positive tautologies, 1-satisfiable formulas and positively satisfiable formulas of $\mathbf{A}$.

For a class K of algebras of the same type, one may generalize (as suggested in [7] for the SAT problems):

$$\mathrm{TAUT}_1^{K} = \{\varphi : \forall \mathbf{A} \in K\, \forall e_{\mathbf{A}}(e_{\mathbf{A}}(\varphi) = 1)\}$$
$$\mathrm{TAUT}_{\mathrm{pos}}^{K} = \{\varphi : \forall \mathbf{A} \in K\, \forall e_{\mathbf{A}}(e_{\mathbf{A}}(\varphi) > 0)\}$$
$$\mathrm{SAT}_1^{K} = \{\varphi : \exists \mathbf{A} \in K\, \exists e_{\mathbf{A}}(e_{\mathbf{A}}(\varphi) = 1)\}$$
$$\mathrm{SAT}_{\mathrm{pos}}^{K} = \{\varphi : \exists \mathbf{A} \in K\, \exists e_{\mathbf{A}}(e_{\mathbf{A}}(\varphi) > 0)\}$$

Unlike in classical logic, for a many-valued semantics there need not be a simple relationship between its TAUT and SAT problems. The 1-tautologies will probably be of most interest (so we often omit the index 1); however, the other sets present themselves to be investigated as well.

In this paper we give an overview of known propositional complexity results for some important logics/algebras. Note that the logics Ł, G, *Π* and BL all enjoy single standard completeness, (i.e., they are complete with respect to a unique t-algebra, see Section 2.1), thus it is unnecessary to distinguish provability and 1-tautologousness in the standard algebra.

2 Tools and methods

The result of Mundici regarding NP-completeness of Łukasiewicz logic (see [30]) is obtained by a direct inspection of *truth tables* of Łukasiewicz formulas that are a particular class of $[0, 1]$-valued functions over a suitable power of $[0, 1]$. In [5], Mundici's method has been extended to Gödel and Product logic, but complexity results for these logics have been obtained first in a different way and can be found in [6] and [21].

In this section we survey basic tools used to establish complexity results, i.e., functional representation arguments, reductions to mixed integer programming tools and analysis of t-norm logic based on the decomposition in ordinal sums. We also give all necessary definitions.

2.1 Free algebras and functional representation

From universal algebra [8] we know that if an algebra $\mathcal{A}$ generates a variety $\mathcal{V}$, then the free algebra in the subvariety of the n-generated algebras in $\mathcal{V}$ is the subalgebra of $\mathcal{A}^{\mathcal{A}^n}$ generated by the projections.

Chang's completeness theorem implies that all standard MV-algebras (i.e. those having $[0, 1]$ as their support), are isomorphic to $[0, 1]_{\mathrm{Ł}}$. In this case we have $\mathrm{TAUT}_1^{[0,1]_{\mathrm{Ł}}} = \mathrm{TAUT}_1^{MV}$ and we shall denote this set by $\mathrm{TAUT}_1^{\mathrm{Ł}}$.

All schematic extensions of BL whose corresponding subvariety of BL-algebras is singly generated by a standard algebra $[0, 1]_*$ — we call this property *single standard completeness* — are amenable to an analysis which can yield a concrete representation of the free algebras over finitely many generators.

This property is not always granted: consider for instance BL plus the precancellation axiom $\neg\neg\vartheta \to (((\varphi\&\vartheta) \to (\psi\&\vartheta)) \to (\varphi \to \psi))$. The only standard algebras in the variety determined by this schematic extension are $[0,1]_{\text{Ł}}$ and $[0,1]_{\Pi}$, but clearly neither of these singly generates the variety.

If $\mathcal{L}$ is a single standard complete logic whose corresponding variety is singly generated by a standard algebra $\mathbf{A}$, we shall set $\mathrm{TAUT}_1^{\mathcal{L}} = \mathrm{TAUT}_1^{\mathbf{A}}$. The same kind of notation will be applied to the other problems in our classification. Moreover, an $\mathbf{A}$-evaluation $e_{\mathbf{A}}$ may be denoted by $e_{\mathcal{L}}$.

Single standard completeness theorems also hold for Basic Logic and its extensions Gödel and Product logic.

This means that free algebras in varieties of BL, Gödel and Product algebras, as well as in the variety of MV-algebras, are subalgebras of $[0,1]^{[0,1]^n}$ generated by projections, in particular, free algebras are algebras of real-valued functions. Each function belonging to the free algebra $\mathbf{F}$ of a variety $\mathbb{V}_{\mathcal{L}}$ corresponding to the logic $\mathcal{L}$ can be thought of as a class of logically equivalent formulas of $\mathcal{L}$, exactly as every function in $\{0,1\}^{\{0,1\}^n}$ is the equivalence class of a boolean formula.

Functional representation theorems tackle the problem of functional incompleteness for t-norm based logics (where the single standard completeness holds): In contrast with propositional boolean logic where each function $f : \{0,1\}^n \to \{0,1\}$ is definable with a formula in n variables, t-norm based logics are *functionally incomplete*, that is not every function $f : [0,1]^n \to [0,1]$ is definable. A functional representation theorem for logic $\mathcal{L}$ describes precisely which functions are definable with formulas of $\mathcal{L}$.

Let $\mathcal{L}$ be a schematic extension of BL for which standard completeness holds with respect to a t-norm $*$. We consider for any formula φ of $\mathcal{L}$ with n variables, the function $\varphi^{\mathcal{L}} : [0,1]^n \to [0,1]$ given by the associated term in the algebra $[0,1]_*$.
Evaluations in the standard algebra are canonically identified with points of $[0,1]^n$: if $e_{\mathcal{L}}(x_i) = t_i$ for all $i \in \{1,\ldots,n\}$, then $e_{\mathcal{L}}(\varphi) = \varphi^{\mathcal{L}}(t_1,\ldots,t_n)$. We shall freely speak of points and functions determined by formulas in contexts related to free algebras and functional representation, and of $\mathcal{L}$-evaluations when dealing directly with standard algebras, chains, ordinal sums.

Functional representation of formulas of a many-valued logic $\mathcal{L}$ can give insights in understanding complexity problems of $\mathcal{L}$. Indeed if a formula φ with $var(\varphi) \subseteq \{p_1,\ldots,p_n\}$ is not a tautology then there exists a *critical point* $\mathbf{x} \in [0,1]^n$ such that $\varphi^{\mathcal{L}}(\mathbf{x}) < 1$. We are interested in showing that a critical point $\mathbf{x}$ can be guessed in such a way that the calculus of $\varphi^{\mathcal{L}}(\mathbf{x})$ is polynomial in the length $||\varphi||$ of the input formula φ. We define $||\varphi||$ as the total number of occurrences of connectives in φ and $|\varphi|$ as the total number of occurrences of propositional variables in φ. Note that the actual length of the string representing a formula given any reasonable coding of propositional variables, is obviously $\geq ||\varphi|| \geq |\varphi| - 1$. Note also that the total number of (occurrences) of subformulas of φ is $||\varphi|| + |\varphi|$.

As we shall see later, we will be mainly involved in finding critical points that are *rational*, i.e., are tuples $\mathbf{x} = (b_1/d_1, b_2/d_2, \ldots, b_n/d_n) \in ([0,1] \cap \mathbb{Q})^n$, for all $b_i \geq 0$ and $d_i > 0$ being integers. Forming the lowest common denominator d of the components b_i/d_i we write $\mathbf{x} = (c_1/d, \ldots, c_n/d)$, for suitably computed integers c_i, and we call the integer $d > 0$ the *denominator* of $\mathbf{x}$ ($\mathrm{den}(\mathbf{x}) = d$) and the $(n+1)$-tuple of integers $(c_1, \ldots, c_n, d)$ *homogeneous coordinates* of $\mathbf{x}$.

Since we are interested in bounding denominators, we define $\mathrm{size}(\mathbf{x}) = \log_2 \mathrm{den}(\mathbf{x})$, this quantity being proportional to the number of bits needed to store the value of each component x_i (the choice of the basis of the logarithm is immaterial, as long as it guarantees a coding of numeric information which is not unary).

Hence, the algorithm *guess and check G&C* for the complement of $\mathrm{TAUT}_1^{\mathcal{L}}$ consists of these two steps:

1. Guess a critical point $\mathbf{x}$ of $\varphi^{\mathcal{L}}$.
2. Check that $\varphi^{\mathcal{L}}(\mathbf{x}) < 1$.

To establish NP-containment of this algorithm it is only needed to show that there exists a one variable polynomial p such that, for all non-tautological φ, there always exists a critical point $\mathbf{x}$ with $\mathrm{size}(\mathbf{x}) \leq p(||\varphi||)$. As a matter of fact, in this case each one of the polynomially many (w.r.t. $||\varphi||$) operations needed to compute $\varphi^{\mathcal{L}}(\mathbf{v})$ can be performed in polynomial time (as long as each one of these operations is not more complex than adding or comparing two numbers of $\mathrm{size}(\mathbf{x})$ bits each), and hence the whole checking step is polynomial time computable by some deterministic Turing machine.

All the problems considered in our classification can be formulated by adapting in the obvious ways the nondeterministic algorithm given above, hence the matter is reduced to the study of sizes of critical points.

2.2 Mixed integer programming

In order to prove coNP containment of $\mathrm{TAUT}^{\mathcal{L}}$ it is possible to shorten the analysis of the piecewise linear geometry of functions in the appropriate free algebras in at least two ways. Having already achieved the coNP containment for a logic at least as combinatorially complex as Łukasiewicz logic (for instance, Gödel logic is too simple, in this context), one can try to work with embeddings between logics and other algebraic constructions like ordinal sum representations in such a way to reduce the problem to the already known results.

The second approach consists in shortening the analysis of functions: instead of working out explicitly the size of critical points for a formula φ, this method reduces to the construction of a set of systems of linear constraints inductively built from subformulas of φ. Each one of these systems can be thought of as a linear program whose size is polynomial in φ. If at least one of the systems in this set is solvable, that is, it has a solution $\mathbf{x}$, then $\varphi^{\mathcal{L}}(\mathbf{x}) < 1$ and then $\varphi \notin \mathrm{TAUT}_1^{\mathcal{L}}$. Checking solvability of linear programs is a problem known to be in P, but here we have to deal in general with exponentially many such programs, as each occurrence of a binary connective generally introduces two mutually exclusive sets of linear constraints in each partially built system, thus splitting each one of them into two new systems. Guessing one of the solvable programs gives us a coNP algorithm for $\mathrm{TAUT}_1^{\mathcal{L}}$. Interestingly, Hähnle in [18,19] (see also the paper [20]), introduces techniques from disjunctive linear programming in order to combine all the exponentially many systems into one mixed integer program of polynomial size with respect to φ. Mixed integer programming requires that there is at least one unknown among those of the system which is obliged to take on only integer (or boolean) values. Mixed integer programming has the same complexity as Integer programming, that is, both problems are NP-complete.

In order to avoid splitting of sets of linear constraints, Hähnle introduces in the system additional boolean valued control variables, which allow to encode both sets of linear constraints introduced by an occurrence of a connective in one set of linear constraints: each one of the two original sets of constraints is recovered in solvable solutions, which one of the two depending on the value the boolean control variable assumes.

Hähnle technique [18], when applicable, allows to export semantic *sets-as-signs* tableaux calculi for finite-valued logics to the infinite-valued setting, thus yielding for a given *t*-norm logic a decision procedure which having in input the formula φ uses tableau rules to generate the appropriate Mixed integer program to be checked for solvability.

2.3 Finite-valued logics

For each integer $n < 0$, let $S_n \subset [0,1]$ be the finite set of truth-values defined as $S_n = \{0, 1/n, 2/n, \ldots, (n-1)/n, 1\}$.

Whenever S_n is closed under a t-norm $*$ and its residuum $\Rightarrow$ we can consider the $(n+1)$-valued algebra $\mathbf{S}_n^* = \langle S_n, * \upharpoonright S_n, \Rightarrow \upharpoonright S_n, 0\rangle$. Then, if a *t*-norm logic $\mathcal{L}$ has single standard completeness we can consider the $(n+1)$-valued logic $\mathcal{L}_n$ having $\mathbf{S}_n^*$ as algebraic semantics. We call $\mathcal{L}_n$ a finite-valued logic approximating $\mathcal{L}$ or a finite-valued $\mathcal{L}$ logic.

One can immediately check the existence of all finite-valued logics $\{\mathrm{G}_i\}_{i>0}$ approximating G and $\{\text{Ł}_i\}_{i>0}$ approximating Ł, while the only finite-valued Product logic is $\Pi_1 = \text{Ł}_1 = \mathrm{G}_1$, that is, Boolean propositional logic.

Applying the non-deterministic algorithm scheme *G&C* introduced before, we trivially see that for a $(n+1)$-valued logic $\mathcal{L}_n$ we have $\mathrm{size}(\mathbf{x}) \le n$ for each critical point $\mathbf{x}$ for a formula φ, and then all $\mathrm{SAT}^{\mathcal{L}_n}$ problems are in NP and all $\mathrm{TAUT}^{\mathcal{L}_n}$ problems are in coNP.

2.4 Ordinal sums

According to the decomposition theorem for continuous t-norms [29], each t-algebra is an ordinal sum of isomorphic copies of $[0,1]_{\text{Ł}}$, $[0,1]_{\mathrm{G}}$ and $[0,1]_{\Pi}$ (the so-called Ł-, G-, and Π-segments). Contact points between intervals in an ordinal sum are habitually called *cutpoints*.

A generalization of this theorem for saturated BL-chains can be found in [22] and [10]. (See also [1] for a *hoop* decomposition of BL-chains).

Regarding decomposition of t-algebras into the three types of segments, we take each maximal interval of idempotents as a single Gödel segment; thus $[0,1]_{\mathrm{G}}$ is a sum consisting of one element and $[0,1]_{\text{Ł}\oplus\mathrm{G}\oplus\Pi}$ is a t-algebra consisting of three segments (Łukasiewicz, Gödel and Product). We denote the three types of segments by Ł, G, and Π.

An important result from [1] states that a t-algebra generates the whole variety BL iff its first segment is Ł and the sum contains infinitely many Ł-segments.

3 Complexity results for Łukasiewicz, Gödel and Product logics

In next sections we shall describe the proofs that firstly appeared for complexity results of logics $\mathcal{L} \in \{\Pi, \mathrm{G}, \text{Ł}\}$.

In Section 3.1 we summarize Mundici's proof: required definitions and results will be given in Sections 3.1.1 and 3.1.2. Then in Section 3.2 results for Gödel and Product logic will be given, taken from [17], [21], [6].

3.1 Complexity of Łukasiewicz logic

3.1.1 McNaughton's Representation Theorem for Łukasiewicz infinite-valued logic From Section 2.1 we know that the free MV-algebra over n generators is $\mathbf{M}_n = \langle M_n, \oplus, \neg, 0\rangle$, where M_n is the smallest set of functions $f : [0,1]^n \to [0,1]$ containing all projections $(x_1, \ldots, x_n) \mapsto x_i$ and closed under the operations $\oplus$ and $\neg$, that are defined by applying the operations of $[0,1]_{\text{Ł}}$ pointwise, that is, for all $\mathbf{x} \in [0,1]^n$:

$$(f \oplus g)(\mathbf{x}) = f(\mathbf{x}) \oplus g(\mathbf{x}) = \min(1, f(\mathbf{x}) + g(\mathbf{x})),$$

$$(\neg f)(\mathbf{x}) = \neg f(\mathbf{x}) = 1 - f(\mathbf{x}).$$

Note that, following tradition, when dealing with MV-algebras and Łukasiewicz logic, we prefer to choose as primitive the $t-conorm$ operation $\oplus$ instead of the t-norm $x \odot y = \max(0, x+y-1)$. Involutiveness of negation and DeMorgan duality between $\oplus$ and $\odot$ guarantees that each one of the following subsets of $\{\odot, \oplus, \to, \neg, 0\}$ is sufficient to derive all the remaining MV-operations: $\{\oplus, \neg\}$,$\{\odot, \neg\}$,$\{\to, \neg\}$.

McNaughton [26] was the first to characterize the set M_n in more concrete terms:

Theorem 1 *For all integers $n \geq 0$, M_n is the set of all* continuous *functions $f : [0,1]^n \to [0,1]$ which are* piecewise linear*, each of the finitely many pieces* having integer coefficients*: that is, there exist linear polynomials $p_1, \ldots, p_u$, each p_i being of the form $p_i(x_1, \ldots, x_n) = \sum_{j=1}^{n} a_{ij}x_j + b_i$ with all a_{ij} and b_i in $\mathbb{Z}$, such that, for all $(t_1, \ldots, t_n) \in [0,1]^n$ there exists $k \in \{1, \ldots, u\}$ for which $f(t_1, \ldots, t_n) = p_k(t_1, \ldots, t_n)$.*

Members of the free MV-algebra $\mathbf{M}_n$ are then called *McNaughton functions*. Showing that each function in M_n satisfies McNaughton's requirements is an easy induction on the structure of formulas. The other side of the proof is more difficult, and in McNaughton's original proof involves at a certain stage a reductio ad absurdum that destroys constructivity. The first constructive proof is due to Mundici [31] and is based on the machinery of Schauder hats (see also [9]).

In both proofs the first step is showing that each *clipped* hyperplane, that is, a function of the form $\max(0, \min(1, \sum_{j=1}^{n} a_j x_j + b))$ belongs to M_n. This is easily seen by proving by induction on $\sum_{j=1}^{n} |a_j|$ the following equation, due to Rose and Rosser [33]:

$$\max(0, \min(1, g(\mathbf{x}) + x_1)) =$$

$$(\max(0, \min(1, g(\mathbf{x}))) \oplus x_1) \odot \max(0, \min(1, g(\mathbf{x}) + 1))$$

where, without loss of generality, we assume $|a_1| = \max_j |a_j|$ and $g(x_1, \ldots, x_n) = (a_1 - 1)x_1 + \sum_{j=2}^{n} a_j x_j + b$.

The difficult step is proving that each McNaughton function f is obtained by assembling clipped hyperplanes via Łukasiewicz connectives. Mundici showed that f is indeed an $\oplus$-sum of a suitably chosen family of finitely many Schauder hats.

To understand Mundici's construction we need to introduce some notions of polyhedral geometry (see [14] for further background). A *polyhedral complex* is a set of polyhedra that contains all their faces and such that any two polyhedra intersect in a common face (possibly $\emptyset$).

An n-simplex is the convex hull of $(n+1)$ many affinely independent points (its vertices). The convex hull of a finite set $S \subseteq \mathbb{R}^k$ is the set $\{\sum_{\mathbf{p} \in S} \lambda_{\mathbf{p}} \mathbf{p} \mid 0 \leq \lambda_{\mathbf{p}} \leq 1 \text{ for each } \mathbf{p} \in S, \sum_{\mathbf{p} \in S} \lambda_{\mathbf{p}} = 1\}$. An n-simplex S with rational vertices in $[0,1]^n$ is *unimodular* when the $(n+1) \times (n+1)$ integral matrix M_S, whose rows are the homogeneous coordinates of the vertices of S, is such that $|\det(M_S)| = 1$.

For any piecewise linear function $f : [0,1]^n \to [0,1]$, with each piece having integer coefficients, the unit hypercube $[0,1]^n$ can be partitioned with a suitable *polyhedral complex* C_f (in the sense that $\bigcup\{P \in C_f\} = [0,1]^n$) of finite cardinality, such that the function f is linear over every polyhedron in C_f and all vertices of polyhedra in C_f are rational. Any such polyhedral complex is said to be *linearly adequate* to f. An application of Minkowski's convex body Theorem from geometry of numbers allows to refine C_f by finitely many polyhedral subdivisions until one gets a *unimodular partition* U_f of $[0,1]^n$, which is a polyhedral complex linearly adequate to f and such that all its full dimensional polyhedra are unimodular simplexes.

The family $SH(U)$ of *Schauder hats* determined by a unimodular partition U of $[0,1]^n$ is the set of all functions H such that:

- there is one vertex $\mathbf{v}$ of U such that $H(\mathbf{v}) = 1/\mathrm{den}(\mathbf{v})$.
- $H(\mathbf{w}) = 0$ for all vertices $\mathbf{w} \neq \mathbf{v}$ of U.
- H is linear over each simplex of U.

Note that the pair $(U, \mathbf{v})$ uniquely determines one Schauder hat. We may denote H by $H_{\mathbf{v}}$ stressing the dependence on the *apex* $\mathbf{v}$ while considering U fixed by the context. Observe that U is linearly adequate to each hat in $SH(U)$. The restriction of the Schauder hat H to each simplex S of U having $\mathbf{v}$ among its vertices coincides with the restriction to S of a hyperplane Q. Here unimodularity of S guarantees that Q has integer coefficients, hence it coincides over S with a clipped hyperplane Q'. Having collected all formulas which represent such clipped hyperplanes, a general –although somewhat complex– argument of lattice-ordered abelian group theory allows to write down each hat H as $H = (\bigwedge_{i \in I} \bigvee_{j \in J(I)} \psi_{ij})^{\text{Ł}}$, for suitable finite index sets $I, J(I)$, where each ψ_{ij} represents a clipped hyperplane (see [9] for the detailed construction). Moreover, $SH(U)$ constitutes a base for all McNaughton functions for which U is linearly adequate, that is, given such a function f, it is possible to identify a *multiplicity* map $\mu_f : SH(U) \to \mathbb{N}$ such that $f = \varphi^{\text{Ł}}$ for

$$\varphi = \bigoplus_{H \in SH(U)} \mu_f(H)H =$$

$$\bigoplus_{\mathbf{v} \text{ is a vertex of } U} \mu_f(H_{\mathbf{v}})H_{\mathbf{v}}$$

where by $m\vartheta$ we mean the $\oplus$-sum $\vartheta \oplus \vartheta \oplus \cdots \oplus \vartheta$ with ϑ taken m times and $0 \leq \mu_f(H_{\mathbf{v}}) \leq \mathrm{den}(\mathbf{v})$. This concludes Mundici's argument.

It is worth noting that Mundici's construction requires in input, as specification of the function f, only the values f takes on a sufficiently large set of points (see Section 3.1 for numerical estimates). If in addition we are explicitly given an effective description of a polyhedral complex $\{P_k\}_{k \in K}$ linearly adequate to f, we can skip construction of Schauder hats and write down f as $f = (\bigwedge_{r\in R} \bigvee_{s\in S(R)} \psi_{\iota(r,s)})^{\text{Ł}}$ where each $\psi_{\iota(r,s)} = \psi_k$ is a clipped hyperplane coinciding over some P_k with f (see [2] for further details).

Knowing $\{P_k\}_{k\in K}$ allows proving McNaughton's Theorem via a *patching* technique (see [32] for a similar construction). Each McNaughton function f is expressed as $f = (\bigvee_{k\in K} \lambda_k \vartheta_k \odot \psi_k)^{\text{Ł}}$, where ψ_k represents the clipped hyperplane coinciding over P_k with f, while $\vartheta_k^{\text{Ł}}$ is 1 over P_k and strictly < 1 elsewhere (such functions exist for all P_k), and quickly going to 0; finally $0 < \lambda_k \in \mathbb{Z}$ is a suitably large integer having the effect to increase sufficiently the slopes with which $\vartheta_k^{\text{Ł}}$ goes to 0.

3.1.2 $\text{SAT}^{\text{Ł}}_{\text{pos}}$ *is NP-complete and derived results* McNaughton's Theorem 1 is the main tool used by Mundici in [30] to prove that $\text{SAT}^{\text{Ł}}_{\text{pos}}$ is in NP.

From linear algebra (or operational research) we know that the maximum (and the minimum) of a linear function l defined over a polyhedron P is attained in a vertex of P. Passing to McNaughton piecewise linear geometry, we have that relative maxima (and relative minima) of a McNaughton function $g \in M_n$ are attained in some vertices of members of a polyhedral complex C_φ linearly adequate to g. Hence the critical point $\mathbf{x}$ such that $\varphi^{\text{Ł}}(\mathbf{x}) > 0$ to be guessed by our candidate NP-algorithm *G&C* of Section 2.1 is among the vertices (of polyhedra) of C_φ.

While polyhedral complexes linearly adequate to formulas are not unique, in [30] it is shown that we may always choose C_φ as to have nice bounds on the size of vertices of its polyhedra (we recall that each such vertex $\mathbf{v}$ is rational).

More precisely, C_φ can be always chosen in such a way that any vertex $\mathbf{w}$ of any polyhedron in C_φ arises as the solution of a system of n many linear equations, each one of them either of the form $p_i(\mathbf{x}) = 0$ or $p_i(\mathbf{x}) = 1$ or $p_i(\mathbf{x}) = p_j(\mathbf{x})$ or $x_h = 0$ or $x_h = 1$, for $p_1, \ldots, p_u$ be the finitely many linear polynomials given by Theorem 1 for the McNaughton function $\varphi^{\text{Ł}}$. Actually, such a C_φ can be built by induction on subformulas of φ as the *roughest* polyhedral complex linearly adequate to each subformula of φ. We shall be more explicit in this construction in Section 5.4. Let us display each p_i as $p_i(\mathbf{x}) = \sum_{j=1}^n a_{ij}x_j + b_i$. An easy induction on the structure of φ shows that $|a_{ij}| \le |\varphi|$ for every coefficient a_{ij}. (Actually, in [3] is proved that $|a_{ij}| \le$ the number of occurrences of x_i in φ, see also Section 5.4). Then, for each $l \in \{1, \ldots, n\}$, the lth linear equation in the system can be written down as $c_{l1}x_1 + \cdots + c_{ln}x_n = d_l$ with each $|c_{li}| \le 2|\varphi|$. Forming the integral $n \times n$ matrix $M_{\mathbf{w}} = \{c_{li}\}_{l,i=1}^n$ and the integral vector $\mathbf{d} = (d_1, \ldots, d_n)$ associated with the system, we have $\mathbf{w} = M_{\mathbf{w}}^{-1}\mathbf{d}$ and hence $\text{den}(\mathbf{w}) \le |\det(M_{\mathbf{w}})|$. Using Hadamard's inequality we may conclude $\text{den}(\mathbf{w}) \le (4n|\varphi|^2)^{n/2} < 2^{(2|\varphi|)^2}$ and thus the guessed vertex $\mathbf{v}$ is such that $\text{size}(\mathbf{v}) \le 2|\varphi|^2$ and $\text{SAT}^{\text{Ł}}_{\text{pos}} \in \text{NP}$.

The polynomial reduction τ from satisfiability in Boolean logic to $\text{SAT}^{\text{Ł}}_{\text{pos}}$ is defined for all formulas ψ of Boolean logic in the variables $p_1, \ldots, p_n$ as the map

$$\tau(\psi) = (p_1 \vee \neg p_1)^2 \odot \cdots \odot (p_n \vee \neg p_n)^2 \odot \psi^2.$$

where ϑ^m denotes the formula $\vartheta \odot \vartheta \odot \cdots \odot \vartheta$ for ϑ occurring m times. It can be verified [21] that φ is satisfiable in Boolean logic if and only if $\tau(\varphi) \in \text{SAT}^{\text{Ł}}_{\text{pos}}$. Then $\text{SAT}^{\text{Ł}}_{\text{pos}}$ is NP-hard. We can now state the following theorem:

Theorem 2 $\text{SAT}^{\text{Ł}}_{\text{pos}}$ *is* NP-complete.

A more detailed analysis on the size of entries of $M_{\mathbf{w}}$ allows to have a smaller bound on the size of guessed vertices. We shall sketch the technique used for the improved bounds in Section 5.4.

Consider now, for $\theta \in [0, 1]$, the set

$$\text{SAT}^{\text{Ł}}_{\theta} = \{\varphi : \forall \mathbf{x} \in [0, 1]^n, \varphi^{\text{Ł}}(\mathbf{x}) \ge \theta\}.$$

NP-containment of the $\text{SAT}^{\text{Ł}}_{\theta}$ problem can be dealt with in the same way for each θ (again, it is sufficient to guess a vertex $\mathbf{v}$ such that $\varphi^{\text{Ł}}(\mathbf{v}) \ge \theta$). NP-hardness of $\text{SAT}^{\text{Ł}}_{\theta}$ can be showed by reducing Boolean satisfiability to $\text{SAT}^{\text{Ł}}_{\theta}$ via the previously defined map τ. Finally, $\varphi \in \text{TAUT}^{\text{Ł}}_1$ if and only if $\neg\varphi \notin \text{SAT}^{\text{Ł}}_{\text{pos}}$, and thus $\text{TAUT}^{\text{Ł}}_1$ is in coNP.

Theorem 3 $\text{SAT}^{\text{Ł}}_{\theta}$ *is* NP-complete *for each* $\theta \in [0, 1] \cap \mathbb{Q}$. *in particular* $\text{SAT}^{\text{Ł}}_1$ *is* NP-complete, $\text{TAUT}^{\text{Ł}}_1$ *and* $\text{TAUT}^{\text{Ł}}_{\text{pos}}$ *are* coNP-complete.

3.2 Complexity of Gödel and Product logic

In this section we consider complexity results for Gödel and Product logic, as appeared in [6] and later in [21]. In section 5.2 we shall investigate the same problem by using methods based on functional representation of logics.

We start by giving some results from [21].

Theorem 4 $\text{SAT}^{\text{G}}_1 = \text{SAT}^{\text{G}}_{\text{pos}} = \text{SAT}^{\Pi}_1 = \text{SAT}^{\Pi}_{\text{pos}}$ *and all these sets are equal to (classical)* SAT.

The proof is easy: as obviously $\text{SAT} \subseteq \text{SAT}^{\{\text{G},\Pi\}}_1 \subseteq \text{SAT}^{\{\text{G},\Pi\}}_{\text{pos}}$, we only need to show that if φ is positively satisfiable in G (Π), then it is classically satisfiable. Having any evaluation in G (Π) which gives a positive value on φ, to obtain a classically satisfying evaluation replace all nonzero values by 1.

Theorem 5 $\text{TAUT}^{\text{G}}_{\text{pos}} = \text{TAUT}^{\Pi}_{\text{pos}} = \text{TAUT}$

The method of proof is analogous to the above (see [21]).

We now concentrate on the 1-tautologies of either of the two logics.

Let φ be a formula with variables $p_1, \ldots, p_n$ and let $I \subseteq \{p_1, \ldots, p_n\}$. We define the formula φ^I be induction on complexity:

$$\begin{aligned} p_i^I = 0 \text{ if } p_i \in I; \quad p_i^I &= p_i \text{ otherwise} \\ 0^I = 0; \quad 1^I &= 1; \\ (\psi_1 \& \psi_2)^I &= 0 \text{ if at least one of } \psi_1, \psi_2 \text{ is } 0; \\ (\psi_1 \& \psi_2)^I &= \psi_1^I \& \psi_2^I \text{ otherwise}; \\ (0 \to \varphi)^I &= 1; \\ (\varphi \to 0)^I &= 0 \text{ if } \varphi^I \text{ is not } 0; \\ (\varphi \to \psi)^I &= \varphi^I \to \psi^I \text{ otherwise.} \end{aligned}$$

Then it is possible to prove that the formula φ^I is either equal to 0 or to 1 or it does not contain any 0.

In other words, in fixing I we set some variables equal to 0 and by φ^I we denote the translation of φ obtained by setting all variables in I equal to 0 and then making a sort of simplification (for example instead of writing $0 \to \psi$ we write directly 1).

Let $\mathcal{C}$ be G or Π. If e is an evaluation such that $e(p_i) = 0$ if and only if $p_i \in I$, then it is easy to prove that $e_{\mathcal{C}}(\varphi) = e_{\mathcal{C}}(\varphi^I)$ and $e_{\mathcal{C}}(\varphi^I) = 0$ if and only if φ^I is the formula 0. Further, if there exists $e_{\mathcal{C}}$ such that $e_{\mathcal{C}}(\varphi) > 0$, then by setting $I = \{p_i \mid e(p_i) = 0\}$ we have then that φ^I is not equal to 0.

Vice-versa, if there exists a set I such that φ^I is different from 0 then the evaluation e such that $e(p_i) = 0$ if $p_i \in I$ and $e(p_i) = 1$ otherwise is such that $e_{\mathcal{C}}(\varphi) > 0$.

Results on $TAUT_1^{\Pi}$ are based on the embeddability of Łukasiewicz logic in Product logic, as depicted in the following lemma [21].

Lemma 1 *For each a such that $0 < a < 1$, the MV-algebra $[0,1]_{\text{Ł}}$ is isomorphic to $([a,1], *_a, \to_a, a, 1)$ where*

$$\begin{aligned} x *_a y &= \max(a, x \cdot y) \\ x \to_a y &= x \to_{.} y \text{ (Product implication).} \end{aligned}$$

Let φ be a formula not in TAUT_1^{Π}. Hence there is an evaluation e such that $e_{\Pi}(\varphi) < 1$ and so there exists a subset I of propositional variables of φ and an evaluation e' such that for every subformula ψ of φ^I ($\psi \preceq \varphi^I$), $e'_{\Pi}(\varphi^I) < 1$ and $e'(\psi) > 0$ (for the fact that in φ^I no 0's appear). Hence, by Lemma 1, by choosing the element a smaller then $e'_{\Pi}(\psi)$ for every $\psi \preceq \varphi^I$, we have $\varphi \notin \mathrm{TAUT}_1^{\Pi}$ if and only if there exists I such that

$$\bigwedge_{\psi \preceq \varphi^I} \psi \wedge \neg(\varphi^I) \in SAT_{\text{pos}}^{\text{Ł}}. \qquad (1)$$

This reduces polynomially the complement of $TAUT_1^{\Pi}$ to $SAT_{\text{pos}}^{\text{Ł}}$, hence by Theorem 2, we have

Theorem 6 *$TAUT_1^{\Pi}$ is coNP complete.*

In the formula (1) we have replaced the infimum on variables not belonging to I, which appears in the original proof in [6] or in [21], by the infimum taken on all subformulas of φ^I, thus fixing a small bug: to exemplify why the original formula does not always work, we consider $\varphi = \neg\neg p \to ((p \to (p\&p)) \to p)$ that is a tautology of Product logic. For $I = \emptyset$, we have $\varphi^I = (p \to (p\&p)) \to p$ and

$$\psi := \bigwedge_{v \notin I} v \wedge \neg(\varphi^I) = p \wedge \neg((p \to (p\&p)) \to p) \in SAT_{\text{pos}}^{\text{Ł}}$$

since $\psi^{\text{Ł}}(1/3) = 1/3$.

Concerning Gödel logic, in the paper [17] Gödel shows that there is an infinite descending chain of logics between the classical propositional logic A and the intuitionistic propositional logic H, and that H (and hence Gödel infinite-valued logic) cannot be viewed as a finite-valued logic. This chain is

$$A = \mathrm{G}_2 \supset \mathrm{G}_3 \supset \ldots \supset \mathrm{G}_n \supset \ldots \supset \mathrm{G}_\infty \supset H,$$

where each G_n is the set of formulas valid in every n-elements linearly ordered Gödel algebras and it is axiomatized by axioms of Gödel logic plus $n+1$ variables axiom

$$F_{n+1} = \bigvee_{1 \le i < k \le n+1} p_i \leftrightarrow p_k.$$

Formula F_{n+1} is valid only when variables are interpreted in sets with less than $n+1$ elements. Then F_{n+1} is an example of a formula with $n+1$ variables that is not a tautology of Gödel logic and of G_n, but is a tautology of $\mathrm{G}_1, \ldots, \mathrm{G}_{n-1}$.

We further investigate 1-tautologies of Gödel logic as in [21]. If φ is not a 1-tautology, there exists an evaluation v of propositional variables $p_1, \ldots, p_n$ of φ such that $v(\varphi) < 1$. Consider the set $D = \{v(p_1), \ldots, v(p_n)\} \subseteq [0,1]$. By checking truth tables of Gödel connectives, it is easy to see that for every subformula ψ of φ, $v(\psi) \in D \cup \{0,1\}$. Hence we can consider, without loss of generality, that

$$D = S_{n+1} = \left\{0, \frac{1}{n+1}, \ldots, \frac{n}{n+1}, 1\right\}.$$

Then consider the following procedure:

1. Guess a subset $\{x_1, \ldots, x_n\}$ of $[0,1]$.
2. Put $v(p_i) = x_i$ for every $i = 1, \ldots, n$ and calculate $v(\varphi)$.
3. Check if $v(\varphi) < 1$.

The above procedure assures that the $\mathrm{TAUT}_1^{\mathrm{G}}$ problem is in coNP, since we can suppose that $\{x_1, \ldots, x_n\} \subseteq D$.

4 Complexity of Basic logic and individual t-norm logics

4.1 Basic logic and t-tautologies

We use the notation TAUT_1^* for the set of t-tautologies. In the propositional case, BL-provable formulas coincide with TAUT_1^* thanks to standard completeness theorem of [10]. Here we reproduce the results of [7], showing that TAUT_1^* is coNP-complete.

Lemma 2 *A propositional formula is a t-tautology iff it is a 1-tautology of all t-algebras which are finite ordinal sum of Ł-segments.*

The proof in [7] is based on Lemma 1.

Theorem 7 *The set of t-tautologies is coNP-complete.*

The coNP-hardness of the t-tautology problem is established by reduction of $\mathrm{TAUT}_1^{Ł}$, namely: a formula $\varphi(p_1, \ldots, p_n)$ is in $\mathrm{TAUT}_1^{Ł}$ iff $\varphi(\neg p_1, \ldots, \neg p_n)$ is a tautology of each finite sum of Ł-segments.

To show the coNP-containment of the problem, a nondeterministic algorithm, running in time polynomial in the length of φ and accepting the complement of TAUT_1^*, is presented. Here we give its slight modification.

The following lemma comes from [25] and strengthens a similar result in [7], where the bound was $4k + 1$.

Lemma 3 *If $\varphi(p_1, \ldots, p_k)$ is not a t-tautology, then it is not a 1-tautology of some t-algebra which is a sum of (at most) $k+1$ Ł-segments.*

Consequently, to find out whether a formula $\varphi(p_1, \ldots, p_k)$ is a t-tautology, it is enough to consider its 1-tautologousness in a single t-algebra, namely, a $k+1$-potent sum of Ł-segments.

Preliminaries and notation. For a propositional formula φ, let m denote the number of subformulas in φ. Further, k denotes the number of propositional variables in φ.

`// algorithm T-TAUT accepting`
`// the complement of` TAUT_1^*

`input:` φ

`begin`

Set $n = k + 1$ and let $\mathbf{A}$ be the t-algebra which is an ordinal sum of n Ł-segments.

`nameSubformulas()` Fix an arbitrary enumeration of all subformulas of φ; introduce variables $x_1 \ldots, x_m$, and assign the variable x_i to the subformula φ_i of φ.

`cutpointVariables()` Introduce variables $z_0, \ldots, z_n$ for the cutpoints of $\mathbf{A}$, enumerated so that z_i represents the i-th cutpoint of $\mathbf{A}$ in the ordering of reals; z_0 is intended for 0 and z_n is intended for 1.

Set $V = \{z_0, \ldots, z_n\} \cup \{x_1, \ldots, x_m\}$.

`guessOrder()` Guess a linear ordering $\preceq$ of variables in V: write the variables down in a sequence, for each consequent pair a and b decide whether $a \approx b$ or $a \prec b$.

`checkOrder()` Check that $\preceq$ satisfies basic natural conditions: first, that it preserves the ordering of the z-variables, i. e., if $i < j$, then $z_i \prec z_j$. Second, any variable assigned to the constant 0 must be $\approx$-equal to z_0. Third, the variable assigned to φ must be strictly smaller than z_n (so that the evaluation whose existence the algorithm verifies assigns value less than 1 to φ).

We say that variables x_j s. t. $z_i \preceq x_j \preceq z_{i+1}$ *belong to* i or *are in* i.

`checkExternal()` Check external soundness of $\preceq$: for φ_i, φ_j subformulas of φ ($1 \le i, j \le m$),

– if $\varphi_i \,\&\, \varphi_j$ is a subformula φ_k of φ for some $k \in \{1, \ldots, m\}$ and, for some $l \in \{1, \ldots, m\}$, we have $x_i \preceq z_l \preceq x_j$, then $x_k \approx x_i$;

– if $\varphi_i \to \varphi_j$ is a subformula φ_k of φ for some $k \in \{1, \ldots, m\}$ and $x_i \preceq x_j$, then $x_k \approx z_n$;

– if $\varphi_i \to \varphi_j$ is a subformula φ_k of φ for some $k \in \{1, \ldots, m\}$ and, for some $l \in \{1, \ldots, m\}$, we have $x_j \prec z_l \preceq x_i$, then $x_k \approx x_j$.

`checkInternal()` Check internal soundness of $\preceq$ for each interval $[z_i, z_{i+1}]$, $i = 0, \ldots, n-1$ in $\preceq$. Consider variables in i. Construct a system $\mathcal{S}_i$ of equations and inequalities; $\mathcal{S}_i$ is initially empty. For each subformula φ_l which is $\varphi_j \& \varphi_k$ (or $\varphi_j \to \varphi_k$), if x_j and x_k are in i, check x_l is also in i and put equation $x_j * x_k = x_l$ (or $x_j \Rightarrow x_k = x_l$ respectively) into $\mathcal{S}_i$. Further, put all equations and inequalities defined by $\preceq$ for the variables in i into $\mathcal{S}_i$. Check whether the system $\mathcal{S}_i$ has a solution in the i-th segment of $\mathbf{A}$, such that z_i and z_{i+1} evaluate to the cutpoints delimiting the i-th segment of $\mathbf{A}$.

`end`

Obviously the system $\mathcal{S}_i$ in the `checkInternal()` step is solvable in the i-th segment of the algebra iff it is solvable in the standard Łukasiewicz algebra.

The ordering $\preceq$ contains clusters of mutually $\approx$-equivalent variables; choose one representative from each cluster and replace with it all the occurrences in $\mathcal{S}_i$ of all the other variables in the cluster; accordingly in $\preceq$ we consider only the strict ordering of these representatives. The total number of representatives is $j \le m + 2$. We retain the notation $\mathcal{S}_i$ for this modified system: $\mathcal{S}_i$ has j variables, a strict ordering of length j, and number of equations at most m.

The following lemma is Hähnle's result [19], obtained by a reduction to a particular Mixed Integer Programming problem (cf. 2.2); see also [7] for comments.

Lemma 4 *The problem of solvability of each $\mathcal{S}_i$ in the standard Łukasiewicz algebra is in NP.*

In [7] it is also shown that SAT_1^* coincide with the classical SAT and thus are NP-complete.

4.2 Individual t-norm logics

In this section we establish coNP-completeness of the set $\mathrm{TAUT}_1^{[0,1]_*}$ for any t-algebra $[0,1]_*$. The result comes from [24]. We note in passing that, as shown in [25], for each t-algebra $[0,1]_*$ the problems SAT_1^* and $\mathrm{SAT}_{\mathrm{pos}}^*$ are NP-complete and the problem $\mathrm{TAUT}_{\mathrm{pos}}^*$ is coNP-complete.

Moreover, it is not difficult to show that for an arbitrary t-algebra, TAUT_1^* is coNP-hard: the algebra either has a first segment Ł, or has the strict negation. In case of a first segment Ł, the set $\mathrm{TAUT}^{[0,1]_{\text{Ł}}}$ is polynomially reducible to $\mathrm{TAUT}^{[0,1]_*}$ as in [7], prefixing a negation to each occurrence of a propositional variable. If $[0,1]_*$ has the strict negation, the set TAUT of classical tautologies can be polynomially reduced to $\mathrm{TAUT}^{[0,1]_*}$ as described in [21], prefixing a double negation to each occurrence of a propositional variable. The latter reduction can be also used for *classes* of algebras with the strict negation (for example SBL): forming a translation $\varphi^{\neg\neg}$ as above, we have $\varphi \in \mathrm{TAUT}$ iff $\varphi^{\neg\neg}$ is a 1-tautology in all standard SBL-algebras.

It remains to show the coNP-containment of the set $\mathrm{TAUT}_1^{[0,1]_*}$ for any $*$. The algorithm presented is a slight modification the one in [24]. Generally, both are based on the algorithm accepting non-t-tautologies, presented in [7] (and its slight modification here in 4.1).

Theorem 8 *Let* $\mathbf{A}$ *be an arbitrary t-algebra. Then* $\mathrm{TAUT}^{\mathbf{A}}$ *is in NP.*

First we give an algorithm for finite ordinal sums of Ł, G, and Π-segments. Then we modify it for infinite sums which generate the variety SBL. Finally we modify it for other infinite sums.

4.2.1 Finite sums We present an algorithm `FIN` (for finite sums) which decides whether an input formula φ has an evaluation in $\mathbf{A}$ s. t. $e_{\mathbf{A}}(\varphi) < 1$; if so, the output is 'yes', otherwise it is 'no'. So the set accepted by `FIN` is the complement of $\mathrm{TAUT}^{\mathbf{A}}$. Throughout a t-algebra $\mathbf{A}$ which is a finite ordered sum is fixed and the type and cardinality of the sum is used as a built-in information.

We claim the nondeterministic algorithm `FIN` works in polynomial time w. r. t. the length of φ. The only step requiring inspection is the last, `checkInternal()` step; we will argue that this step is an NP subroutine for G- and Π-segments (we already know this for Ł-segments from 4.1).

Notation. Let n denote the number of segments in $\mathbf{A}$. For a propositional formula φ, let m denote the number of subformulas in φ.

```
// algorithm FIN for finite sum A
input: φ
begin
nameSubformulas()
cutpointVariables()
guessOrder()
checkOrder()
checkExternal()
checkInternal()
end
```

We discuss the polynomial nature of the `checkInternal()` step, considering the situation in the i-th segment, for i fixed. The step defines a system $\mathcal{S}_i$ of equations of type $x * y = z$ and of type $x \Rightarrow y = z$, and of equations and inequalities imposed by $\preceq$ (see 4.1). [7] presents an NP routine which checks solvability of $\mathcal{S}_i$ in $[0,1]_{\text{Ł}}$, so it remains to consider $[0,1]_{\mathrm{G}}$ and for $[0,1]_{\Pi}$. It is shown in [24] that the solvability of the system $\mathcal{S}_i$ in $[0,1]_{\mathrm{G}}$ can be checked in time linear in m. Namely, we consider the variables themselves as a Gödel chain and check that the equations are sound w. r. t. the ordering.

For the Product t-algebra we use the following lemma, coming also from [24].

Lemma 5 *The system* $\mathcal{S}_i$ *is solvable in* $[0,1]_{\Pi}$ *iff it is solvable in a t-algebra* $[0,1]_{\text{Ł}\oplus\text{Ł}}$*, in such a way that* x_{i_0} *is evaluated by* $0_{\text{Ł}\oplus\text{Ł}}$*,* $x_{i_{j-1}}$ *is evaluated by* $1_{\text{Ł}\oplus\text{Ł}}$ *and* $x_{i_1}, \ldots, x_{i_{j-2}}$ *are evaluated in* $(1/2, 1)$*, where* $1/2$ *is the non-extremal cutpoint.*

To check solvability in the Product t-algebra, we first eliminate all equations involving y_{i0}, it being possible to check the soundness of any such equation "externally". Then we consider the remaining equations and strict inequalities in Ł, introducing a new inequality $0 < y_{i1}$, and check solvability of this system of equations and inequalities using the algorithm for solvability in $[0,1]_{\text{Ł}}$.

We conclude that there is a nondeterministic algorithm, running in time polynomial in m, checking the solvability of the system $\mathcal{S}_i$ in $[0,1]_{\Pi}$.

Finally, it is obvious from the construction of the algorithm that the output is 'yes' (on at least one branch) iff the formula φ has a counterexample evaluation in A, i. e., is not an A-tautology. Thus the algorithm solves the problem and the set of A-tautologies is in coNP.

4.2.2 SBL-generic t-algebras The propositional logic SBL, 'S' for 'strict', is a schematic extension of BL with an axiom or axioms stating that the negation $\neg$ is *strict*: in any BL-chain—and particularly under the standard semantics—its truth function is two-valued and assigns 1 to argument 0, while 0 is the value for all non-zero arguments. One possible axiom that can be added to BL to obtain this logic is the axiom $\Pi 2$ $(\varphi \wedge \neg\varphi \rightarrow 0)$.

SBL has been investigated in [12]. Since it is a schematic extension of BL, it enjoys completeness w. r. t. SBL-chains; moreover, standard completeness for SBL has been proved in [10].

The following is an easy consequence of Lemma 1. The converse implication also holds and has been proved in [13]. We will not need it here.

Lemma 6 *Let* $\mathbf{A} = \bigoplus_{i\in I} A_i$ *be an SBL t-algebra. If there are infinitely many i's s. t.* $\mathbf{A}_i$ *is an Ł-segment, then* $\mathbf{A}$ *generates the variety SBL.*

Proof Assume $\mathbf{A}$ is as above, φ is not an SBL-tautology, and let $\mathbf{B}$ be an SBL t-algebra in which φ does not hold. We may assume $\mathbf{B}$ is a finite sum of Ł-segments and Π-segments only (starting with a Π). Then the counterexample evaluation on $\mathbf{B}$ can be locally embedded in $\mathbf{A}$, mapping 0 to 0, for the semi-open initial Π segment of $\mathbf{B}$ using any Ł-segment of $\mathbf{A}$, for each of the following Ł-segments of $\mathbf{B}$ using an Ł-segment of $\mathbf{A}$, and for each of the following Π-segments of $\mathbf{B}$ using two (not necessarily adjacent) Ł-segments of $\mathbf{A}$, all in increasing order w. r. t. the ordering of the segments in $[0, 1]$. This gives a counterexample in $\mathbf{A}$.

Now we show that SBL is in coNP, thus solving the complexity problem for all t-algebras which generate the variety SBL.

If φ is not an SBL-tautology, then it has a counterexample in a standard finite sum whose first element is not an Ł, and with a suitably small number of segments. In fact, it is an easy modification of Lemma 3 that $\varphi(p_1, \dots, p_k)$ is not an SBL-tautology, iff it is not a 1-tautology of a t-algebra of type $\Pi \oplus Ł \oplus \cdots \oplus Ł$, with k Ł-segments.

Thus we may modify the algorithm `T-TAUT` to work with an ordinal sum $\mathbf{A}$, determined by the input formula $\varphi(p_1, \dots, p_k)$: its number of segments is $n = k + 1$, the first segment is Π, the other segments are Ł.

```
// algorithm for SBL
input: φ
begin
//the algorithm works with A, n = k + 1
nameSubformulas()
cutpointVariables()
guessOrder()
checkOrder()
checkExternal()
checkInternal()
end
```

It is obvious that this modification is an algorithm running in polynomial time and accepting SBL counterexamples, so the propositional tautology problem for the logic SBL is in coNP.

4.2.3 Other infinite sums Putting aside those t-algebras which generate either BL (characterized in [1]) or SBL, we are left with algebras which have only finitely many (possibly no) Ł-segments. We will show that this property allows for a finite description of the tautologies of each such t-algebra.

First a lemma showing that the tautologies of any infinite sum without Ł-segments coincide with the tautologies of an infinite sum of Π-segments only (all of which also have the same set of tautologies).

Lemma 7 *Let* $[0, 1]_*$ *and* $[0, 1]_{*'}$ *be two standard algebras which are infinite sums without Ł-segments. Then* $\mathrm{TAUT}^{[0,1]_*} = \mathrm{TAUT}^{[0,1]_{*'}}$.

Proof Observe that there are infinitely many Π-segments in both sums and the presence/absence of G-segments inbetween Π-segments and the ordering type do not matter. Thus each of the sums embeds counterexamples from the other one.

This can be generalized: let $[0, 1]_*$ be an arbitrary t-algebra with two non-extremal idempotents $0 < c_1 < c_2 < 1$. Define two new t-algebras by substituting a copy of two arbitrary infinite sums without Ł-segments into the interval $[c_1, c_2]$. Then the resulting two t-algebras will have the same sets of tautologies.

For each t-algebra which is an infinite sum, we now encode the sets of its tautologies with a finite string. The idea that this can be done comes from [24]; it has been further developed in [13], where the concept of *canonical algebra* is introduced, and shown that for each t-algebra there is a canonical one with the same set of tautologies.

Definition 2 (canonical t-algebra) *A t-algebra is canonical iff it is an infinite sum of Ł-segments only, or a sum of a Π-segment followed by infinite sum of Ł-segments only, or a finite sum of segments of type Ł, G, Π and $\infty\Pi$ (standing for an infinite sum of Π-segments only), where each G-segment is not preceded or followed by another G, and each segment $\infty\Pi$ is not preceded or followed by G, Π or another $\infty\Pi$.*

It follows from the above lemma and discussion that canonical algebras cover all possible subvarieties of BL generated by a single t-algebra; moreover, [13] shows that distinct canonical algebras generate distinct varieties. The strings in the alphabet $\{$Ł, G, Π, $\infty\Pi\}$ give a nice finite-string representation of the subvarieties (or, equivalently, sets of propositional tautologies) distinct from both BL and SBL.

Now we show that the 1-tautologies of any canonical t-algebra $\mathbf{A}$ which is an infinite sum with finitely many Ł-segments are in coNP. Use $l(A)$ for the length of the finite string in the alphabet $\{$Ł,G,Π,$\infty\Pi\}$ representing $\mathbf{A}$.

As before, for a propositional formula φ the symbol m denotes the number of subformulas in φ, and k stands for the number of propositional variables in φ.

```
// algorithm INF for infinite sum A
input: φ
begin
```

`guessCardinality()` Pick at random a natural n, $0 < n \leq k+1$.

`guessLayout()` Assign to each $i = 1, \ldots, n$ one of the symbols Ł, G, Π, signifying the type of the i-th segment of the sum.

We use the term 'constructed sum' and the symbol $\mathbf{C}$ to denote this finite sum.

`checkEmbedding()` Check whether the constructed sum is $1-1$ embeddable into $\mathbf{A}$ (as a sequence of symbols into a sequence of symbols, see the following lemma), in such a way that a potential initial Ł of the constructed sum is mapped to an initial Ł in $\mathbf{A}$.

```
// from now on the algorithm works with C
nameSubformulas()
cutpointVariables()
guessOrder()
checkOrder()
checkExternal()
checkInternal()
end
```

Lemma 8 *The embeddability of the constructed sum* $\mathbf{C}$ *into* $\mathbf{A}$ *can be checked by an algorithm running in polynomial time w. r. t. the length* n *of* $\mathbf{C}$.

Proof The algorithm works in two stages: first it guesses, for each symbol in $\mathbf{C}$, an index into the representation of $\mathbf{A}$, i. e., a natural number in $[0, l(A)-1]$. If the guess is sound, this should be a 1-1 embedding of $\mathbf{C}$ into $\mathbf{A}$. Note that the information guessed is polynomial since $l(A)$ is constant. Then it performs a verification of whether the guess was sound: an (initial) Ł-segment in $\mathbf{C}$ may only map to an (initial) Ł-segment in $\mathbf{A}$; a G-segment in $\mathbf{C}$ may only map to a G-segment in $\mathbf{A}$, and a Π-segment in $\mathbf{C}$ may map either to a Π segment or an $\infty\Pi$-segment in $\mathbf{A}$. The indices must be nondecreasing (w. r. t. the sequence $\mathbf{C}$), no two Ł-segments in $\mathbf{C}$ may have the same index, no two G-segments in $\mathbf{C}$ may have the same index, and two Π-segments may have the same index iff it specifies an $\infty\Pi$-segment of $\mathbf{A}$. Obviously this check is polynomial in n since it is sufficient to consider the indices of each two neighbouring segments in turn.

Note that checking the embeddability of $\mathbf{C}$ into $\mathbf{A}$ can also be performed in deterministic polynomial time, by easily adapting the algorithm for checking embeddability of a string in another one.

If there is a counterexample evaluation in $\mathbf{A}$, then there is a finite subsum of $\mathbf{A}$ harbouring it. We know it is enough to search all finite subsums up to length $k+1$. The algorithm works with each such subsum as finite sum and works in exactly the same way as in the case for finite sums.

5 Finite countermodels and reductions to finite-valued logics

Finite-valued reduction for an infinitely valued logic $\mathcal{L}$ allows to reduce the problem of tautologousness of a generic formula φ in $\mathcal{L}$ to the tautologousness problem of φ in a finite set of finitely valued logics. From another point of view, this amounts to limiting, in a uniform way, the search space for finite countermodels for any formula φ that is not a tautology.

In this section we shall survey results of [2],[3],[4] for Łukasiewicz, Gödel and Product logic, by describing functional representation of Gödel and Product logic first.

Throughout this section φ will be a formula with variables among $p_1, \ldots, p_n$.

5.1 Functional representation for Gödel logic

In this section we shall describe truth tables of Gödel formulas, thus giving a functional description of free Gödel algebras. Results of this section have been adapted from results of [15].

We introduce a subdivision of $[0,1]^n$ taking into account the possible orders between components of each point $\mathbf{x} = (x_1, \ldots, x_n) \in [0,1]^n$.

Consider the following equivalence relations between points of $[0,1]^n$:

- $(x_1, \ldots, x_n) \equiv_1 (y_1, \ldots, y_n)$ if, for every $i \in \{1, \ldots, n\}$, $x_i = 0$ if and only if $y_i = 0$.
- $(x_1, \ldots, x_n) \equiv_2 (y_1, \ldots, y_n)$ if, for every $i, j \in \{1, \ldots, n\}$,
  - $(x_1, \ldots, x_n) \equiv_1 (y_1, \ldots, y_n)$,
  - $x_i < x_j$ if and only if $y_i < y_j$,
  - $x_i = x_j$ if and only if $y_i = y_j$.

Then let $C^1(\mathbf{x})$ and $C^2(\mathbf{x})$ be the class of equivalence of $\mathbf{x}$ with respect to equivalence relations $\equiv_1$ and $\equiv_2$ respectively.

Then the formula

$$\varphi_{C^1(\mathbf{x})} = \bigwedge_{x_i=0} \neg p_i \wedge \neg\neg \bigwedge_{x_j \neq 0} p_j$$

is a Gödel formula such that the function $\varphi^{\mathrm{G}}_{C^1(\mathbf{x})}$ is the characteristic function of $C^1(\mathbf{x})$. On the contrary, it is not possible to describe Gödel functions that are characteristic functions of regions $C^2(\mathbf{x})$.

Let $\mathcal{C}$ be the set of all $C^2(\mathbf{x})$, with $\mathbf{x} \in [0,1]^n$. Note that $\mathcal{C}$ is not a partition of $[0,1]^n$. The functional representation for Gödel logic is given by the following theorem:

Theorem 9 *The restriction of* φ^{G} *on every* $C \in \mathcal{C}$ *is either equal to* 0 *or to* 1 *or is a projection. Vice-versa, if* f *is a function such that, restricted to an element of* $\mathcal{C}$ *is either equal to* 0 *or to* 1 *or to a projection, then there exists a Gödel formula* φ *such that* f *is the truth table of* φ.

For details on proof see [15]. It is worth noticing that:

(i) Regions of linearity of Gödel functions can be defined independently from formulas (apart from the dimension n of the domain),
(ii) Gödel functions are not continuous, but their discontinuities are located only on boundaries of regions $C \in \mathcal{C}$.

5.2 Functional representation for Product Logic

Since negation has the same interpretation in both Gödel and Product logic, we can repeat the argument of Section 5.1, describing the characteristic function of regions with some component equal to zero.

An *integer monomial* in the variables $x_1, \dots, x_n$ is a function $x_1^{k_1} \dots x_n^{k_n}$ where $k_1, \dots, k_n \in \mathbb{Z}$.

By *piecewise monomial function* we mean a continuous function f such that there exist subsets $D_1, \dots, D_u$ of $(0,1]^n$ and monomial $\pi_1, \dots, \pi_u$ such that the restriction of f to each D_i is either a monomial function or the function constantly (on D_i) equal to 0.

The functional representation theorem for Product logic can be summarized in the following way:

Theorem 10 *Let I be a subset of $\{1, \dots, n\}$, and*

$$C_I = \{\mathbf{x} \in [0,1]^n \mid x_i = 0 \textit{ for } i \in I, 0 < x_j \le 1, j \notin I\}.$$

Then let $f : [0,1]^n \to [0,1]$ be a function such that the restriction of f on every C_I (for every $I \subseteq \{1, \dots, n\}$) is a piecewise monomial function. Then there exists a formula φ such that $\varphi^{\Pi} = f$. Further, for any C_I there exists a formula ψ_I in which only variables p_j with $j \notin I$ appear, and having length less or equal than the length of φ, such that the restriction of f to C_I is equal to the restriction of ψ_I^{Π} to C_I.

For details see [16], [11].

In order to deal with Product logic by means of piecewise linear functions, we shall introduce an infinite-valued logic Σ whose domain of interpretation is the set $[0, \infty]$ of nonnegative real numbers plus a distinct symbol for infinity, and connectives are interpreted as sum and truncated difference.

Interpretation of connectives in Σ is given by the map

$$\iota : x \in [0,1] \to \begin{cases} \log(x^{-1}), & \text{if } x > 0 \\ \infty & \text{otherwise} \end{cases}$$

(the choice of the base of logarithms is immaterial, for concreteness we assume we are dealing with natural logarithms) in such a way that for any evaluation e,

$$e_{\Sigma}(\varphi) = \iota(e_{\Pi}(\varphi)).$$

In this way $\iota(x \cdot y) = \iota(x) + \iota(y)$ and $\iota(x \to y) = 0$ if $\iota(x) \ge \iota(y)$ and $\iota(x \to y) = \iota(x) - \iota(y)$ otherwise. If φ is a tautology of Π then for any evaluation e, $e_{\Sigma}(\varphi) = \iota(e_{\Pi}(\varphi)) = 0$.

Then truth tables of formulas of Σ can be characterized by applying the logarithmic transformation to truth tables of Product formulas.

A continuous piecewise linear function with each piece of the form $\sum_{i \in I} a_i x_i$ is a homogeneous piecewise linear function.

Theorem 11 *Let I be a subset of $\{1, \dots, n\}$, and*

$$C_I = \{\mathbf{x} \in [0, \infty]^n \mid x_i = \infty, x_j \in [0, \infty) \textit{ for } i \in I, j \notin I\}$$

Let $f : [0, \infty]^n \to [0, \infty]$ be a function such that the restriction of f on every C_I (for every $I \subseteq \{1, \dots, n\}$) is either an homogeneous piecewise linear function with integer coefficients or is identically equal to ∞. Then there exists a formula φ such that $\varphi^{\Sigma} = f$. Further, for any C_I there exists a formula ψ_I in which only variables p_j with $j \notin I$ appear, and having length less or equal than the length of φ, such that the restriction of f to C_I is equal to the restriction of ψ_I^{Σ} to C_I.

5.3 Functional representation for $Free_{BL}$ on one generator and $Free_{BL_\Delta}$

In [27,28] Montagna undertakes the study of concrete functional representation of free BL-algebras. Let $m[0,1]_{\text{Ł}}$ be the algebra defined as the ordinal sum of m copies of the standard MV-algebra $[0,1]_{\text{Ł}}$. The free BL-algebra over n generators is the subalgebra $\mathbf{B}_n$ of $((n+1)[0,1]_{\text{Ł}})^{((n+1)[0,1]_{\text{Ł}})^n}$ generated by the projections $(x_1, \dots, x_n) \mapsto x_i$. The problem tackled by Montagna is to characterize this subalgebra as an algebra of functions from $[0, n+1]^n \to [0, n+1]$. While the full characterization is still an open problem, in [27] the problem is completely solved for the free BL-algebra over 1 generator, while in [28] it is solved the easier problem of characterizing free n-generated algebras in the variety of BL_{Δ}-algebras, obtained by adding to BL-algebras the operator

$$\Delta(x) = \begin{cases} 1 & \text{if } x = 1 \\ 0 & \text{otherwise.} \end{cases}$$

For all $a \in \mathbb{R}$ we denote by $i(a)$ the largest integer $b \le a$. For all integers $i > 0$ let $x \odot_i y = \max\{i, x + y - 1 - i\}$ and $x \to_i y = \min\{i+1, 1 - x + y + i\}$. The algebra $m[0,1]_{\text{Ł}} = ([0,m], \&, \to, 0)$ has operations defined as follows:

$$x \& y = \begin{cases} x \odot_{i(x)} y & \text{if } i(x) = i(y) < m \\ \min(x,y) & \text{otherwise,} \end{cases}$$

$$x \to y = \begin{cases} m & \text{if } x \le y \\ y & \text{if } i(y) < i(x) \\ x \to_{i(x)} y & \text{otherwise.} \end{cases}$$

The algebra $m[0,1]_{\text{Ł}}^{\Delta} = ([0,m], \&, \to, \Delta, 0)$ adds to the set of operations above the unary operator Δ such that $\Delta(m) = m$ and $\Delta(x) = 0$ for $x \neq m$.

Let $\varphi^{m\text{Ł}}$ denote the function in $(m[0,1]_{\text{Ł}})^{(m[0,1]_{\text{Ł}})^n}$ determined by formula φ in n variables (with no occurences of Δ).

Let f be a McNaughton function of one variable ($f \in M_1$). Then the functions $f_1, f_2 : [0,2] \to [0,2]$ are defined as follows:

$$f_1(x) = \begin{cases} f(x) \text{ if } f(x) \neq 1 \\ 2 \quad \text{otherwise} \end{cases} \qquad f_2(x) = 1 + f(x-1).$$

Then, [27] proves the following:

Theorem 12 *(i) If $\varphi^{\text{Ł}}(1) = 1$, then there is a function $h \in M_1$ (possibly, $h \neq \varphi^{\text{Ł}}$) such that $h(1) = 1$ and*

$$\varphi^{2\text{Ł}}(x) = \begin{cases} \varphi_1^{\text{Ł}}(x) & \textit{if } 0 \leq x < 1 \\ h_2(x) & \textit{otherwise.} \end{cases}$$

(ii) If $\varphi^{\text{Ł}}(1) = 0$, then

$$\varphi^{2\text{Ł}}(x) = \begin{cases} \varphi_1^{\text{Ł}}(x) & \textit{if } x < 1 \\ 0 & \textit{otherwise.} \end{cases}$$

The free 1-generated BL-algebra is the algebra whose domain is the set of all functions satisfying (i) and (ii) with pointwise defined operations.

The cases of free BL-algebras with $n > 1$ generators are much more complex to deal with, since not all relationships and dependencies linking the behaviour of functions over different regions of the domain $[0, n+1]$ are fully described. Free BL$_\Delta$-algebras allow an easier analysis due to the expressive power of the Δ operator which kills off those inter-dependencies.

Let $I(n)$ denote the set of all $\mathbf{t} = (t_1, \dots, t_n) \in [0, n+1]^n$ such that $\{i(t_j) \mid j = 1, \dots, n\} \cup \{0\} \setminus \{n+1\}$ is an initial segment of natural numbers. Let $I_{n+1} = \{n+1\}$ and $I_j = [j, j+1)$ for $0 \leq j \leq n$. Observe that $I(n)$ is the disjoint union of all sets $I_{k_1} \times I_{k_2} \times \cdots \times I_{k_n}$ for $\{0, k_1, \dots, k_n\} \setminus \{n+1\}$ being an initial segment of $\{0, 1, \dots, n\}$. Such sets are called *cells*.

Let $\mathbf{y} = (y_1, \dots, y_k)$ be a k-tuple of variables among $x_1, \dots, x_n$, let $f \in M_k$ be a McNaughton function and fix $i \leq n+1$. The function $f_{i,n} : I_i^k \to [i, i+1] \cup \{0, n+1\}$ is defined as follows:

- If $i = 0$ then $f_{i,n}(\mathbf{y}) = f(\mathbf{y})$ if $f(\mathbf{y}) \neq 1$, $f_{i,n}(\mathbf{y}) = n+1$ otherwise.
- If $0 < i \leq n$, if $f(1, \dots, 1) = 1$ and $f(y_1 - i, \dots, y_k - i) \neq 1$ then $f_{i,n}(\mathbf{y}) = f(y_1 - i, \dots, y_k - i) + i$.
- If $0 < i \leq n$, if $f(1, \dots, 1) = 1$ and $f(y_1 - i, \dots, y_k - i) = 1$ then $f_{i,n}(\mathbf{y}) = n+1$.
- If $i = n+1$ and $f(1, \dots, 1) = 1$ then $f_{i,n}(\mathbf{y}) = n+1$.
- If $0 < i \leq n+1$ and $f(1, \dots, 1) = 0$ then $f_{i,n}(\mathbf{y}) = 0$.

A subset Y of a cell C is *linear semialgebraic* if there is a system E of linear inequations of the form $f(\mathbf{x}) \lhd g(\mathbf{x})$, for $\lhd \in \{\leq, <\}$, such that Y is the subset of all solutions in C of system E and the following conditions hold for any inequation $e \equiv f(\mathbf{x}) \lhd g(\mathbf{x})$ in E:

i) There is $i(e) \in \{0, 1, \dots, n+1\}$ such that for all $j \leq n$, if either f or g depends on x_j, then $i(t_j) = i(e)$ for all $\mathbf{t} \in C$.
ii) There is a k-tuple $\mathbf{y} = (y_1, \dots, y_k)$ of variables among $x_1, \dots, x_n$ and McNaughton functions $h, k \in M_k$, such that f and g only depend on $\mathbf{y}$, and for all $\mathbf{x} \in C$ one has $f(\mathbf{x}) = h_{i(e),n}(\mathbf{y})$ and $g(\mathbf{x}) = k_{i(e),n}(\mathbf{y})$. Moreover, if $i(e) > 0$, then $h(1, \dots, 1) = k(1, \dots, 1)$.

A partition of $I(n)$ into semialgebraic subsets of cells is a BL$_\Delta$*-partition.* f is an *elementary* BL$_\Delta$*-function* if its domain is an algebraic subset Y of some cell C and there are an $i(f) \in \{1, \dots, n+1\}$ and a McNaughton function g such that for $j = 1, \dots, n$, if g depends on x_j, then for all $\mathbf{t} \in C^n$ we have $i(t_j) = i(f)$. Moreover $f = g_{i(f),n}$.

Theorem 13 *[28] The free BL$_\Delta$-algebra over n generators can be represented as $(D(n), \&, \to, \Delta, 0)$ where $D(n)$ is the set of all functions from $I(n) \to [0, n+1]$ for which there are a BL$_\Delta$-partition $\{P_1, \dots, P_u\}$ and elementary BL$_\Delta$-functions $f_1, \dots, f_u$ such that, for all $i \in \{1, \dots, u\}$, the domain of f_i is P_i, and $f \upharpoonright P_i$ is f_i. The operations $\&, \to, \Delta$ are defined pointwise.*

Once we have Theorem 13, it is easy to find an isomorphic representation of the free BL$_\Delta$-algebra over n generators in terms of an algebra of functions from $[0, n+1]^n \to [0, n+1]$, since all information coded in $I^c(n) := [0, n+1]^n \setminus I(n)$ is actually redundant (see Lemma 9 to get an idea, in a slightly different setting, about how cells $\subseteq I^c(n)$ are just replicas of cells $\subseteq I(n)$. From Theorem 13 it is also trivial to give an isomorphic presentation of the same algebra as an algebra of functions from $[0,1]^n \to [0,1]$. An analogous representation is explored in [5] for studying a class of functions which properly contains all members of free n-generated BL-algebras in order to achieve a finite-valued reduction of Basic Logic: see 5.4 for some details of this construction.

5.4 Finite-valued reductions

As it is shown in [2,3], results of Section 3.1.2 can be considerably improved by applying a technique which has been generalized to other logics in [4]. In this section we review cases in which the technique has been successfully applied.

5.4.1 Łukasiewicz logic Our aim is to find a bound $b_{\text{Ł}} : \varphi \mapsto b_{\text{Ł}}(\varphi) \in \mathbb{N}$ such that for any non-tautological φ a rational point $\mathbf{v} \in [0,1]^n$ such that $\varphi^{\text{Ł}}(\mathbf{v}) < 1$ can always be found with $\text{den}(\mathbf{v}) \leq b_{\text{Ł}}(\varphi)$. As a matter of fact, for Łukasiewicz logic, every vertex of a suitably constructed polyhedral complex $C_\varphi^{\text{Ł}}$ linearly adequate to $\varphi^{\text{Ł}}$ is shown to have denominator $\leq (|\varphi|/n)^n$, for $p_1, \dots, p_n$ being the propositional variables occurring in φ ([3]). We give some details about this result here.

We start by giving a canonical algorithm to form $C^{\text{Ł}}_{\varphi}$, which proceeds by inductively combining the polyhedral complexes linearly adequate to all subformulas of φ, as specified below (where we choose $\{\oplus, \neg\}$ as sufficient set of connectives and where $\mathrm{cl}\, X$ denotes the topological closure of set X):

- $C^{\text{Ł}}_{p_i} = [0,1]^n$.
- $C^{\text{Ł}}_{\neg\psi} = C^{\text{Ł}}_{\psi}$.
- $C^{\text{Ł}}_{\psi\oplus\vartheta} = \{D^+, D^- \mid D = D_1 \cap D_2, D_1 \in C^{\text{Ł}}_{\psi}, D_2 \in C^{\text{Ł}}_{\vartheta}, D^+ = \{\mathbf{x} \in D \mid (\psi \oplus \vartheta)^{\text{Ł}}(\mathbf{x}) = 1\}, D^- = \mathrm{cl}\{\mathbf{x} \in D \mid (\psi \oplus \vartheta)^{\text{Ł}}(\mathbf{x}) < 1\}\}$.

$C^{\text{Ł}}_{\varphi}$ is the *roughest* polyhedral complex linearly adequate to all subformulas of φ, in the sense that any other complex with the same property is obtained by polyhedral subdivision of $C^{\text{Ł}}_{\varphi}$.

The construction in [3] consists in keeping track of the *resources* used to build equations of boundaries of polyhedra of $C^{\text{Ł}}_{\varphi}$.

As a matter of fact, in the inductive definition above, the only place were we split polyhedra and introduce new face boundaries, is when we describe the intersection $D^+ \cap D^-$ with the equation $(\psi \oplus \vartheta)^{\text{Ł}}(\mathbf{x}) = 1$. On every polyhedron D the form of this equation is $p(\mathbf{x}) + q(\mathbf{x}) = 1$ for linear polynomials with integer coefficients p, q such that over D the function $\psi^{\text{Ł}}$ coincides with p while $\vartheta^{\text{Ł}}$ coincides with q.

Each boundary equation $r(\mathbf{x}) = 1$ of some polyhedron of $C^{\text{Ł}}_{\varphi}$ has been introduced the first time by some subformula ψ_r of φ. If ψ introduces the boundary $r(\mathbf{x}) = 1$ we shall denote r by r_{ψ}. Further, we can express each boundary $r(\mathbf{x})$ as $\sum_{i\in I_r} \pm r_{\psi_i}(\mathbf{x})$ where the collection $\{\psi_i\}_{i\in I_r}$ is formed by mutually disjoint occurrences of subformulas of ψ_r.

Thus, let $\mathbf{v}$ be a generic vertex of $C^{\text{Ł}}_{\varphi}$, and let $M_{\mathbf{v}}$ be the integral $n \times n$ matrix of the coefficients of a system $M_{\mathbf{v}}\mathbf{x} = \mathbf{d}$, having $\mathbf{v}$ as its unique solution and whose rows are as described in Section 3.1.2. Each one of the n many rows of the system $M_{\mathbf{v}}\mathbf{x} = \mathbf{d}$, represents a boundary r and has the form $\sum_{i\in I_r} \pm r_{\psi_i}(\mathbf{x})$. Then, determinant preserving multilinear operations, each one of them consisting in subtracting from a row an integral linear combination of the other rows, allows us to transform $M_{\mathbf{v}}$ into an integral matrix M' which has the same determinant of $M_{\mathbf{v}}$ and it is such that the union of all formulas ψ_i in all rows constitutes a set of mutually disjoint occurrences of subformulas of φ (see [2,3] for the many technical details we are omitting here).

Properties of matrix M' allows to relate Hadamard's bound on the determinant of M' to the length of φ. Indeed, let $r_{\psi}(\mathbf{x}) = \sum_{i=1}^{n} a_i x_i$: then an easy induction on the structure of formulas shows that $|a_i|$ is always smaller or equal to the number of occurrences of the variable x_i in ψ. Hence, the sum $\sum_{i,j} |c_{ij}|$ over all entries c_{ij} of M' is smaller or equal to $|\varphi|$. Thus, by Hadamard's inequality, $\mathrm{den}(\mathbf{v}) \leq |\det(M')| \leq (|\varphi|/n)^n$.

A continuity argument and boundedness of slopes of linear pieces of $\varphi^{\text{Ł}}$ shows that whenever there is a vertex $\mathbf{v}$ with $\mathrm{den}(\mathbf{v}) \leq (|\varphi|/n)^n$ such that $\varphi^{\text{Ł}}(\mathbf{v}) < 1$ then there exists a nearby rational point $\mathbf{w}$ with $\mathrm{den}(\mathbf{v})$ dividing $2^{|\varphi|-1}$ such that $\varphi^{\text{Ł}}(\mathbf{w}) < 1$. Then:

Theorem 14 *For any formula φ in the variables $p_1, \ldots, p_n$, Ł $\models \varphi$ if and only if*

$$\text{Ł}_k \models \varphi \ \textit{for all}\ k \leq (|\varphi|/n)^n \quad \textit{iff} \quad \text{Ł}_{2^{|\varphi|-1}} \models \varphi.$$

5.4.2 Gödel logic Functional representation of Gödel logic (see 5.1) allows to conclude, much more directly than in the Łukasiewicz case, that each vertex of C^{G}_{φ} has denominator 1. Due to discontinuity at the intersection of faces of polyhedra in C^{G}_{φ}, vertices does not constitute a sufficiently large set of points where to test φ^{G}: in order to form a sufficiently large set of critical points, we need to add one point picked from the relative interior of each face of each polyhedron in C^{G}_{φ}. This can be accomplished easily by taking *Farey mediants* of sets of vertices: The Farey mediant of a set $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_k\}$ of rational points in $[0,1]^n$ is the point $\mathbf{y}$, lying in the relative interior of the convex hull of X, given by $y_i = \sum_{j=1}^{k} h_{ij} / \sum_{j=1}^{k} d_{ij}$ for $x_{ij} = h_{ij}/d_{ij}$. Hence, we have to check points up to denominator $n+1$. Actually, we can always find a point of denominator exactly $n+1$ in the relative interior $\mathrm{rel\,int} F$ of each face F of C^{G}_{φ}. Then, we find again:

Theorem 15 *For any formula φ in the variables $p_1, \ldots, p_n$:*

$$\text{G} \models \varphi \quad \textit{iff} \quad \text{G}_{n+1} \models \varphi.$$

5.4.3 Product logic The piecewise-linear formulation of Product logic (see 5.2) is apt for applying our technique: consider a function $\varphi^{\Sigma} : [0,\infty]^n \to [0,\infty]$ determined by a formula φ of Product logic.

Then, when dealing with the restriction of φ^{Σ} to $[0,\infty)^n$, one can limit the search space of test points to any closed hypersurface $\mathcal{S}$ splitting $[0,\infty)^n$ in two halves, one of which containing properly $B(0,\epsilon) \cap [0,\infty)^n$ for some n-dimensional ball $B(0,\epsilon)$ centered in 0 and having as radius an arbitrary $0 < \epsilon \in \mathbb{R}$. One convenient choice for such a hypersurface $\mathcal{S}$ is defined by the union of all outer faces of the unit hypercube $[0,1]^n$, that is, $\mathcal{S} = \bigcup_{I\subseteq N} F_I$ for $N = \{1, \ldots, n\}$ and

$$F_I = \{\mathbf{x} \in [0,1]^n \mid x_i = 1 \text{ iff } i \in I\}.$$

It is sufficient to study separately the restriction of φ^{Σ} to each one of the F_I which is $(n-1)$-dimensional, which is equivalent to considering only singleton index sets $F_I = F_{\{x_k\}}$.

Upon identifying $F_{\{x_k\}}$ with the unit hypercube $[0,1]^{n-1}$, we proceed analogously to what we have done for Ł and define a rational polyhedral complex C^{Σ}_{φ} linearly adequate to φ and all its subformulas, by setting

$C^{\Sigma}_{p_i} = F_{\{x_k\}}$, $C^{\Sigma}_{\psi\&\vartheta} = \{D_1 \cap D_2 \mid D_1 \in C^{\Sigma}_{\psi}, D_2 \in C^{\Sigma}_{\vartheta}\}$ and $C^{\Sigma}_{\psi\to\vartheta} = \{D^+, D^- \mid D = D_1 \cap D_2, D_1 \in C^{\Sigma}_{\psi}, D_2 \in C^{\Sigma}_{\vartheta}, D^+ = \mathrm{cl}\{\mathbf{x} \in D \mid (\psi \to \vartheta)^{\Sigma}(\mathbf{x}) > 0\}, D^- = \{\mathbf{x} \in D \mid (\psi \to \vartheta)^{\Sigma}(\mathbf{x}) = 0\}\}$. Then [4] shows that the maximum of denominators of vertices in C^{Σ}_{φ} is not larger than the maximum denominators of vertices in $C^{\text{Ł}}_{\psi}$, where $\psi^{\text{Ł}} : [0,1]^{n-1} \to [0,1]$ is the McNaughton function determined by the formula ψ obtained from φ by substituting all occurrences of p_k with 1. The restriction of φ^{Σ} to $[0,\infty]^n \setminus [0,\infty)^n$ is dealt with in a similar way: let us display $[0,\infty]^n \setminus [0,\infty)^n = \bigcup_{I\subseteq N} G_I$ for $G_I = \{\mathbf{x} \in [0,\infty]^n \setminus [0,\infty)^n \mid x_i = \infty \text{ iff } i \in I\}$. Then, for each $I \subseteq N$, there exists a formula ψ_I of Product logic, not longer than φ, in the variables $p_{\iota(1)}, \ldots, p_{\iota(n-|I|)}$ for $\iota(h)$ being the hth smallest element of $N \setminus I$, such that $(\varphi^{\Sigma} \restriction G_I)(\mathbf{t}) = \psi^{\Sigma}_I(\mathbf{t}')$, for every $\mathbf{t} \in G_I$ and where $t'_h = t_{\iota(h)}$ for all $1 \le h \le n - |I|$.

Apart from Boolean logic, there are no naturally defined finite-valued Product logics: the problem in the linear formulation of the logic is that the range of $\varphi^{\Sigma} \restriction F_I$ is not included in $[0,1]$ and it may include values as high as $|\varphi|$. Then, to state the finite-valued reduction in terms of finitely valued approximating logics, we have to introduce a family of ml-valued logics $\mathcal{S}^l_m$ having an additional unary connective $\diamond$ whose semantics is $\diamond(x) = \lfloor lx \rfloor / lm$ and not to deal directly with φ but with the formula $\hat{\varphi}$ obtained by prefixing each occurrence of variable with $\diamond$ [4].

Theorem 16 *For any formula φ in the variables $p_1, \ldots, p_n$ (with $n > 1$):*

(i) φ is not a tautology if and only if there exists a point $\mathbf{v} \in (([0,1] \cap \mathbb{Q}) \cup \{\infty\})^n$ such that $\mathrm{den}(\mathbf{v}) < ((|\varphi| - 1)/(n-1))^{n-1}$ *and $\varphi^{\Sigma}(\mathbf{v}) > 0$.*

(ii) $\Pi \models \varphi$ iff $\mathcal{S}^l_m \models \hat{\varphi}$, for $m = |\varphi|$ and $l = 2^{m-1}$.

If $n = 1$ then (i) holds with $\mathrm{den}(\mathbf{v}) = 1$ *and (ii) trivially holds.*

5.4.4 Basic logic First we derive a finite-valued reduction for the logic Ł+G obtained by considering all connectives from both Łukasiewicz and Gödel logics. Actually, we just add to Łukasiewicz connectives an additional unary connective $\sim$ interpreted by Gödel negation ($\sim^{\text{Ł}+G} = \neg^{\text{G}}$). The set $\{\oplus^{\text{Ł}}, \neg^{\text{Ł}}, \sim^{\text{Ł}+G}\}$ is sufficient to express all the other operations of Ł + G.

With every formula φ in the expanded language we can canonically associate a polyhedral complex $C^{\text{Ł}+G}_{\varphi}$ linearly adequate to $\varphi^{\text{Ł}+G}$: we just replicate the analogous definition given for the Łukasiewicz case and add $C^{\text{Ł}+G}_{\sim\varphi} = C^{\text{Ł}+G}_{\varphi}$. The maximum size of vertices of $C^{\text{Ł}+G}_{\varphi}$ turns out not being greater than the maximum size of vertices of the complex $C^{\text{Ł}}_{\varphi}$. As in the case of Gödel logics one has to take into account discontinuity between different faces of $C^{\text{Ł}+G}_{\varphi}$ and use Farey mediants to get a point from each relative interior of a face. Then:

Theorem 17 *For any formula φ in the variables $p_1, \ldots, p_n$, Ł + G $\models \varphi$ if and only if*

$$(\text{Ł} + \text{G})_k \models \varphi \;\; \textit{for all} \;\; k \le (n+1)(|\varphi|/n)^n.$$

Theorem 17 allows to find a finite-valued reduction of Basic Logic: in [5] the authors use a partial functional representation of logics given by ordinal sums mŁ of m many copies of Łukasiewicz t-norm. This partial characterization is strictly linked to functional representation of free BL$_{\Delta}$-algebras. Even if the class of functions introduced in [5] does not coincide neither with the free $(m-1)$-generated BL-algebra nor with the free $(m-1)$-generated BL$_{\Delta}$-algebra, however yields a finite-valued reduction of Basic Logic to finite-valued logics $m\text{Ł}_k$. The aim is to show that each critical point $\mathbf{v}$ for $\varphi^{m\text{Ł}}$ is mapped to a critical point $\mathbf{w}$ for $\psi^{\text{Ł}+G}$, for a suitably constructed formula ψ not longer than φ, in such a way that $\mathrm{den}(\mathbf{v}) = m\,\mathrm{den}(\mathbf{w})$.

Functions associated with formulas of $m[0,1]_{\text{Ł}}$ are defined over $[0,1]^n$ (instead of $[0,m]^n$ as it has been done in [28] for free BL$_{\Delta}$-algebras: this change is immaterial, compare with the end of 5.3), hence $[0,1]^n$ is partitioned into m^n semi-closed cubic *cells* each of them having the form

$$D_{\mathbf{k}} = I^m_{k_1} \times I^m_{k_2} \times \cdots \times I^m_{k_n}$$

where each $k_i \in \{1, \ldots, m\}$ and each I^m_j is an interval $I^m_j = [(j-1)/m, j/m)$ apart from $I^m_m = [(m-1)/m, 1]$.

Let $\sigma^m_i : [0,1] \to \mathrm{cl}\, I^m_i$ be defined by $\sigma^m_i(x) = (i - 1 + x)/m$. The behavior of function $\varphi^{m\text{Ł}}$ determined by a formula φ in $m[0,1]_{\text{Ł}}$ can be studied separately over each cell $D_{\mathbf{k}}$. For this purposes it is convenient to introduce an equivalence relation between cells as follows:

Two functions π, $\pi' : \{1, \ldots, n\} \to \{1, \ldots, m\}$ are equivalent if

- $\pi^{-1}(1) = \pi'^{-1}(1)$,
- For every $i, j \in \{1, \ldots, n\}$, $\pi(i) < \pi(j)$ if and only if $\pi'(i) < \pi'(j)$ and $\pi(i) = \pi(j)$ if and only if $\pi'(i) = \pi'(j)$.

Given $\pi : \{1, \ldots, n\} \to \{1, \ldots, m\}$ we let $D_{\pi} = D_{\mathbf{k}}$ for $\mathbf{k} = (\pi(1), \ldots, \pi(n))$. Two cells D_{π} and $D_{\pi'}$ are equivalent if π and π' are equivalent.

For instance, the cell $I_4 \times I_2 \times I_3$ is equivalent to $I_6 \times I_3 \times I_5$, but it is not equivalent to any of the following: $I_2 \times I_4 \times I_3$, $I_4 \times I_1 \times I_3$, $I_4 \times I_2 \times I_2$.

Lemma 9 *Let $\tau_{\pi\pi'}(x_1, \ldots, x_n) = (\tau^m_{\pi(1)\pi'(1)}(x_1), \ldots, \tau^m_{\pi(n)\pi'(n)}(x_n))$ and $\tau^m_{ij} = \sigma^m_j(\sigma^m_i)^{-1}$. If π and π' are equivalent then for every $\mathbf{x} \in D_{\pi}$,*

- *if $\varphi^{m\text{Ł}}(\mathbf{x}) \in I_{\pi(i)}$, then $\tau_{\pi(i)\pi'(i)}(\varphi^{m\text{Ł}}(\mathbf{x})) = \varphi^{m\text{Ł}}(\tau_{\pi\pi'}(\mathbf{x}))$,*
- *if $\varphi^{m\text{Ł}}(\mathbf{x}) = b \in \{0,1\}$, then $\varphi^{m\text{Ł}}(\tau_{\pi\pi'}(\mathbf{x})) = b$.*

Furthermore,

Lemma 10 *Let $S \subseteq D_\pi$ be a k-dimensional polyhedron in the canonical polyhedral complex $C_\varphi^{m\text{Ł}}$ linearly adequate to $\varphi^{m\text{Ł}}$, for some $k \in \{0, \ldots, n\}$ and some π. Then $\varphi^{m\text{Ł}} \restriction \operatorname{rel\,int} S$ depends only on variables (having indexes) in $\pi^{-1}(j)$ for a unique $j \in \{1, \ldots, m\}$.*

If $\mathbf{x} \in D_\pi$ then $\varphi^{m\text{Ł}}(\mathbf{x}) \in \bigcup_{j=1}^{n} I_{\pi(j)} \cup \{0,1\}$. Further, if $\varphi^{m\text{Ł}}(\mathbf{x}) \in I_i$, there exists a formula ψ not longer than φ such that

$$\varphi^{m\text{Ł}}(\mathbf{x}) = \sigma_i(\psi^{\text{Ł}+\text{G}}(\sigma_\pi^{-1}(\mathbf{x}))).$$

Moreover, for each vertex $\mathbf{x}$ of $C_\varphi^{m\text{Ł}}$, the point $\sigma_\pi^{-1}(\mathbf{x})$ is a vertex of $C_\psi^{\text{Ł}+G}$.

Lemma 10 states that each critical point for a function $\varphi^{m\text{Ł}}$ in n variables can be found as the inverse image under a map σ_i of a critical point for a formula ψ shorter than φ in the logic Ł + G. Hence, denominator upper bounds for Ł + G can be used to derive bounds for logic mŁ. Since evaluating φ in $(n+1)$Ł is all we need to check whether it is a tautology we finally have:

Theorem 18 *For any formula φ in the variables $x_1, \ldots, x_n$, BL $\models \varphi$ if and only if*

$$(n+1)\text{Ł}_k \models \varphi \text{ for all } k \leq (n+1)(|\varphi|/n)^n.$$

Acknowledgments

The third author acknowledges gratefully a partial support of the Ministry of Education of the Czech Republic under Research and Development project LN00A056, Institute of Theoretical Computer Science — ITI.

References

1. P. Aglianò, F. Montagna, Varieties of BL-algebras I: general properties, *Journal of Pure and Applied Algebra*, **181**, 105–129, 2003.
2. S. Aguzzoli, *Geometric and Proof–Theoretic Issues in Łukasiewicz Propositional Logics*. PhD thesis, University of Siena, Italy, 1998.
3. S. Aguzzoli, A. Ciabattoni, Finiteness in Infinite-valued Łukasiewicz Logic, *Journal of Logic, Language and Information*, **9**, 5–29, 2000.
4. S. Aguzzoli, B. Gerla, Finite-Valued Reductions of Infinite-Valued Logics, *Archive for Mathematical Logic*, **41**, 361–399, 2002.
5. S. Aguzzoli, B. Gerla, On Countermodels in Basic Logic. *Neural Network World*, **12**, 407–420, 2002.
6. M. Baaz, P. Hájek, J. Krajíček, D. Švejda, Embedding logics into Product logic, *Studia Logica*, **61**, 35–42, 1998.
7. M. Baaz, P. Hájek, F. Montagna, H. Veith, Complexity of t-tautologies, *Annals of Pure and Applied Logic*, **113**, 3–11, 2002.
8. S. Burris, H.P. Sankappanavar, **A Course in Universal Algebra**, Springer, 1981.
9. R. Cignoli, I. M. L. D'Ottaviano, D. Mundici, **Algebraic Foundations of Many-Valued Reasoning**, Trends in Logic, vol. 7, Kluwer, Dordrecht, 2000.
10. R. Cignoli, F. Esteva, L. Godo, A. Torrens, Basic Logic is the Logic of Continuous t-norms and Their Residua, *Soft Computing*, **4**, 106–112, 2000.
11. P. Cintula, B. Gerla, Semi-normal Forms and Functional Representation of Product Fuzzy Logic. *Fuzzy Sets and Systems*, to appear.
12. F. Esteva, L. Godo, P. Hájek, M. Navara, Residuated fuzzy logics with an involutive negation, *Archive for Mathematical Logic*, **39**, 106–112, 2000.
13. F. Esteva, L. Godo, F. Montagna, Equational characterization of the subvarieties of BL generated by t-norm algebras, preprint 2003.
14. G. Ewald, **Combinatorial convexity and algebraic geometry**, Graduate texts in Mathematics. vol. 168, Springer Verlag, Berlin, 1996.
15. B. Gerla, A Note on Functions Associated with Gödel Formulas, *Soft Computing*, **4**, 206–209, 2000.
16. B. Gerla, *Functional representation of many-valued logics based on continuous t-norms.* PhD thesis, University of Milano, Italy, 2000.
17. K. Gödel, Zum intuitionistischen Aussagenkalkül, *Anzeiger Akademie der Wissenschaften Wien*, Math.-naturwissensch. Klasse 69, (1932), 65–66. Also in *Ergebnisse eines matematischen Kolloquiums* 4:40, 1933
18. R. Hähnle, **Automated Deduction in Multiple-Valued Logics**, Oxford University Press, Oxford, 1993.
19. R. Hähnle, Many-valued logic and mixed integer programming, *Annals of Mathematics and Artificial Intelligence*, **12**, 231–264, 1994.
20. R. Hähnle, Proof theory of many-valued logic—linear optimization—logic design: connections and interactions, *Soft Computing*, **1**, 107–119, 1997.
21. P. Hájek, **Metamathematics of Fuzzy Logic**, Kluwer, Dordrecht, 1998.
22. P. Hájek, Basic Fuzzy Logic and BL-algebras, *Soft Computing*, **2**, 124–128, 1998.
23. Z. Haniková, Standard algebras for fuzzy propositional calculi, *Fuzzy Sets and Systems*, **124**, 309–320, 2001.
24. Z. Haniková, A note on the complexity of propositional tautologies of individual t-algebras, *Neural Network World*, **12**, 453–460, 2002.
25. Z. Haniková, *Mathematical and Metamathematical Properties of Fuzzy Logic*, PhD Thesis, Charles University in Prague, 2003.
26. R. McNaughton, A theorem about infinite-valued sentential logic, *Journal of Symbolic Logic*, **16**, 1–13, 1951.
27. F. Montagna, The Free BL-algebra on One Generator, *Neural Network World*, **5**, 837–844, 2000.
28. F. Montagna, Free BL$_\Delta$ Algebras, in **Lectures on Soft Computing and Fuzzy Logic**, Antonio Di Nola and Giangiacomo Gerla editors, 159–171, 2001.
29. P. S. Mostert, A. L. Shields, On the structure of semigroups on a compact manifold with boundary, *Annals of Mathematics*, **65**, 117–143, 1957.
30. D. Mundici, Satisfiability in Many-valued Sentential Logic is NP-complete, *Theoretical Computer Science*, **52**, 145–153, 1987.

31. D. Mundici, A constructive proof of McNaughton's Theorem in infinite-valued logics. *Journal of Symbolic Logic*, **59**, 596–602, 1994.
32. V. Novák, I. Perfilieva, and J. Močkoř. *Mathematical Principles of Fuzzy Logic.* Kluwer, Boston/Dordrecht/London, 1999.
33. A. Rose and J.B. Rosser, Fragments of many-valued statement calculi. *Trans. Amer. Math. Soc.*, 87:1–53, 1958.