On the efficiency of normal form systems for representing Boolean functions

Miguel Couceiro, Erkko Lehtonen, Pierre Mercuriali, Romain Péchoux

On the efficiency of normal form systems for representing Boolean functions

Miguel Couceiro[a], Erkko Lehtonen[b], Pierre Mercuriali[a], Romain Péchoux[a]

[a] *Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France*
{miguel.couceiro, pierre.mercuriali, romain.pechoux}@loria.fr
[b] *Technische Universität Dresden, Institut für Algebra, 01062 Dresden, Germany*
Erkko.Lehtonen@tu-dresden.de

Abstract

A normal form system (NFS) for representing Boolean functions is thought of as a set of stratified terms over a fixed set of connectives. For a fixed NFS $\mathbf{A}$, the complexity of a Boolean function f with respect to $\mathbf{A}$ is the minimum of the sizes of terms in $\mathbf{A}$ that represent f. This induces a preordering of NFSs: an NFS $\mathbf{A}$ is polynomially as efficient as an NFS $\mathbf{B}$ if there is a polynomial P with nonnegative integer coefficients such that the complexity of any Boolean function f with respect to $\mathbf{A}$ is at most the value of P in the complexity of f with respect to $\mathbf{B}$. In this paper we study monotonic NFSs, i.e., NFSs whose connectives are increasing or decreasing in each argument. We describe the monotonic NFSs that are optimal, i.e., that are minimal with respect to the latter preorder. We show that these minimal monotonic NFSs are all equivalent. Moreover, we address some natural questions, e.g.: does optimality depend on the arity of connectives? Does it depend on the number of connectives used? We show that optimal monotonic NFSs are exactly those that use a single connective or one connective and the negation. Finally, we show that optimality does not depend on the arity of the connectives.

Keywords: Boolean function, Normal form system, Complexity, Efficient representation

1. Introduction

Motivation. In this paper, we investigate efficient representations of Boolean functions by terms. The terms we consider are standard terms or Boolean

expressions that can be found in term rewriting systems or standard programming languages [1]. The notion of efficiency that we consider is related to the number of function symbols, called connectives, in a term representing a given Boolean function. In this paper we study normal form systems at a structural level. In particular, we address the following questions.

1. *Does the efficiency depend on the number of such connectives?* One might think that adding extra connectives increases the efficiency but, as we will see, this is not the case.

2. *Does the efficiency depend on the arity of the connectives?* One might think that connectives of greater arity improve efficiency as more information is processed by each connective but, again, this is not the case.

We consider a normal form system (NFS) to be a family of terms with a fixed structure that is complete with respect to Boolean functions, i.e., every Boolean function has a representation in the NFS. A similar framework was considered in [5] based on the notion of clone composition. In fact, not every composition of two clones is a clone. The composition of two clones is contained in their join, and the first main result of [5] is a clone composition table ([5, Table 1, Theorem 2]), which indicates for each pair of clones of Boolean functions whether their composition is a clone or not. With the help of this table, factorizations of the clone Ω of all Boolean functions into minimal clones were considered further, and so-called descending irredundant factorizations of Ω were seen to correspond to certain well-known NFSs of Boolean functions, namely the median, conjunctive, disjunctive, polynomial, and dual polynomial NFSs. Such NFSs were compared in terms of complexity, and the median normal form system proved more efficient than the others.

In the current paper, we relax the conditions for an NFS. In contrast to the framework of [5], which only uses connectives of minimal arity that are generators of minimal clones, we now allow arbitrary connectives (of any arity, not necessarily generating a minimal clone). As in [5], for a fixed NFS $\mathbf{A}$, the complexity $C_{\mathbf{A}}(f)$ of a Boolean function f with respect to $\mathbf{A}$ is the minimum of the sizes of terms in $\mathbf{A}$ that represent f. In this way, we can compare NFSs with respect to this complexity measure: an NFS $\mathbf{A}$ is polynomially as efficient as an NFS $\mathbf{B}$ if there is a polynomial P with nonnegative integer coefficients such that for any Boolean function f, $C_{\mathbf{A}}(f) \leq P(C_{\mathbf{B}}(f))$.

In this paper, we focus on monotonic NFSs, i.e., NFSs whose connectives are increasing or decreasing in each argument.

Main contributions. The main contributions of this paper are the following:

(i) Optimal monotonic NFSs (a monotonic NFS is *optimal* if it is minimal with respect to the preorder just defined) are exactly those monotonic NFSs that use a single connective or one connective and the negation. Moreover, such NFSs are all equivalent, which motivates the notion of optimality.

(ii) The arity of connectives does not impact the efficiency of monotonic NFSs.

Related works. Terms can represent formulas, i.e., circuits where all internal gates have fan-out 1. Studying terms rather than circuits distinguishes syntax and semantics in a clearer manner, and we can profit from the inherent structure of interpretations of terms to derive useful results using clone theory. It was proved in [3, 19] that given a Boolean formula C involving only binary connectives, there is an equivalent formula C' using connectives in $\{\wedge, \vee, \neg\}$ such that

$$\text{leafsize}(C') \leq \text{leafsize}(C)^{\alpha}$$

where $\text{leafsize}(C)$ is the number of leaves in the tree representation of C and for α such that $\frac{1+2^{\alpha}}{3^{\alpha}} \leq \frac{1}{2}$. Our generalization of this result is threefold: first, connectives occurring in terms are applied in a stratified manner, i.e., with respect to some order in the depth of the terms; second, we consider connectives of arbitrary arity and not only binary; third, we consider minimal representations of Boolean functions. A classification of the complexity of satisfiability problems with respect to clause connectives was established in [17]; the paper [2] provides an alternative proof that relies on the Galois connection between functions and relations and Post's classification. Here we do not focus on computational complexity but rather on the representational complexity.

Outline. In Section 2 we recall basic notions on Boolean functions, clones, terms, and term operations, and present some preliminary results. In Section 3 we introduce stratified sets of terms and monotonic NFSs and state some of their properties. Section 4 lays down a framework for comparing

NFSs based on the representational complexity of functions. For that purpose, we introduce reductions between NFSs and show that they translate into comparabilities between NFSs. In particular, we establish the equivalence between several monotonic NFSs. Section 5 is devoted to characterizing the optimal monotonic NFSs. We show that optimal monotonic NFSs are exactly those that use a single connective or one connective and the negation. To this effect, we first show that the median NFS is optimal among monotonic NFS (Theorem 44). The remainder of the proof is obtained by a case analysis showing – making use of reductions between NFSs – that every monotonic NFS based on a single connective and the negation is at least as efficient as the median NFS. In particular, it follows that the representational complexity does not depend on the arity of the connective. This still holds for NFSs that are based on at least two non-unary connectives. This is shown in Section 6, where we furthermore show that any such NFS is equivalent to the conjunctive, disjunctive, polynomial, or dual polynomial NFS.

Still in Section 6, we discuss the case of non-monotonic NFSs and conjecture that they are strictly more efficient than the monotonic ones. In Section 7, we put in perspective the results of the paper and mention some topics of further research.

2. Preliminaries

In this section we recall basic notions of clone theory and normal form systems in the context of Boolean functions. For further background on clone theory, see [13].

2.1. Boolean functions

Throughout the paper we will denote by $\mathbb{B}$ the 2-element set $\{0, 1\}$. We will often designate tuples with boldface letters and their entries by corresponding italic letters with subscripts, e.g., $\mathbf{a} = (a_1, \ldots, a_n)$. The *Hamming distance* between two tuples $\mathbf{a}$ and $\mathbf{b}$, denoted $d(\mathbf{a}, \mathbf{b})$, is the number of positions in which they differ. The *Hamming weight* of a tuple $\mathbf{a}$, denoted $w(\mathbf{a})$, is defined as the number of nonzero entries of $\mathbf{a}$, that is, $w(\mathbf{a}) := d(\mathbf{a}, (0, \ldots, 0))$.

The set $\mathbb{B}$ is endowed with the natural ordering $0 \leq 1$. The set $\mathbb{B}^n$ can thus be endowed with the component-wise ordering of tuples, i.e., $(a_1, \ldots, a_n) \leq (b_1, \ldots, b_n)$ if and only if $\forall i, 1 \leq i \leq n,\ a_i \leq b_i$. A tuple $\mathbf{b}$ is said to *cover* another tuple $\mathbf{a}$, if $\mathbf{a} < \mathbf{b}$ and there is no tuple $\mathbf{c}$ such that $\mathbf{a} < \mathbf{c} < \mathbf{b}$.

a	$\neg(a)$
0	1
1	0

a	b	$a \vee b$	$a \wedge b$	$a \oplus b$	$a \uparrow b$	$a \downarrow b$
0	0	0	0	0	1	1
0	1	1	0	1	1	0
1	0	1	0	1	1	0
1	1	1	1	0	0	0

a	b	c	$\mu(a,b,c)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Table 1: Well-known Boolean functions.

A *Boolean function* is a map $f\colon \mathbb{B}^n \to \mathbb{B}$, for some integer $n \geq 0$ called the *arity* of f. The arity of f is denoted by $\mathtt{ar}(f)$. For a fixed arity n, the n different *projection maps* are the functions defined by $e_i^{(n)}\colon (a_1, \ldots, a_n) \mapsto a_i, 1 \leq i \leq n$. The nullary operations are constants corresponding to the elements of $\mathbb{B}$. With no danger of ambiguity, we will denote any constant function of any arity taking value 0 (resp. 1) by a boldface $\mathbf{0}$ (resp. $\mathbf{1}$).

Other well-known examples of Boolean functions are the unary function $\neg$ (negation, NOT), the binary functions $\vee$ (disjunction, OR), $\wedge$ (conjunction, AND), $\oplus$ (addition modulo 2, exclusive or, XOR), $\uparrow$ (Sheffer stroke, negated conjunction, NAND), $\downarrow$ (Peirce's arrow, negated disjunction, NOR) and ternary function μ (majority), which are defined by the operation tables shown in Table 1.

We will use both prefix and infix notation, e.g., $\vee(a_1, a_2) = a_1 \vee a_2$. For a binary function f, let f_n be defined inductively by $f_2(a_1, a_2) = f(a_1, a_2)$ and $f_{n+1}(a_1, \ldots, a_{n+1}) = f(a_1, f_n(a_2, \ldots, a_{n+1}))$, e.g., $\wedge_3(a_1, a_2, a_3) = \wedge(a_1, \wedge(a_2, a_3))$.

A tuple $\mathbf{a}$ is called a *true point* (resp. *false point*) of a function f if $f(\mathbf{a}) = 1$ (resp. $f(\mathbf{a}) = 0$). We say that $\mathbf{a}$ is a *minimal true point of* f if $\mathbf{a}$ is a true point of f and there is no true point $\mathbf{b}$ of f such that $\mathbf{b} < \mathbf{a}$. Similarly, we say that $\mathbf{a}$ is a *maximal false point of* f if $\mathbf{a}$ is a false point of f and there is no false point $\mathbf{b}$ of f such that $\mathbf{a} < \mathbf{b}$.

For $a \in \mathbb{B}$, a function $f\colon \mathbb{B}^n \to \mathbb{B}$ is *a-preserving* if $f(a, \ldots, a) = a$. A function is *constant-preserving* if it is both 0- and 1-preserving. For a function $f\colon \mathbb{B}^n \to \mathbb{B}$, the *dual* of f is defined as

$$f^{\mathrm{d}}(a_1, \ldots, a_n) := \neg(f(\neg(a_1), \ldots, \neg(a_n))).$$

A function $f\colon \mathbb{B}^n \to \mathbb{B}$ is *self-dual* if $f = f^{\mathrm{d}}$. A function $f\colon \mathbb{B}^n \to \mathbb{B}$ is

symmetric if for any permutation π of $\{1,\ldots,n\}$, we have $f(a_1,\ldots,a_n) = f(a_{\pi(1)},\ldots,a_{\pi(n)})$, for all $a_1,\ldots,a_n \in \mathbb{B}$. A function $f\colon \mathbb{B}^n \to \mathbb{B}$ is *monotone* if for all $\mathbf{a},\mathbf{b} \in \mathbb{B}^n$, $\mathbf{a} \leq \mathbf{b}$ implies $f(\mathbf{a}) \leq f(\mathbf{b})$.

A function $f\colon \mathbb{B}^n \to \mathbb{B}$ is *increasing* (*decreasing,* resp.) in the i-th argument, if for all $a_1,\ldots,a_n,b_i \in \mathbb{B}$, $a_i \leq b_i$ implies

$$f(a_1,\ldots,a_{i-1},a_i,a_{i+1},\ldots,a_n) \leq f(a_1,\ldots,a_{i-1},b_i,a_{i+1},\ldots,a_n)$$
$$(f(a_1,\ldots,a_{i-1},a_i,a_{i+1},\ldots,a_n) \geq f(a_1,\ldots,a_{i-1},b_i,a_{i+1},\ldots,a_n), \text{ resp.}).$$

A function is *pseudo-monotone* if it is increasing or decreasing in each argument.

Fact 1. *A function $f\colon \mathbb{B}^n \to \mathbb{B}$ is pseudo-monotone if and only if there exist a monotone function $g\colon \mathbb{B}^n \to \mathbb{B}$ and a subset $S \subseteq \{1,\ldots,n\}$ such that for all $a_1,\ldots,a_n \in \mathbb{B}$,*

$$f(a_1,\ldots,a_n) = g(l_1,\ldots,l_n),$$

where $l_i = a_i$ if $i \in S$ and $l_i = \neg(a_i)$ if $i \notin S$.

Given $f : \mathbb{B}^n \to \mathbb{B}$, the ith argument of f is *essential* in f, if there exists $(a_1,\ldots,a_n) \in \mathbb{B}^n$ such that

$$f(a_1,\ldots,a_{i-1},0,a_{i+1},\ldots,a_n) \neq f(a_1,\ldots,a_{i-1},1,a_{i+1},\ldots,a_n).$$

Two functions f and g are *equivalent*, denoted $f \cong g$, if each one can be obtained from the other by permutation of arguments and by addition or deletion of inessential arguments. It is not difficult to see that the number of essential arguments is preserved by duality and equivalence of functions. For further background, see e.g., [8, 9, 16, 21].

Example 2. As an illustration of the notions introduced in this subsection, for each one of the Boolean functions $\mathbf{0}$, $\mathbf{1}$, $\neg$, $\vee$, $\wedge$, $\oplus$, $\uparrow$, $\downarrow$, and μ, Table 2 shows its dual and indicates whether it is self-dual, symmetric, monotone, increasing or decreasing in the i-th argument (the indicated property holds for every i), or pseudo-monotone. The projection $e_i^{(n)}$ is both 0-and 1-preserving and self-dual; it is symmetric if and only if $n = 1$; it is monotone, increasing in every argument, decreasing in every argument except the i-th one, and pseudo-monotone.

Every argument is essential in $\neg$, $\vee$, $\wedge$, $\oplus$, $\uparrow$, $\downarrow$ and μ. No argument is essential in $\mathbf{0}$ and $\mathbf{1}$. In $e_i^{(n)}$, the only essential argument is the i-th one.

	$\mathbf{0}$	$\mathbf{1}$	$\neg$	$\vee$	$\wedge$	$\oplus$	$\uparrow$	$\downarrow$	μ
0-preserving	yes	no	no	yes	yes	yes	no	no	yes
1-preserving	no	yes	no	yes	yes	no	no	no	yes
dual	$\mathbf{1}$	$\mathbf{0}$	$\neg$	$\wedge$	$\vee$	$\neg \circ \oplus$	$\downarrow$	$\uparrow$	μ
self-dual	no	no	yes	no	no	no	no	no	yes
symmetric	yes	yes	yes	yes	yes	yes	yes	yes	yes
monotone	yes	yes	no	yes	yes	no	no	no	yes
increasing in the i-th argument	yes	yes	no	yes	yes	no	no	no	yes
decreasing in the i-th argument	yes	yes	yes	no	no	no	yes	yes	no
pseudo-monotone	yes	yes	yes	yes	yes	no	yes	yes	yes

Table 2: Properties of well-known Boolean functions.

All projections are equivalent to each other. All constant functions taking the same value are equivalent to each other. The functions from Table 2 are pairwise non-equivalent. ∎

A *class* of Boolean functions is a subset $\mathcal{C} \subseteq \bigcup_{n \geq 1} \mathbb{B}^{\mathbb{B}^n}$. If f is n-ary and $g_1, \ldots, g_n$ are all m-ary, then their *composition* $f(g_1, \ldots, g_n)$ is the m-ary function given by

$$f(g_1, \ldots, g_n)(a_1, \ldots, a_m) = f(g_1(a_1, \ldots, a_m), \ldots, g_n(a_1, \ldots, a_m)),$$

for all $(a_1, \ldots, a_m) \in \mathbb{B}^m$. This notion extends naturally to classes of functions $\mathcal{I}$ and $\mathcal{J}$. The *composition of $\mathcal{I}$ with $\mathcal{J}$*, denoted $\mathcal{I} \circ \mathcal{J}$, is defined by

$$\mathcal{I} \circ \mathcal{J} := \{f(g_1, \ldots, g_n) \mid n, m \geq 1,\ f \text{ } n\text{-ary in } \mathcal{I},\ g_1, \ldots, g_n \text{ } m\text{-ary in } \mathcal{J}\}.$$

2.2. Clones of Boolean functions

A *clone* is a class $\mathcal{C}$ of Boolean functions that contains all projection maps and that satisfies $\mathcal{C} \circ \mathcal{C} \subseteq \mathcal{C}$ (i.e., it is closed under composition). Ordered by inclusion, the clones of Boolean functions constitute an algebraic lattice where the largest clone is the set of all Boolean functions and the smallest clone is the set of all projections, and where the meet of two clones is their intersection and the join of two clones is the smallest clone that contains their union. This lattice, called Post's lattice, was completely described in [15] and

its Hasse diagram is presented in Figure 1. We will use the nomenclature of [5] and [11].

- The clone of all Boolean functions is denoted by Ω.
- For $a \in \mathbb{B}$, the clone of a-preserving functions is denoted by T_a, and $T_{\mathrm{c}} := T_0 \cap T_1$ is the clone of constant-preserving functions.
- The clone of all monotone functions is denoted by M, and $M_x := M \cap T_x$, for $x \in \{0, 1, \mathrm{c}\}$.
- The clone of all self-dual functions is denoted by S, and $S_{\mathrm{c}} := S \cap T_{\mathrm{c}}$, $SM := S \cap M$.
- The clone of all linear functions is denoted by L, i.e.,

$$L := \{f \in \Omega \mid f \cong \oplus_n \text{ or } f \cong \neg(\oplus_n) \text{ for some } n \geq 2\}$$
$$\cup \{e_i^{(n)}, \neg(e_i^{(n)}) \mid 1 \leq i \leq n\} \cup \{\mathbf{0}, \mathbf{1}\},$$

  $L_x := L \cap T_x$, for $x \in \{0, 1, \mathrm{c}\}$, and $LS := L \cap S$.

A set $A \subseteq \{0,1\}^n$ is said to be *a-separating*, $a \in \mathbb{B}$, if there is some i, $1 \leq i \leq n$, such that for every $(a_1, \ldots, a_n) \in A$ we have $a_i = a$. A function f is said to be *a-separating* if $f^{-1}(a)$ is a-separating. The function f is said to be *a-separating of rank $k \geq 2$* if every subset $A \subseteq f^{-1}(a)$ of size at most k is a-separating.

For example, $\wedge$ is 1-separating but not 0-separating and, dually, $\vee$ is 0-separating but not 1-separating. For any $n \geq 3$, the function $f\colon \mathbb{B}^n \to \mathbb{B}$ defined by the rule $f(\mathbf{x}) = 1 \iff w(\mathbf{x}) \geq n-1$ is 1-separating of rank $n-1$ but not 1-separating of rank n.

- For $m \geq 2$, the clones of all 1- and 0-separating functions of rank m are denoted by U_m and W_m, respectively, and the clones of all 1- and 0-separating functions are denoted by U_∞ and W_∞, respectively. For $m = 2, \ldots, \infty$, $T_{\mathrm{c}}U_m := T_{\mathrm{c}} \cap U_m$, $T_{\mathrm{c}}W_m := T_{\mathrm{c}} \cap W_m$, $MU_m := M \cap U_m$, $MW_m := M \cap W_m$, $M_{\mathrm{c}}U_m := M_{\mathrm{c}} \cap U_m$, $M_{\mathrm{c}}W_m := M_{\mathrm{c}} \cap W_m$.
- The clone of all conjunctions and constants is denoted by Λ, i.e.,

$$\Lambda := \{f \in \Omega \mid f \cong \wedge_n \text{ for some } n \geq 2\} \cup \{e_i^{(n)} \mid 1 \leq i \leq n\} \cup \{\mathbf{0}, \mathbf{1}\},$$

  and $\Lambda_x := \Lambda \cap T_x$, for $x \in \{0, 1, \mathrm{c}\}$.

- The clone of all disjunctions and constants is denoted by V, i.e.,

  $$V := \{f \in \Omega \mid f \cong \vee_n \text{ for some } n \geq 2\} \cup \{e_i^{(n)} \mid 1 \leq i \leq n\} \cup \{\mathbf{0}, \mathbf{1}\},$$

  and $V_x := V \cap T_x$, for $x \in \{0, 1, \mathrm{c}\}$.

- The clone of all projections, negated projections, and constants is denoted by $\Omega(1)$, the clone of all projections and negated projections is denoted by I^*, the clone of all projections and constants is denoted by I, and $I_x := I \cap T_x$, for $x \in \{0, 1, \mathrm{c}\}$.

Let $\mathcal{F}$ be a set of Boolean functions. The clone *generated* by $\mathcal{F}$, denoted $\mathcal{C}(\mathcal{F})$, is the smallest clone that contains $\mathcal{F}$, i.e., $\mathcal{C}(\mathcal{F}) = \bigcap_{\mathcal{C} \text{ a clone, } \mathcal{F} \subseteq \mathcal{C}} \mathcal{C}$. In the particular case where $\mathcal{F} = \{f\}$, we write simply $\mathcal{C}(f)$ and say that f is a *generator* of $\mathcal{C}(f)$.

Example 3. The clone SM is generated by the $(2k+1)$-ary *majority function* $\mu_{2n+1} \colon \mathbb{B}^{2k+1} \to \mathbb{B}$ $(k \geq 1)$, defined by the rule $\mu_{2n+1}(\mathbf{x}) = 1$ if and only if $w(\mathbf{x}) \geq k+1$. Note that $\mu_3 = \mu$ (see Table 1). The clones $M_{\mathrm{c}}U_\infty$ and $M_{\mathrm{c}}W_\infty$ are generated by the ternary functions $\mathtt{u}$ and $\mathtt{w}$, respectively, which are defined by

$$\mathtt{u}(a_1, a_2, a_3) := (a_1 \vee a_2) \wedge a_3,$$
$$\mathtt{w}(a_1, a_2, a_3) := (a_1 \wedge a_2) \vee a_3,$$

for all $a_1, a_2, a_3 \in \mathbb{B}$. ∎

Definition 4 (Sheffer and quasi-Sheffer functions). A function f is *Sheffer* (resp. *quasi-Sheffer*) if $\Omega = \mathcal{C}(f)$ (resp. $\Omega = \mathcal{C}(f) \circ \Omega(1)$). A clone is *precomplete* if it contains quasi-Sheffer functions.

Example 5. Clearly every Sheffer function is also quasi-Sheffer but the converse is not true. For example, the function $\downarrow$ is Sheffer and thus quasi-Sheffer, whereas the ternary majority μ is quasi-Sheffer but not Sheffer. Indeed $\mathcal{C}(\mu) = SM \neq \Omega$.

Proposition 6. *A function f is quasi-Sheffer if and only if $f \notin V \cup L \cup \Lambda$.*

Proof. The result can be read directly from the clone composition table of [5]. Since the function class composition is monotone with respect to subset inclusion, and since $SM \circ \Omega(1) = M_{\mathrm{c}}U_\infty \circ \Omega(1) = M_{\mathrm{c}}W_\infty \circ \Omega(1) = \Omega$, if a function f is not quasi-Sheffer, then $\mathcal{C}(f) \subset V \cup L \cup \Lambda$. Conversely, if $f \in V \cup \Lambda \cup L$, then $\mathcal{C}(f) \circ \Omega(1) \neq \Omega$. □

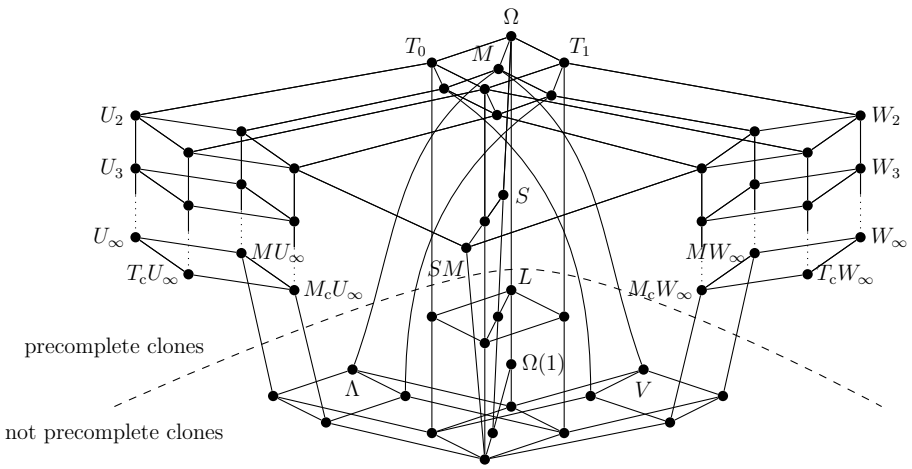

Figure 1: Post's lattice with precomplete clones.

Example 7. The clone SM of self-dual monotone functions is precomplete as it contains the ternary majority function μ that is quasi-Sheffer. The clone Λ of conjunctions is not precomplete. ■

The separation between precomplete and not precomplete clones is illustrated in Figure 1.

2.3. Terms, algebras, term operations

We briefly recall from universal algebra the notions of terms, algebras, and term operations. Let F be a set of *operation symbols* or *connectives,* and let $\tau\colon F \to \mathbb{N}$ be a map, called an (*algebraic similarity*) *type,* that assigns to each operation symbol its *arity.* An *algebra* is a pair $\mathbf{A} = (A, F^{\mathbf{A}})$, where A is a nonempty set, called the *carrier* or *universe* of $\mathbf{A}$, and $F^{\mathbf{A}} = (f^{\mathbf{A}} : f \in F)$ is an indexed family of operations on A, each $f^{\mathbf{A}}$ of arity $\tau(f)$.

Let $\tau\colon F \to \mathbb{N}$ be an algebraic similarity type, and let X be a set disjoint from F. The elements of X are called *variables.* We define *terms* of type τ over X inductively as follows:

(i) Each variable $x \in X$ is a term.

(ii) If $c \in F$ and $\tau(c) = 0$, then c is a term.

(iii) If $f \in F$, $\tau(f) > 0$, and $t_1, \ldots, t_{\tau(f)}$ are terms, then $ft_1 \ldots t_{\tau(f)}$ is a term.

We denote by $T_\tau(X)$ the set of all terms of type τ over X.

Unless otherwise mentioned, we consider terms over the so-called standard set of variables, that is, $X = \{x_i : i \in \mathbb{N}\}$. We say that a term $t \in T_\tau(X)$ is *n-ary* if the variables occurring in t are among $x_1, \ldots, x_n$.

When writing down a term, we may add some commas and parentheses for the sake of clarity. Thus we may write $f(t_1, \ldots, t_{\tau(f)})$ for $ft_1 \ldots t_{\tau(f)}$. These punctuation symbols are formally not part of a term. We may also use the usual infix notation for binary function symbols. Thus we may write $t_1 \, \alpha \, t_2$ for $\alpha t_1 t_2$ if α is a binary operation symbol.

Let $\mathbf{A}$ be an algebra of type τ. Each n-ary term $t \in T_\tau(X)$ induces an n-ary operation $t^{\mathbf{A}}$ on A as follows:

(i) If $t = x_i \in X$, then $t^{\mathbf{A}}$ is the i-th n-ary projection map $e_i^{(n)}$.

(ii) If $t = c \in F$ with $\tau(c) = 0$, then $t^{\mathbf{A}}$ is the n-ary constant operation taking value c everywhere.

(iii) If $t = ft_1 \ldots t_{\tau(f)}$, then $t^{\mathbf{A}} = f^{\mathbf{A}}(t_1^{\mathbf{A}}, \ldots, t_{\tau(f)}^{\mathbf{A}})$.

The operation $t^{\mathbf{A}}$ is called the *term operation* induced by t on $\mathbf{A}$. We also say that $t^{\mathbf{A}}$ is the *interpretation* of t in $\mathbf{A}$, or that the term t *represents* the function $t^{\mathbf{A}}$.

Note that if a term is n-ary, then it is also n'-ary for all $n' \geq n$. Hence the arity is not an inherent part of a term, and it should be specified whenever term operations are considered. It will, however, in most cases be clear from the context.

A term t is *linear* if no variable occurs more than once in t. Any subword of a term t that is itself a term is called a *subterm* of t. Given a term t with variables $x_1, \ldots, x_n$ and terms $t_1, \ldots, t_n$, the term $t\{t_1/x_1, \ldots, t_n/x_n\}$ is obtained from t by replacing every instance of x_i by t_i in t.

In this paper, we consider a particular algebraic similarity type τ and a particular algebra of type τ. Namely, we take as the set of operation symbols the set of all Boolean functions, that is, $F = \Omega$, and we define $\tau\colon \Omega \to \mathbb{N}$ as $\tau(f) := \mathtt{ar}(f)$ for all $f \in \Omega$. We let $\mathbf{B} = (\mathbb{B}, \Omega^{\mathbf{B}})$ be the algebra of type τ, where for each $f \in \Omega$, $f^{\mathbf{B}} = f$. In this way, we can build terms using

Boolean functions as operation symbols, and they are interpreted in $\mathbf{B}$ in an obvious, natural way as Boolean functions.

We will use letters $s, t, s', t', \ldots$ to designate terms in $T_\tau(X)$. Variables and terms of the form $\neg(x_i)$ for some variable x_i are called *literals.* Given a term t and an integer $k > 0$, let t^k be a shorthand notation for the string defined inductively by $t^1 := t$ and $t^{n+1} := tt^n$.

We say that two n-ary terms $s, t \in T_\tau(X)$ are *equivalent,* and we write $s \equiv t$, if $s^{\mathbf{B}} = t^{\mathbf{B}}$. For a term $t \in T_\tau(X)$, we often denote the term operation $t^{\mathbf{B}}$ by $[t]$. For a set $S \subseteq T_\tau(X)$, we define the *interpretation* of S as $[S] := \{[t] \mid t \in S\}$.

Example 8. Consider the binary terms $\mu(x_1, x_2, 1)$ and $x_1 \vee x_2$ in $T_\tau(X)$. It is rather easy to verify that

$$[\mu(x_1, x_2, 1)] = \mu(e_1^{(2)}, e_2^{(2)}, 1) = \vee(e_1^{(2)}, e_2^{(2)}) = [x_1 \vee x_2].$$

In other words, we have $\mu(x_1, x_2, 1) \equiv x_1 \vee x_2$, and both terms represent the function $\vee$. ∎

3. Normal form systems

In this section, we adapt the notion of normal form systems from [5] and make explicit the structure of the terms they induce.

3.1. Normal form systems

Definition 9. Given a sequence $\alpha_1, \ldots, \alpha_n$ of distinct connectives, we say that a term $t \in T_\tau(X)$ is *stratified* with respect to $\alpha_1, \ldots, \alpha_n$ if

(i) the operation symbols occurring in t are among $\alpha_1, \ldots, \alpha_n, 0, 1$; and

(ii) every subterm in t of the form $\alpha_j(t_1, \ldots, t_{\mathtt{ar}(\alpha_j)})$, $j \in \{1, \ldots, n\}$, has no subterm of the form $\alpha_i(t'_1, \ldots, t'_{\mathtt{ar}(\alpha_i)})$ with $i < j$.

We denote by $T(\alpha_1 \cdots \alpha_n)$ the set of all terms stratified with respect to $\alpha_1, \ldots, \alpha_n$.

Example 10. Let α be a ternary connective and β a binary connective. The term $\beta(\alpha(x_1, 0, x_3), x_4)$ belongs to $T(\beta\alpha)$, $T(\beta\alpha\neg)$ and $T(\neg\beta\alpha)$ but it does not belong to $T(\alpha)$ and $T(\alpha\beta)$. The term $\beta(\neg(\alpha(x_1, x_2, x_3)), 1)$ belongs to $T(\beta\neg\alpha)$ but not to $T(\beta\alpha\neg)$. ∎

Remark 11. In a term t of $T(\alpha_1\alpha_2\cdots\alpha_n\neg)$, the negation $\neg$ can only be applied to (iterated negations of) variables or constants. For example, $\neg(\neg(x))$ and $\neg(0)$ can be subterms of t but $\neg(\alpha_i(t_1,\ldots,t_{\mathtt{ar}(\alpha_i)}))$ cannot.

Definition 12. For a sequence $\alpha_1,\ldots,\alpha_n$ of distinct connectives, we say that the set $T(\alpha_1\cdots\alpha_n)$ of stratified terms is *redundant* if there exists an $i\in\{1,\ldots,n\}$ such that $[T(\alpha_1\cdots\alpha_{i-1}\alpha_{i+1}\cdots\alpha_n)]=[T(\alpha_1\cdots\alpha_n)]$. Otherwise it is called *irredundant*.

Example 13. For example, the set $T(\uparrow\neg)$ is redundant because $[T(\uparrow)]=\Omega$ since $\uparrow(x,x)\equiv\neg x$. However $T(\mu\neg)$ is irredundant as $[T(\mu)]=SM\neq\Omega$ and $[T(\neg)]=I^*\neq\Omega$. ∎

In [5], it was observed that factorizations of the clone Ω yield NFSs. For example, the factorization

$$\Omega=\mathcal{C}(\vee)\circ\mathcal{C}(\wedge)\circ\mathcal{C}(\neg)$$

expresses the fact that every Boolean function has a representation in disjunctive normal form (DNF).

We adapt the notion of NFS slightly to make explicit the connectives appearing in the NFS.

Definition 14. A *normal form system* (NFS) is an irredundant set $T(\alpha_1\cdots\alpha_n)$ of stratified terms such that $[T(\alpha_1\cdots\alpha_n)]=\Omega$. If all α_i are pseudo-monotone functions then $T(\alpha_1\cdots\alpha_n)$ is called *monotonic.*

Definition 15. The NFSs defined below are called *basic NFSs.*

- $\mathbf{M}:=T(\mu\neg)$;
- $\mathbf{M}_{2n+1}:=T(\mu_{2n+1}\neg)$;
- $\mathbf{W}:=T(\mathtt{w}\neg)$;
- $\mathbf{U}:=T(\mathtt{u}\neg)$;
- $\mathbf{D}:=T(\vee\wedge\neg)$;
- $\mathbf{C}:=T(\wedge\vee\neg)$;
- $\mathbf{S}:=T(\uparrow)$;
- $\mathbf{S}^{\mathrm{d}}:=T(\downarrow)$;
- $\mathbf{P}:=T(\oplus\wedge)$;
- $\mathbf{P}^{\mathrm{d}}:=T(\oplus\vee)$.

The NFSs $\mathbf{M}$, $\mathbf{C}$, $\mathbf{D}$, $\mathbf{P}$ and $\mathbf{P}^{\mathrm{d}}$ respectively, correspond to the usual median, conjunctive, disjunctive, polynomial and dual polynomial normal forms. Notice that apart from $\mathbf{P}$ and $\mathbf{P}^{\mathrm{d}}$ all the basic NFSs are monotonic.

3.2. Properties of normal form systems

The interpretation of any term in $T(\alpha_1 \cdots \alpha_n)$ can be expressed as an ordered composition of functions in $\mathcal{C}(\alpha_1), \ldots, \mathcal{C}(\alpha_n)$ and I, respectively.

Fact 16. $[T(\alpha_1 \cdots \alpha_n)] = \mathcal{C}(\alpha_1) \circ \cdots \circ \mathcal{C}(\alpha_n) \circ I$.

For example, $[\mathbf{M}] = [T(\mu\neg)] = \Omega = \mathcal{C}(\mu) \circ \mathcal{C}(\neg) \circ I$.

The clone composition table in [5] reveals the following.

Fact 17. *For every clone $\mathcal{C}$, the composition $\mathcal{C} \circ I$ is a clone, namely the clone generated by $\mathcal{C} \cup I$. Moreover, $\mathcal{C} \circ I = I \circ \mathcal{C} \circ I$. Consequently, for any clones $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_n$, we have $\mathcal{C}_1 \circ \mathcal{C}_2 \circ \ldots \mathcal{C}_n \circ I = (\mathcal{C}_1 \circ I) \circ (\mathcal{C}_2 \circ I) \circ \cdots \circ (\mathcal{C}_n \circ I)$.*

Lemma 18. *Let $\alpha_1, \ldots, a_n$ and $\beta_1, \ldots, \beta_n$ be connectives such that $[T(\alpha_i)] = [T(\beta_i)]$, i.e., $\mathcal{C}(\alpha_i) \circ I = \mathcal{C}(\beta_i) \circ I$, for all $i \in \{1, \ldots, n\}$. Then $[T(\alpha_1 \cdots \alpha_n)] = [T(\beta_1 \cdots \beta_n)]$.*

Proof. By Facts 16 and 17,

$$\begin{aligned}[T(\alpha_1 \cdots \alpha_n)] &= \mathcal{C}(\alpha_1) \circ \cdots \circ \mathcal{C}(\alpha_n) \circ I = (\mathcal{C}(\alpha_1) \circ I) \circ \cdots \circ (\mathcal{C}(\alpha_n) \circ I) \\ &= (\mathcal{C}(\beta_1) \circ I) \circ \cdots \circ (\mathcal{C}(\beta_n) \circ I) = \mathcal{C}(\beta_1) \circ \cdots \circ \mathcal{C}(\beta_n) \circ I \\ &= [T(\beta_1 \cdots \beta_n)]. \end{aligned}$$

□

If at least two connectives and the negation are used to build an NFS, then irredundancy forces those connectives to belong to the clone of conjunctions, to the clone of disjunctions or to the clone of linear functions.

Lemma 19. *If $T(\alpha_1 \cdots \alpha_n\neg)$, with $n > 1$, is an NFS, then each α_i is in $V \cup L \cup \Lambda$.*

Proof. Suppose that there exists an i such that α_i is not in $V \cup L \cup \Lambda$. By Proposition 6, α_i is quasi-Sheffer, i.e., $\mathcal{C}(\alpha_i) \circ \Omega(1) = \Omega$. Hence $[T(\alpha_i\neg)] = \mathcal{C}(\alpha_i) \circ \mathcal{C}(\neg) \circ I = \mathcal{C}(\alpha_i) \circ \Omega(1) = \Omega$, so $T(\alpha_1 \cdots \alpha_n\neg)$ is redundant and not an NFS. □

Moreover, by irredundancy again, there cannot be more than 2 such connectives.

Proposition 20. *If $\alpha_1, \ldots, \alpha_n \in \Omega \setminus \Omega(1)$ and $T(\alpha_1 \cdots \alpha_n\neg)$ is an NFS, then $n \leq 2$.*

Proof. Suppose, to the contrary, that $n \geq 3$. By Lemma 19, $\alpha_1, \ldots, \alpha_n$ are all in $V \cup L \cup \Lambda$. For $i \in \{1, \ldots, n\}$, let

$$\beta_i := \begin{cases} \vee & \text{if } \alpha_i \in V \setminus \Omega(1), \\ \wedge & \text{if } \alpha_i \in \Lambda \setminus \Omega(1), \\ \oplus & \text{if } \alpha_i \in L \setminus \Omega(1). \end{cases}$$

Then $\mathcal{C}(\alpha_i) \circ I = \mathcal{C}(\beta_i) \circ I$ for all $i \in \{1, \ldots, n\}$, so it follows from Lemma 18 that $[T(\alpha_{i_1} \cdots \alpha_{i_\ell} \neg)] = [T(\beta_{i_1} \cdots \beta_{i_\ell} \neg)]$ for any $i_1, \ldots, i_\ell \in \{1, \ldots, n\}$.

If $\beta_i = \beta_{i+1}$ for some $i \in \{1, \ldots, n-1\}$, then

$$\begin{aligned} [T(\alpha_1 \cdots \alpha_n \neg)] &= [T(\beta_1 \cdots \beta_n \neg)] = [T(\beta_1 \cdots \beta_i \beta_{i+2} \cdots \beta_n \neg)] \\ &= [T(\alpha_1 \cdots \alpha_i \alpha_{i+2} \cdots \alpha_n \neg)]; \end{aligned}$$

therefore $T(\alpha_1 \cdots \alpha_n \neg)$ is redundant and hence not an NFS, a contradiction.

Assume now that $\beta_i \neq \beta_{i+1}$ for all $i \in \{1, \ldots, n-1\}$. Then there must exist indices i, j with $i < j$ such that $(\beta_i, \beta_j) \in \{(\vee, \wedge), (\wedge, \vee), (\oplus, \wedge), (\oplus, \vee)\}$. Since $T(\vee\wedge\neg)$, $T(\wedge\vee\neg)$, $T(\oplus\wedge)$ and $T(\oplus\vee)$ are basic NFSs (see Definition 15), it follows that $T(\beta_1 \cdots \beta_n \neg)$ and consequently also $T(\alpha_1 \cdots \alpha_n \neg)$ is redundant and hence not an NFS, again a contradiction. □

In [20], Wernick shows that there is no non-redundant complete sets of more than three binary logical connectives. Proposition 20 extends this result to logical connectives of arbitrary arity in the case of stratified terms with negations at the bottom.

Corollary 21. *If $T(\alpha_1 \cdots \alpha_n \neg)$ is an NFS, then either $n = 2$ and each α_i is in $V \cup \Lambda$, or $n = 1$ and α_1 is quasi-Sheffer.*

This corollary motivates the following definition.

Definition 22. An NFS is Sheffer (quasi-Sheffer) if it is of the form $T(\alpha)$ ($T(\alpha\neg)$, respectively).

4. Efficiency of normal form systems

In [5], a preordering of NFSs was introduced that relates two NFSs **A** and **B** if **A** is polynomially at least as efficient as **B**. In this section we extend this preorder to compare arbitrary sets of terms, not necessarily NFSs. We propose reductions to convert terms from one NFS to another, that we use to compare NFSs and extend the results of [5] to all basic NFSs.

4.1. Efficiency

Given a term $t \in T_\tau(X)$ and $b \in \Omega \cup X$, denote by $|t|_b$ the number of occurrences of the symbol b in t. The *size* of a term t, denoted by $|t|$, is the number of occurrences of all connectives, distinct from 0, 1 and $\neg$, in t: $|t| = \sum_{\alpha \in \Omega \setminus \{0,1,\neg\}} |t|_\alpha$. E.g., $|x \wedge (\neg y \vee 1)| = |x \wedge (\neg y \vee 1)|_\wedge + |x \wedge (\neg y \vee 1)|_\vee = 1 + 1 = 2$.

Remark 23. Our definition of the size of a term is perhaps a bit unusual, as we choose not to count constants, negations, nor variables. It is easy to see that the number of variables or constants occurring in a term is linear in the number of (non-nullary) connectives. Moreover, in the shortest representation of a given function in a given NFS, the number of negations is bounded above by the number of variables. Polynomial differences in the size of terms are insignificant in the analysis of the efficiency of normal form systems that will follow. Consequently, whether constants, negations, and variables are counted or not has no bearing on our results, and we simply omit them in order to make calculations a little easier.

Definition 24 (Complexity). Given a set of terms T, the *complexity* of a Boolean function $f \in [T]$ with respect to T, denoted $C_T(f)$, is defined by:

$$C_T(f) = \min\{|t| \colon t \in T,\ [t] = f\}.$$

Example 25. We have that $C_{\mathbf{M}}(\mu) = 1$ because $\mu(x, y, z)$ is the smallest term in $T(\mu\neg)$ that represents μ. However, $C_{T(\vee\wedge\neg)}(\mu) = 5$ because the term $(x \wedge y) \vee (y \wedge z) \vee (z \wedge x)$ is the smallest term in $T(\vee \wedge \neg)$ that represents μ. ∎

Notice that for a set of terms T, the complexity C_T is a partial function on Ω. However the complexity $C_{\mathbf{A}}$ of an NFS $\mathbf{A}$ is a total function as $[\mathbf{A}] = \Omega$. We generalize the notion of efficiency of [5] to compare sets of terms that are not necessarily NFSs.

Definition 26 (Efficiency). Let T, S be two sets of terms such that $[S] \subseteq [T]$.

- T is *polynomially at least as efficient as* S, denoted $T \preceq S$, if there is a polynomial $P \in \mathbb{N}[X]$ such that $\forall f \in [S],\ C_T(f) \leq P(C_S(f))$.
- T and S are *incomparable*, denoted $T \parallel S$, if $T \not\preceq S$ and $S \not\preceq T$.
- T is *polynomially more efficient than* S, denoted $T \prec S$, if $T \preceq S$ and $S \not\preceq T$.

- T and S are *equivalent*, denoted $T \sim S$, if $T \preceq S$ and $S \preceq T$.

For convenience, we will write $S \succeq T$ if $T \preceq S$.

Hence $\preceq$ is a preorder that is not total and $\sim$ is an equivalence relation on the power set of the set of all terms.

Theorem 27 ([5, Theorem 5])**.** *For every pair of NFSs* $\mathbf{A}, \mathbf{B} \in \{\mathbf{C}, \mathbf{D}, \mathbf{P}, \mathbf{P}^{\mathrm{d}}\}$*, if* $\mathbf{A} \neq \mathbf{B}$*, then* $\mathbf{A} \parallel \mathbf{B}$*. Moreover,* $\mathbf{M} \prec \mathbf{C}, \mathbf{D}, \mathbf{P}, \mathbf{P}^{\mathrm{d}}$*.*

We are interested in minimal monotonic NFSs, i.e., monotonic NFSs that are minimal for the preorder $\preceq$.

As we will see, such minimal NFSs exist and they are all equivalent. This motivates the following notion of optimality.

Definition 28. A monotonic NFS $\mathbf{A}$ is *optimal* if $\mathbf{A}$ is minimal and there is no monotonic NFS $\mathbf{B}$ that is incomparable to $\mathbf{A}$, or, equivalently, if for every monotonic NFS $\mathbf{B}$, we have $\mathbf{A} \preceq \mathbf{B}$.

4.2. Linear and quasi-linear reductions

As will become clear from Theorems 44 and 46, the optimal monotonic NFSs are of the form $T(\alpha)$ or $T(\alpha\neg)$. For this reason, in the remainder of this section and in Section 5, we will focus on NFSs of these forms.

In this subsection, we will define relations between sets of terms based on the way we can convert terms from one set to the other. We shall make use of these relations to establish the equivalence between optimal monotonic NFSs. The most fortunate situation is that the connectives of one NFS can be represented as linear terms in the other; then a straightforward substitution of such terms for connectives provides an efficient conversion. As we will see, efficient conversions are possible also under more relaxed conditions.

To illustrate, consider the equivalence $\mathtt{u}(x, y, z) \equiv \mu(\mu(x, 1, y), 0, z)$ that allows us to convert terms in $\mathbf{U}$ into terms in $\mathbf{M}$. Using this equality, terms are converted with at most an affine increase of size: each connective $\mathtt{u}$ is replaced by exactly two connectives μ and variables are not repeated. Indeed, if $t_{\mathbf{U}}^{\min}$ is a minimal representation in $\mathbf{U}$ of a Boolean function $f \in \Omega$, then $C_{\mathbf{U}}(f) = |t_{\mathbf{U}}^{\min}|$. If $t_{\mathbf{M}}$ is the result of converting $t_{\mathbf{U}}^{\min}$ using the above equivalence, then $C_{\mathbf{M}}(f) \leq |t_{\mathbf{M}}|$, and as $|t_{\mathbf{M}}| = 2|t_{\mathbf{U}}^{\min}|$, we obtain $C_{\mathbf{M}}(f) \leq 2C_{\mathbf{U}}(f)$.

Definition 29 (Reductions). Consider two sets of terms $A = T(\alpha\neg)$ (or $T(\alpha)$) and $B = T(\beta\neg)$ (or $T(\beta)$) such that $[A] \subseteq [B]$. We say that

- there exists a *linear reduction* from A to B, or that A is *linearly reducible to* B, denoted $A \sqsupseteq B$, if there exists a linear term $t \in T(\beta)$ such that $\alpha(x_1, \ldots, x_{\mathtt{ar}(\alpha)}) \equiv t$;

- there exists a *universal quasi-linear reduction* from A to B, or that A is *universally reducible to* B, denoted $A \sqsupseteq_\forall B$, if for all $j \in \{1, \ldots, \mathtt{ar}(\alpha)\}$, there exists a $t_j \in T(\beta)$ such that $\alpha(x_1, \ldots, x_{\mathtt{ar}(\alpha)}) \equiv t_j$ and $|t_j|_{x_j} = 1$;

- there exists an *existential quasi-linear reduction* from A to B, or that A is *existentially reducible to* B, denoted $A \sqsupseteq_\exists B$, if there exists a $t \in T(\beta)$ such that $\alpha(x_1, \ldots, x_{\mathtt{ar}(\alpha)}) \equiv t$ and $|t|_{x_j} = 1$ for some $j \in \{1, \ldots, \mathtt{ar}(\alpha)\}$.

Remark 30. Linear reducibility is somewhat related to the notion of *read-once (Boolean) functions*: a function f is called *read-once* if it can be represented by a linear term (see, e.g., [6, 10]).

Proposition 31. *For any sets A and B of terms, $A \sqsupseteq B$ implies $A \sqsupseteq_\forall B$, and $A \sqsupseteq_\forall B$ implies $A \sqsupseteq_\exists B$; in other words, $\sqsupseteq \subset \sqsupseteq_\forall \subset \sqsupseteq_\exists$.*

Proof. The fact that the inclusions $\sqsupseteq \subseteq \sqsupseteq_\forall \subseteq \sqsupseteq_\exists$ hold is clear from the definition. It remains to show that these inclusions are indeed all strict.

To see that $\sqsupseteq$ is strictly included in $\sqsupseteq_\forall$, consider the NFSs $\mathbf{M}$ and $\mathbf{U}$. We can infer $\mathbf{M} \sqsupseteq_\exists \mathbf{U}$ from $\mu(x, y, z) \equiv \mathtt{u}(\mathtt{u}(x, 0, y), \mathtt{u}(x, y, z), 1)$, and thus $\mathbf{M} \sqsupseteq_\forall \mathbf{U}$ since μ is symmetric. However, $\mathbf{M} \sqsupseteq \mathbf{U}$ does not hold, as the following argument shows.

Suppose, to the contrary, that there is a ternary linear term $t \in \mathbf{U}$ equivalent to $\mu(x_1, x_2, x_3)$, and assume that t has the smallest possible size among such terms. By minimality, no subterm of t of the form $\mathtt{u}(t_1, t_2, t_3)$ satisfies $t_1 = 1$ or $t_2 = 1$ or $t_3 = 0$, because for any terms s, s', we have $\mathtt{u}(1, s, s') \equiv \mathtt{u}(s, 1, s') \equiv s'$ and $\mathtt{u}(s, s', 0) \equiv 0$; hence we could obtain a smaller term equivalent to t by replacing the subterm $\mathtt{u}(t_1, t_2, t_3)$ by t_3 or 0 accordingly. A similar argument shows that no subterm of t of the form $\mathtt{u}(t_1, t_2, t_3)$ satisfies $t_1 = t_2 = 0$, or $t_1 = 0$ and $t_3 = 1$, or $t_2 = 0$ and $t_3 = 1$, because for any term s, we have $\mathtt{u}(0, 0, s) \equiv 0$, $\mathtt{u}(0, s, 1) \equiv \mathtt{u}(s, 0, 1) \equiv s$. Consequently, in every subterm of the form $\mathtt{u}(t_1, t_2, t_3)$, at most one of the terms t_1, t_2, t_3 is a constant.

There are no two subterms $\mathtt{u}(t_1, t_2, t_3)$, $\mathtt{u}(s_1, s_2, s_3)$ of t such that two of the terms t_1, t_2, t_3 are variables and two of the terms s_1, s_2, s_3 are variables,

because then some variable appears at least twice in t (recall that t is ternary), contradicting the linearity of t. Consequently, there is no subterm of the form $\mathtt{u}(t_1, t_2, t_3)$ with $t_{i_1} = \mathtt{u}(t_{i_1,1}, t_{i_1,2}, t_{i_1,3})$, $t_{i_2} = \mathtt{u}(t_{i_2,1}, t_{i_2,2}, t_{i_2,3})$, $i_1 \neq i_2$. Otherwise t_{i_1} and t_{i_2} would have subterms of the form $\mathtt{u}(s_1, s_2, s_3)$ where the s_i are constants or variables, and, as we have seen above, these subterms must have at least two variables each, which contradicts again the linearity of t.

It is clear that no term in $\mathbf{U}$ with at most one occurrence of $\mathtt{u}$ is equivalent to $\mu(x_1, x_2, x_3)$. The only remaining possibility is that $t = \mathtt{u}(t_1, t_2, t_3)$ where one of the terms t_1, t_2, t_3 is a constant, one is a variable, and the remaining one is of the form $\mathtt{u}(s_1, s_2, s_3)$ where one of the terms s_1, s_2, s_3 is a constant and the other two are variables, so that all three variables appear and the conditions established above for the constants are satisfied. For such terms, the following equivalences hold (here $\{i, j, k\} = \{1, 2, 3\}$):

$$
\begin{aligned}
&\mathtt{u}(\mathtt{u}(x_i, x_j, 1), x_k, 1) \equiv \mathtt{u}(x_i, \mathtt{u}(x_j, x_k, 1), 1) \equiv x_1 \vee x_2 \vee x_3,\\
&\mathtt{u}(\mathtt{u}(x_i, x_j, 1), 0, x_k) \equiv \mathtt{u}(0, \mathtt{u}(x_i, x_j, 1), x_k) \equiv (x_i \vee x_j) \wedge x_k \equiv \mathtt{u}(x_i, x_j, x_k),\\
&\mathtt{u}(\mathtt{u}(x_i, 0, x_j), x_k, 1) \equiv \mathtt{u}(\mathtt{u}(0, x_i, x_j), x_k, 1) \equiv (x_i \wedge x_j) \vee x_k \equiv \mathtt{w}(x_i, x_j, x_k),\\
&\mathtt{u}(\mathtt{u}(x_i, 0, x_j), 0, x_k) \equiv \mathtt{u}(\mathtt{u}(0, x_i, x_j), 0, x_k) \equiv \mathtt{u}(0, \mathtt{u}(x_i, 0, x_j), x_k)\\
&\qquad\qquad\qquad\quad \equiv \mathtt{u}(0, \mathtt{u}(0, x_i, x_j), x_k) \equiv x_1 \wedge x_2 \wedge x_3.
\end{aligned}
$$

Clearly none of the above is equivalent to $\mu(x_1, x_2, x_3)$. We have reached a contradiction, and we conclude that $\mathbf{M} \not\supseteq \mathbf{U}$, as claimed.

Now, to see that $\sqsupseteq_\forall$ is strictly included in $\sqsupseteq_\exists$, consider the NFS $T(\sigma)$ where $\sigma := [(x_1 \wedge x_2) \vee (\neg x_1 \wedge x_3)]$. Clearly, $T(\sigma) \sqsupseteq_\exists \mathbf{M}$ since $\sigma(x_1, x_2, x_3) \equiv \mu(\mu(x_1, x_2, 0), \mu(\neg x_1, x_3, 0), 1)$. However, it follows from Corollary 60 that $T(\sigma) \sqsupseteq_\forall \mathbf{M}$ does not hold. □

Lemma 32. *If $\alpha \notin M$, then there exists a unary linear term $t \in T(\alpha)$ such that $|t|_\alpha = 1$ and $t \equiv \neg x_1$.*

Proof. Assume α is n-ary. Since α is not monotone, there exist tuples $(a_1, \ldots, a_n), (b_1, \ldots, b_n) \in \mathbb{B}^n$ such that

$$(a_1, \ldots, a_n) \leq (b_1, \ldots, b_n) \quad \text{and} \quad \alpha(a_1, \ldots, a_n) > \alpha(b_1, \ldots, b_n).$$

Consider the sequence $\mathbf{d}_0, \mathbf{d}_1, \ldots, \mathbf{d}_n$, where

$$\begin{aligned}
\mathbf{d}_0 &:= (a_1, \ldots, a_n),\\
\mathbf{d}_1 &:= (b_1, a_2, \ldots, a_n),\\
&\vdots\\
\mathbf{d}_i &:= (b_1, \ldots, b_i, a_{i+1}, \ldots, a_n),\\
&\vdots\\
\mathbf{d}_n &:= (b_1, \ldots, b_n).
\end{aligned}$$

For all $j \in \{1, \ldots, n\}$, we have $\mathbf{d}_{j-1} \leq \mathbf{d}_j$ and $\mathbf{d}_{j-1}$ and $\mathbf{d}_j$ potentially differ only at the j-th component. Since $\alpha(\mathbf{d}_0) > \alpha(\mathbf{d}_n)$, there must exist an index $i \in \{1, \ldots, n\}$ such that $\alpha(\mathbf{d}_{i-1}) > \alpha(\mathbf{d}_i)$. It follows that $\alpha(b_1, \ldots, b_{i-1}, x_1, a_{i+1}, \ldots, a_n) \equiv \neg x_1$. □

4.3. Properties of (quasi-)linear reductions

In this subsection, we show that these reductions entail the preorder $\succeq$.

Proposition 33. *For any sets A and B of terms, $A \sqsupseteq B$ implies $A \succeq B$; in other words, $\sqsupseteq \subseteq \succeq$.*

Proof. Given sets of terms $A = T(\alpha\neg)$ (or $T(\alpha)$) and $B = T(\beta\neg)$) (or $(T(\beta))$ such that $A \sqsupseteq B$, there exists a linear term $t \in T(\beta)$ such that $\alpha(x_1, \ldots, x_{\mathtt{ar}(\alpha)}) \equiv t$. Then we can convert any term s of A into an equivalent term in B by replacing every occurrence of α by t; more precisely, by replacing each subterm of the form $\alpha(s_1, \ldots, s_{\mathtt{ar}(\alpha)})$ by $t\{s_1/x_1, \ldots, s_{\mathtt{ar}(\alpha)}/x_{\mathtt{ar}(\alpha)}\}$. Since t is linear, the size of the resulting term in B equals $|t| \cdot |s|$, that is, a constant multiple of $|s|$. □

Example 34. $\mathbf{M} \sqsupseteq \mathbf{M}_5$ holds, since $\mu(x, y, z) \equiv \mu_5(0, 1, x, y, z)$. For instance, using this equivalence we can convert the term $t_1 = \mu(\mu(x, y, z), u, v)$ into the term $t_2 = \mu_5(0, 1, \mu_5(0, 1, x, y, z), u, v)$. Furthermore, we have $|t_1| = |t_2|$. By Proposition 33, it follows that $\mathbf{M} \succeq \mathbf{M}_5$. ■

Proposition 35. *If $A = T(\alpha\neg)$ (or $T(\alpha)$) and $B = T(\beta\neg)$ (or $T(\beta)$) are two sets of terms such that $A \sqsupseteq_\forall B$ and $t_1, \ldots, t_{\mathtt{ar}(\alpha)}$ are terms satisfying the conditions of a universal quasi-linear reduction from A to B, then for any $f \in [A]$ it holds that $C_B(f) \leq nk(C_A(f))^q + 1$, where $n := \mathtt{ar}(\beta)$, $k := \max_i\{|t_i|_\beta\}$ and $q := \max_{i,j}\{|t_i|_{x_j}\}$. Consequently, $\sqsupseteq_\forall \subseteq \succeq$.*

Proof. Let A and B be two sets of terms such that $A \sqsupseteq_\forall B$ holds. That is:

$$\forall i, \exists t_i \in T(\beta),\ \alpha(x_1, \ldots, x_{\mathtt{ar}(\alpha)}) \equiv t_i \quad \text{and} \quad |t_i|_{x_i} = 1.$$

To prove that $A \succeq B$, we give a recursive and efficient way of converting a term in A into an equivalent term in B. We then prove that the size of the converted term is polynomial in the size of the original term in A.

We need to distinguish between the cases when $B = T(\beta\neg)$ and when $B = T(\beta)$. We consider first the case when $B = T(\beta\neg)$. Let s be a term in A. Recall that for a sequence of n integers $(r_i)_{i=1}^n$, $\mathtt{argmax}_i(r_i)$ is the smallest integer j such that for all $i \in \{1, \ldots, n\}$, $r_j \geq r_i$. We denote by $\textsc{conv}_{A\to B}(s)$ the term in B equivalent to s inductively defined as follows.

- If s is a variable or a constant, then $\textsc{conv}_{A\to B}(s) := s$;
- if $s = \neg t$, then $\textsc{conv}_{A\to B}(s) := s$;
- if $s = \alpha(s_1, \ldots, s_{\mathtt{ar}(\alpha)})$, then

  $$\textsc{conv}_{A\to B}(s) := t_\ell\{\textsc{conv}_{A\to B}(s_1)/x_1, \ldots, \textsc{conv}_{A\to B}(s_{\mathtt{ar}(\alpha)})/x_{\mathtt{ar}(\alpha)}\},$$

  with $\ell := \mathtt{argmax}_i(|\textsc{conv}_{A\to B}(s_i)|)$.

The idea behind this recursive conversion process is to avoid repeating a subterm of maximal size that has already been converted. As we will see, this is sufficient to ensure an efficient conversion. The fact that $\textsc{conv}_{A\to B}(s) \equiv s$ is assured by the stability of interpretations under substitution.

Let $k := \max_i\{|t_i|_\beta\}$ and $q := \max_{i,j}\{|t_i|_{x_j}\}$.

Let s be a term of A that represents a Boolean function f. We will prove by induction on the structure of terms of A that $|\textsc{conv}_{A\to B}(s)| \leq k|s|^q$.

- Suppose that s is a literal or a constant. Then $|\textsc{conv}_{A\to B}(s)| = 0 = |s| = k|s|^q$.
- Suppose that $s = \alpha(s_1, s_2, \ldots, s_{\mathtt{ar}(\alpha)})$ with $s_i \in T(\alpha)$ for all i. Recall

that ℓ is defined by $\ell = \mathtt{argmax}_i(|\textsc{conv}_{A\to B}(s_i)|)$. Thus:

$$|\textsc{conv}_{A\to B}(s)| = |t_\ell\{\textsc{conv}_{A\to B}(s_i)/x_i\}| = |t_\ell|_\beta + \sum_{j=1}^{\mathtt{ar}(\alpha)} |t_\ell|_{x_j}|\textsc{conv}_{A\to B}(s_j)|$$

$$\leq k + q \sum_{j=1,\ j\neq\ell}^{\mathtt{ar}(\alpha)} |\textsc{conv}_{A\to B}(s_j)| + |\textsc{conv}_{A\to B}(s_\ell)|$$

$$\leq k + |\textsc{conv}_{A\to B}(s_1)| + q \sum_{j=2}^{\mathtt{ar}(\alpha)} |\textsc{conv}_{A\to B}(s_j)|$$

The penultimate inequality holds since $|t_\ell|_{x_\ell} = 1$, $|t_i|_{x_j} \leq q$ and $|t_\ell|_\beta \leq k$, and the last inequality holds because $|\textsc{conv}_{A\to B}(s_1)| \leq |\textsc{conv}_{A\to B}(s_\ell)|$ by the definition of ℓ, whence it follows that

$$q|\textsc{conv}_{A\to B}(s_1)|+|\textsc{conv}_{A\to B}(s_\ell)| \leq |\textsc{conv}_{A\to B}(s_1)|+q|\textsc{conv}_{A\to B}(s_\ell)|.$$

Now suppose without loss of generality that $|s_1| \geq |s_2| \geq \cdots \geq |s_{n-1}| \geq |s_{\mathtt{ar}(\alpha)}|$. Hence

$$|\textsc{conv}_{A\to B}(s)| \leq k(1 + |s_1|^q + q \sum_{i=2}^{\mathtt{ar}(\alpha)} |s_i|^q) \quad \text{by induction hypothesis.}$$

$$\leq k(1 + |s_1| + |s_2| + \cdots + |s_{\mathtt{ar}(\alpha)}|)^q = k|s|^q.$$

The last inequality holds because of the fact that $|s_{i+1}|^q \leq |s_{i+1}|^{q-1}|s_i|$.

Let $f \in [A] \subseteq [B]$ and let s be a term of minimal size in A that represents f. Then we have $C_A(f) = |s|$. We also have $C_B(f) \leq |\textsc{conv}_{A\to B}(s)|$. Since $|\textsc{conv}_{A\to B}(s)| \leq k|s|^q$, we have: $C_B(f) \leq |\textsc{conv}_{A\to B}(s)| \leq k|s|^q = k(C_A(f))^q$, and the claimed inequality clearly holds. Thus, $A \succeq B$.

We now consider the case when $B = T(\beta)$. If $A = T(\alpha)$, then the conversion from A to B described above works as such, and the same polynomial upper bound for the size of the converted term as above holds. If $A = T(\alpha\neg)$, then we need a way of dealing with the negations that may appear in a term $s \in T(\alpha\neg)$. In this case, β must be non-monotone, so by Lemma 32, there exists a unary linear term $t \in T(\beta)$ such that $|t|_\beta = 1$ and $t \equiv \neg x_1$. Let $s' := \textsc{conv}_{A\to T(\beta\neg)}(s)$ be the conversion of s into an equivalent

term in $T(\beta\neg)$ as described above, and let s'' be the term obtained from s' by replacing each subterm of the form $\neg u$ by $t\{u/x_1\}$. Then clearly $s'' \in T(\beta)$ and $s'' \equiv s$. If s is a term of the smallest possible size in A representing a function f, then there are no iterated negations in s, and the number of negations in s' is at most $(n-1)|s'|+1$, where $n := \mathtt{ar}(\beta)$. Since each negation of s' gets replaced by a term with a single occurrence of β, we have that $C_B(f) \leq |s''| \leq |s'|+(n-1)|s'|+1 = n|s'|+1 \leq nk|s|^q+1 = nk(C_A(f))^q+1$. Thus, $A \succeq B$ also in this case. □

Example 36. For the connective μ, the following equivalences hold:

$$\mu(x,y,z) \equiv (y{\uparrow}z){\uparrow}(x{\uparrow}((y{\uparrow}1){\uparrow}(z{\uparrow}1))),$$
$$\mu(x,y,z) \equiv (x{\uparrow}z){\uparrow}(y{\uparrow}((x{\uparrow}1){\uparrow}(z{\uparrow}1))),$$
$$\mu(x,y,z) \equiv (y{\uparrow}x){\uparrow}(z{\uparrow}((y{\uparrow}1){\uparrow}(x{\uparrow}1))).$$

As each equivalence is linear in one variable (x, y and z respectively), $\mathbf{M} \sqsupseteq_\forall \mathbf{S}$ holds. By Proposition 35, we deduce that $\mathbf{M} \succeq \mathbf{S}$. ■

We can handle the case of existential quasi-linear reduction if one of the connectives is a symmetric function.

Proposition 37. *Consider two sets of stratified terms $A = T(\alpha\neg)$ and $B = T(\beta\neg)$. If $A \sqsupseteq_\exists B$ and α is a symmetric function, then $A \sqsupseteq_\forall B$; consequently, $A \succeq B$.*

Proof. The symmetry of α allows us to exhibit a universal quasi-linear reduction from the reduction $A \sqsupseteq_\exists B$. We can then apply Proposition 35. □

4.4. Equivalences between basic Sheffer and basic quasi-Sheffer NFSs

As an application of linear reducibility, we show that basic Sheffer and basic quasi-Sheffer NFSs are all equivalent to **M** and, consequently, strictly more efficient than other basic non-Sheffer and non-quasi-Sheffer NFSs. For that purpose, we adapt the median decomposition scheme of [14] to terms.

Proposition 38. *The basic NFSs* **U**, **W**, *and* **M** *are pairwise equivalent, i.e.,* $\mathbf{U} \sim \mathbf{W} \sim \mathbf{M}$.

Proof. Consider the equivalences

$$\mathtt{u}(x,y,z) \equiv \mu(\mu(x,1,y),0,z)$$

and

$$\mu(x,y,z) \equiv \mathtt{u}(\mathtt{u}(x,0,y),\mathtt{u}(x,y,z),1).$$

We have $|\mu(\mu(x,1,y),0,z)|_w = 1$ for all $w \in \{x,y,z\}$ and $|\mathtt{u}(\mathtt{u}(x,0,y),\mathtt{u}(x,y,z),1)|_z = 1$. Consequently, $\mathbf{U} \sqsupseteq \mathbf{M}$ and $\mathbf{M} \sqsupseteq_\exists \mathbf{U}$ hold. Corollary 33 and Proposition 37 and the symmetry of μ imply $\mathbf{U} \sim \mathbf{M}$. A dual reasoning can be used to prove $\mathbf{W} \sim \mathbf{M}$. □

Proposition 39. *The basic NFSs* $\mathbf{S}$, $\mathbf{S}^{\mathrm{d}}$, *and* $\mathbf{M}$ *are pairwise equivalent, i.e.,* $\mathbf{S} \sim \mathbf{S}^{\mathrm{d}} \sim \mathbf{M}$.

Proof. Consider the equivalences

$$x{\uparrow}y \equiv \mu(\neg x, 1, \neg y)$$

and

$$\mu(x,y,z) \equiv (y{\uparrow}z){\uparrow}(x{\uparrow}((y{\uparrow}1){\uparrow}(z{\uparrow}1)))$$

(see Example 36). Remark that

$$|\mu(\neg x,1,\neg y)|_x = |\mu(\neg x,1,\neg y)|_y = 1,$$

and

$$|(y{\uparrow}z){\uparrow}(x{\uparrow}((y{\uparrow}1){\uparrow}(z{\uparrow}1)))|_x = 1.$$

Thus, $\mathbf{S} \sqsupseteq_\exists \mathbf{M}$ and $\mathbf{M} \sqsupseteq_\exists \mathbf{S}$ both hold. Remark also that both μ and $\uparrow$ are symmetric functions. From Proposition 37 it then follows that $\mathbf{M} \sim \mathbf{S}$. A dual reasoning can be used to prove $\mathbf{S}^{\mathrm{d}} \sim \mathbf{M}$. □

Proposition 40 (Median decomposition scheme [14, Theorem 17])**.** *Let* α *be a monotone Boolean function. Then for any* $k \in \{1,\dots,\mathtt{ar}(\alpha)\}$*:*

$$\alpha(x_1,\dots,x_{\mathtt{ar}(\alpha)}) \equiv \mu(\alpha(x_1,\dots,x_{\mathtt{ar}(\alpha)})\{0/x_k\}, x_k, \alpha(x_1,\dots,x_{\mathtt{ar}(\alpha)})\{1/x_k\}).$$

The term on the right side of the above equivalence is called a *median decomposition* of α with respect to the *pivot variable* x_k.

The subterms $\alpha(x_1,\dots,x_{\mathtt{ar}(\alpha)})\{c/x_k\}$, $c \in \mathbb{B}$, appearing in the median decomposition induce monotone functions, provided that α is monotone. Applying the median decomposition scheme recursively on the subterms $\alpha(x_1,\dots,x_{\mathtt{ar}(\alpha)})\{c/x_k\}$, $c \in \mathbb{B}$, selecting always a new pivot variable, produces a term in $\mathbf{M}$ representing α. Note that the firstly chosen pivot variable appears only once in this term.

This method of constructing a median representation of a monotone Boolean function was presented as an algorithm in [7]. The algorithm was then adapted for arbitrary Boolean functions by considering an encoding of a non-monotone function as a monotone function having twice as many variables.

Example 41. The function $\alpha := [(x \wedge y) \wedge z]$ is monotone. By the median decomposition scheme applied to x, the equivalence

$$\alpha(x, y, z) \equiv \mu(\alpha(0, y, z), x, \alpha(1, y, z))$$

holds. After decomposing the remaining subterms in α with respect to y and z, we obtain the conversion equivalence

$$\alpha(x, y, z) \equiv \mu(\mu(\mu(0, z, 0), y, \mu(0, z, 0)), x, \mu(\mu(0, z, 0), y, \mu(0, z, 1)))$$

in which x only occurs once[1]. ∎

Proposition 42. *For all $n \geq 1$, the basic NFSs $\mathbf{M}_{2n+1}$ and $\mathbf{M}$ are equivalent, i.e., $\mathbf{M}_{2n+1} \sim \mathbf{M}$.*

Proof. From the median decomposition scheme and the fact that μ_{2n+1} is a monotone and symmetric function, it follows by Proposition 37 that $\mathbf{M} \preceq \mathbf{M}_{2n+1}$. By Proposition 37 again and the equivalence $\mu(x, y, z) \equiv \mu_{2n+1}(z, x^n, y^n)$, $\mathbf{M}_{2n+1} \preceq \mathbf{M}$. □

Propositions 38, 39 and 42, together with Theorem 27, give us the classification of Figure 2 for basic NFSs.

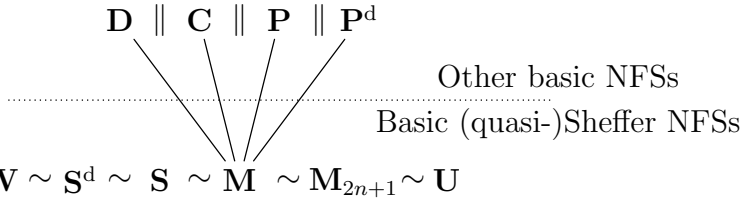

Figure 2: Semilattice of basic NFSs ordered by $\preceq$.

[1]Remark that the right hand side of this equivalence can be simplified further into $\alpha(x, y, z) \equiv \mu(0, x, \mu(0, y, z))$.

5. Optimality for monotonic NFSs

In this section we will show that the optimal monotonic NFSs are exactly those that use a single connective or one connective with negation. For this, we first show that the median NFS is optimal among monotonic NFS (Theorem 44). Then, by making good use of reductions between NFSs, we will show by case analysis that every monotonic NFS based on a single connective and the negation is at least as efficient as the median NFS.

5.1. Optimality of the median normal form **M**

In this subsection, we extend the results of the previous section by showing that **M** is *optimal* among monotonic NFSs.

Proposition 43. *For any pseudo-monotone α, $\mathbf{M} \preceq T(\alpha)$ holds.*

Proof. Since α is pseudo-monotone, there exists a monotone function g and literals $l_i \in \{x_i, \neg(x_i)\}$, $1 \leq i \leq n$ such that $\alpha(x_1, \ldots, x_n) \equiv g(l_1, \ldots, l_n)$. By applying the median decomposition scheme on g, choosing the first pivot variable in different ways, we can produce n terms $t_i \in T(\mu)$ such that $t_i \equiv g(x_1, \ldots, x_n)$ and $|t_i|_{x_i} = 1$. Define $t'_i := t_i\{l_1/x_1, \ldots, l_n/x_n\}$; we have $t'_i \in T(\mu\neg)$, $t'_i \equiv \alpha(x_1, \ldots, x_n)$ and $|t'_i|_{x_i} = 1$. Using the self-duality of μ to propagate negations on variables, it is not difficult to see that $T(\alpha) \sqsupseteq_\forall T(\mu\neg) = \mathbf{M}$. By Proposition 35, we obtain that $\mathbf{M} \preceq T(\alpha)$. □

Theorem 44. **M** *is optimal among monotonic NFSs.*

Proof. We first show by induction on the number n of connectives that for any irredundant set of terms $T(\alpha_1 \cdots \alpha_n)$, where each α_i is a pseudo-monotone function, the inequality $\mathbf{M} \preceq T(\alpha_1 \cdots \alpha_n)$ holds.

If $n = 1$, then the result holds by Proposition 43.

Suppose that the induction hypothesis holds for any natural number smaller than n. We will show that it holds for n. Set $T_i := T(\alpha_1 \cdots \alpha_i)$, for $i \leq n$.

For a given function f, consider a term $t \in T_n$ such that $C_{T_n}(f) = |t|$, and $[t] \cong f$. The term t can be written as $t'\{t_1/x_1, \ldots, t_j/x_j\}$ where t' is a j-ary term in T_{n-1}, for some integer j, and $t_i \in T(\alpha_n)$ for all $i \leq j$.

We have $C_{T_{n-1}}([t']) = |t'|$ and $C_{T(\alpha_n)}([t_i]) = |t_i|$ by the minimality of t. Otherwise it contradicts the fact that t is a term of minimal size in T_n representing f. Moreover, $C_{T_n}([t]) = C_{T_{n-1}}([t']) + \Sigma_{i=1}^{j} C_{T(\alpha_n)}([t_i])$ by irredundancy.

By the induction hypothesis, $\mathbf{M} \preceq T_{n-1}$. Consequently, there exists a polynomial P such that $C_{\mathbf{M}}([t']) \leq P(C_{T_{n-1}}([t']))$. As the size of the minimal term in $\mathbf{M}$ equivalent to t' is bounded by $P(C_{T_{n-1}}([t']))$, we know that it contains no more than $3P(C_{T_{n-1}}([t']))$ leaves since μ is a ternary connective. Consequently, $j \leq 3P(C_{T_{n-1}}([t']))$.

By Proposition 43, there exists a polynomial Q such that for all i, $1 \leq i \leq j$, $C_{\mathbf{M}}([t_i]) \leq Q(C_{T(\alpha_n)}([t_i]))$. Consequently, we have:

$$\begin{aligned} C_{\mathbf{M}}([t]) &\leq C_{\mathbf{M}}([t']) + \Sigma_{i=1}^{j} C_{\mathbf{M}}([t_i]) \\ &\leq P(C_{T_{n-1}}([t'])) + 3P(C_{T_{n-1}}([t'])) \max_i Q(C_{T(\alpha_n)}([t_i])) \\ &\leq P(C_{T_n}([t])) + 3P(C_{T_n}([t]))Q(C_{T_n}([t])) \\ &= R(C_{T_n}([t])) \end{aligned}$$

with $R = P + 3P \cdot Q$. In the above, the last inequality holds because $C_{T_{n-1}}([t']) \leq C_{T_n}([t])$, $C_{T(\alpha_n)}([t_i]) \leq C_{T_n}([t])$ for all i, and P and Q are polynomials with nonnegative coefficients and hence the polynomial functions induced by P and Q are monotone increasing.

We have shown that $\mathbf{M} \preceq T(\alpha_1 \cdots \alpha_n)$ holds for any irredundant set of terms $T(\alpha_1 \cdots \alpha_n)$ with pseudo-monotone connectives $\alpha_1, \ldots, \alpha_n$. Consequently, it holds for any monotonic NFS. □

5.2. Optimality of monotonic Sheffer and quasi-Sheffer NFSs

In this subsection, we generalize the results obtained in Subsection 4.4 by showing that any monotonic Sheffer or quasi-Sheffer NFS is optimal.

Lemma 45. *For any NFS $T(\alpha)$, we have $T(\alpha) \sim T(\alpha\neg)$.*

Proof. Consider an NFS $T(\alpha)$. Clearly, $T(\alpha\neg) \preceq T(\alpha)$ is immediate because a term in $T(\alpha)$ is also a term in $T(\alpha\neg)$. By Fact 16, $[T(\alpha)] = \mathcal{C}(\alpha) \circ I$. The function α is necessarily non-monotone and non-constant. (For, if $\alpha \in M$, then $[T(\alpha)] = \mathcal{C}(\alpha) \circ I \subseteq M \circ I = M \subsetneq \Omega$ and $T(\alpha)$ is not an NFS.) Thus, there exist constants $c_1, \ldots, c_{\mathtt{ar}(\alpha)-1}$ and a permutation π such that $\alpha(\pi(c_1, \ldots, c_{\mathtt{ar}(\alpha)-1}, x)) \equiv \neg x$. This highlights a reduction $T(\alpha\neg) \sqsupseteq T(\alpha)$. □

Theorem 46. *All monotonic Sheffer and quasi-Sheffer NFSs are optimal.*

Proof. By Lemma 45, it suffices to consider sets of terms of the form $T(\alpha\neg)$ such that $[T(\alpha\neg)] = \Omega$ and to show that $T(\alpha\neg) \sim \mathbf{M}$ holds. By Theorem 44,

$\mathbf{M} \preceq T(\alpha\neg)$ holds. It remains to show that $T(\alpha\neg) \preceq \mathbf{M}$ also holds. This inequality directly depends on the nature of the function α. By Fact 16, we need to consider functions α satisfying $\Omega = [T(\alpha\neg)] = \mathcal{C}(\alpha) \circ \mathcal{C}(\neg) \circ I = \mathcal{C}(\alpha) \circ \Omega(1)$. The clones $\mathcal{C}$ satisfying $\mathcal{C} \circ \Omega(1) = \Omega$ can be read off from the table of clone composition in [5]; they are the following: Ω, T_0, T_1, T_c, M, M_0, M_1, M_c, S, S_c, SM, and for $k = 2, \ldots, \infty$, U_k, MU_k, $T_\mathrm{c}U_k$, $M_\mathrm{c}U_k$, W_k, MW_k, $T_\mathrm{c}W_k$, $M_\mathrm{c}W_k$. Thus, we need to consider functions α that generate one of the clones listed above. Note that the clones M, M_0, M_1, MU_k, MW_k are not generated by a single function, so these need not be considered. In the following subsection, we are going to establish for each one of the relevant clones $\mathcal{C}$ a proposition of the form: *If* $\mathcal{C}(\alpha) = \mathcal{C}$, *then* $T(\alpha\neg) \preceq \mathbf{M}$. More explicitly, the clones and the respective propositions are the following: SM (Proposition 48), $M_\mathrm{c}U_k$, $M_\mathrm{c}W_k$, for $k = 2, \ldots, \infty$ (Proposition 50), M_c (Proposition 52), U_k, $T_\mathrm{c}U_k$, W_k, $T_\mathrm{c}W_k$, for $k = 2, \ldots, \infty$ (Proposition 53), Ω, T_0, T_1, T_c, S, S_c (Proposition 55). Putting all these propositions together, the current theorem follows. □

5.3. Proof of optimality

In this subsection, we will use the following notation. Given a tuple $\mathbf{x}$ and a permutation π, $\pi(\mathbf{x})$ indicates the tuple obtained by permuting the coordinates of $\mathbf{x}$ following π. For $b \in \mathbb{B}$ and an integer $k > 0$, let b^k be a shorthand notation for $b, \ldots, b$ with k occurrences of b.

5.3.1. The clone SM

Lemma 47. *Let $f \in SM$ be a function of arity $n \geq 2$ that is not a projection, and let $\mathbf{x}$ be a minimal true point of f. Then there exists a true point $\mathbf{y}$ such that $d(\mathbf{x}, \mathbf{y}) = n - 1$. Furthermore, the unique common coordinate of $\mathbf{x}$ and $\mathbf{y}$ has value 1.*

Proof. Let $\mathbf{x}$ be a minimal true point of the function $f : \mathbb{B}^n \to \mathbb{B}$ in SM. If $w(\mathbf{x}) = 0$, then f is constant (equal to 1) by monotonicity; this case does not occur, because SM does not contain any constant function. If $w(\mathbf{x}) = 1$, then f is a projection by monotonicity and self-duality. We can thus assume that $w(\mathbf{x}) \geq 2$, and, without loss of generality, $\mathbf{x} = (1, 1^k, 0^l)$ with $k > 0$ and $k + l + 1 = n$. $\mathbf{z} := (0, 1^k, 0^l)$ is a false point because $\mathbf{x}$ is a minimal true point. Thus $\mathbf{y} := \overline{\mathbf{z}} = (1, 0^k, 1^l)$ is a true point by self-duality. Also, we have $d(\mathbf{x}, \mathbf{y}) = k + l = n - 1$, and the common coordinate of $\mathbf{x}$ and $\mathbf{y}$ is 1. □

Proposition 48. *If α is a connective such that $\mathcal{C}(\alpha) = SM$, then $T(\alpha\neg) \preceq \mathbf{M}$.*

Proof. Note that every self-dual monotone function is either a projection or has at least three essential arguments. Since α generates the clone SM, it cannot be a projection; hence $\mathtt{ar}(\alpha) \geq 3$. Since α is monotone and self-dual, we can separate its true and false points in two sets of same size (self-duality) and such that no true point of α is covered by a false point of α (monotonicity).

By Lemma 47, there exist two true points $\mathbf{x}, \mathbf{y}$ with $\mathbf{x}$ minimal, at distance $\mathtt{ar}(\alpha) - 1$: there exists exactly one coordinate in which both are equal to 1. There exists a permutation π such that $\pi(\mathbf{x}) = (1, 1^k, 0^l)$ and $\pi(\mathbf{y}) = (1, 0^k, 1^l)$, for some positive integers k, l such that $1 + k + l = \mathtt{ar}(\alpha)$.

Let α' be the ternary function

$$\alpha' := [\alpha(\pi^{-1}(x_1, x_2^k, x_3^l))].$$

Since α' is obtained from α by composing with projections, we have $\alpha' \in SM$.

We are going to show that $\alpha' = \mu$. We have $\alpha'(1,1,0) = \alpha(\mathbf{x}) = 1$ and $\alpha'(1,0,1) = \alpha(\mathbf{y}) = 1$. Since $\mathbf{x}$ is a minimal true point of α and $\pi^{-1}(1,0,0) < \mathbf{x}$, we have $\alpha'(1,0,0) = \alpha(\pi^{-1}(1,0,0)) = 0$. By the self-duality of α' we obtain $\alpha'(0,0,1) = \alpha'(0,1,0) = 0$, $\alpha'(0,1,1) = 1$, and by the monotonicity of α' we get $\alpha'(0,0,0) = 0$, $\alpha'(1,1,1) = 1$. Thus $\alpha' = \mu$. Recall that the median μ is symmetric, which yields the following "partial" symmetry for α:

$$\alpha(x_1, x_2^k, x_3^l) \equiv \alpha(x_2, x_1^k, x_3^l) \equiv \alpha(x_3, x_1^k, x_2^l).$$

This means that $\mathbf{M} \sqsupseteq_\forall T(\alpha\neg)$ holds, with $t_1 = \alpha(x_1, x_2^k, x_3^l)$, $t_2 = \alpha(x_2, x_1^k, x_3^l)$, and $t_3 = \alpha(x_3, x_1^k, x_2^l)$. Thus by Proposition 35 we obtain $T(\alpha\neg) \preceq \mathbf{M}$. □

5.3.2. The clones M_cU_k and M_cW_k

Lemma 49. *For any clone $\mathcal{C}$ with $\Lambda \subsetneq \mathcal{C} \subseteq M$, any generator of $\mathcal{C}$ has at least two minimal true points.*

Proof. Monotone functions with fewer than two minimal true points are constants, projections, or conjunctions of variables and hence cannot generate $\mathcal{C}$. □

Recall that $\mathtt{u}$ and $\mathtt{w}$ are generators of minimal arity of M_cU_∞ and M_cW_∞, respectively.

Proposition 50. *If α is a connective such that $\mathcal{C}(\alpha) = M_c U_k$ or $\mathcal{C}(\alpha) = M_c W_k$ for some $k \in \{2, \ldots, \infty\}$, then $T(\alpha\neg) \preceq \mathbf{M}$.*

Proof. We study the case when $\mathcal{C}(\alpha) = M_c U_k$ for some $k \in \{2, \ldots, \infty\}$. The dual case $M_c W_k$ can be proved similarly. First, recall from Proposition 38 that $\mathbf{U} \sim \mathbf{W} \sim \mathbf{M}$. Therefore, it will suffice to show that $T(\alpha\neg) \preceq \mathbf{U}$ or $T(\alpha\neg) \preceq \mathbf{W}$ or $T(\alpha\neg) \preceq \mathbf{M}$.

By Lemma 49, there exist two minimal incomparable true points $\mathbf{x}, \mathbf{y}$ for α. Since $\alpha \in U_2$, they have a coordinate in common with the value 1. Permuting arguments if necessary, we may assume without loss of generality that $\mathbf{x} = (1^k, 1^l, 1, 0^m, 0^n)$ and $\mathbf{y} = (0^k, 1^l, 1, 1^m, 0^n)$ for some integers k, l, m, n such that $k > 0$, $m > 0$, $l \geq 0$, $n \geq 0$ and $1 + k + l + m + n = \mathtt{ar}(\alpha)$.

Let α' be the ternary function defined by

$$\alpha' := [\alpha(x_1, 1^{k+l-1}, x_3, x_2^m, 0^n)].$$

Note that α' is monotone, because it is obtained from the monotone function α by identifying variables and substituting constants for variables, in other words, $\alpha' \in \mathcal{C}(\alpha) \circ I \subseteq M \circ I = M$.

We will show that $\alpha' \in \{\mathtt{u}, \mu, \mathtt{w}\}$. We can deduce from the information we have thus far that the function α' satisfies the following (see the leftmost Hasse diagram in Figure 3):

- $\alpha'(1,0,1) = \alpha(\mathbf{x}) = 1$. By monotonicity, $\alpha'(1,1,1) = 1$. Moreover, since $\mathbf{x}$ is a minimal true point of α, it follows that $(1,0,1)$ is a minimal true point of α'; hence $\alpha'(0,0,1) = \alpha'(1,0,0) = \alpha'(0,0,0) = 0$.
- $\alpha'(0,1,1) = \alpha(\mathbf{y}')$, where $\mathbf{y}' = (0, 1^{k-1}, 1^l, 1, 1^m, 0^n)$. Since $\mathbf{y}' > \mathbf{y}$ and $\mathbf{y}$ is a (minimal) true point of α, it follows that $\alpha'(0,1,1) = \alpha(\mathbf{y}') = 1$.

The values of α' at $(0,1,0)$ and $(1,1,0)$ remain undetermined, but the monotonicity of α' gives $\alpha'(0,1,0) \leq \alpha'(1,1,0)$. This leaves us with three possibilities (see the Hasse diagrams in Figure 3):

- $\alpha'(0,1,0) = \alpha'(1,1,0) = 0$, in which case $\alpha' = \mathtt{u}$;
- $\alpha'(0,1,0) = 0$, $\alpha'(1,1,0) = 1$, in which case $\alpha' = \mu$;
- $\alpha'(0,1,0) = \alpha'(1,1,0) = 1$, in which case $\alpha' = \mathtt{w}$.

Let us consider the consequences of the three different possibilities for α'.

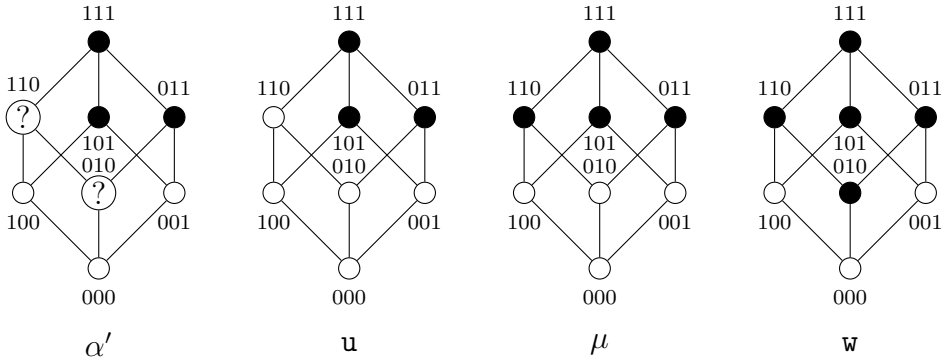

Figure 3: Hasse diagrams of the different possibilities for α'.

1. If $\alpha' = \mathtt{u}$, then the equivalence $\mathtt{u}(x_1, x_2, x_3) \equiv \alpha(x_1, 1^{k+l-1}, x_3, x_2^m, 0^n)$ holds. Recall that $\mathtt{u}(x_1, x_2, x_3) \equiv (x_1 \vee x_2) \wedge x_3$, which shows, by the symmetry of $\vee$, that $\mathtt{u}$ is invariant under the transposition of the first two arguments, that is,

$$\mathtt{u}(x_1, x_2, x_3) \equiv \mathtt{u}(x_2, x_1, x_3),$$

which yields the following equivalence:

$$\mathtt{u}(x_1, x_2, x_3) \equiv \alpha(x_2, 1^{k+l-1}, x_3, x_1^m, 0^n).$$

We have found for each variable x_i ($i \in \{1, 2, 3\}$) a term in $T(\alpha\neg)$ that is equivalent to $\mathtt{u}(x_1, x_2, x_3)$ and has only one occurrence of x_i. Consequently, $\mathbf{U} \sqsupseteq_\forall T(\alpha\neg)$ holds. Thus by Proposition 35 we obtain $T(\alpha\neg) \preceq \mathbf{U}$.

2. If $\alpha' = \mu$, then $\mu(x_1, x_2, x_3) \equiv \alpha(x_1, 1^{k+l-1}, x_3, x_2^m, 0^n)$. Since μ is a symmetric function, $T(\mu\neg) \sqsupseteq_\forall T(\alpha\neg)$ holds by Proposition 37, and $T(\alpha\neg) \preceq \mathbf{M}$.

3. If $\alpha' = \mathtt{w}$, then following the same reasoning as above in case 1 (recall that $\mathtt{w}(x_1, x_2, x_3) \equiv (x_1 \wedge x_2) \vee x_3$) we obtain that $T(\alpha\neg) \preceq \mathbf{W}$. □

5.3.3. The clone M_c

Lemma 51. *Let α be a connective such that $\mathcal{C}(\alpha) = M_c$. Then there exists a binary linear term $t \in T(\alpha)$ such that $t \equiv x_1 \wedge x_2$.*

Proof. Let α be an n-ary connective such that $\mathcal{C}(\alpha) = M_{\mathrm{c}}$. Let $\mathbf{a}$ be a minimal true point of α. We must have $w(\mathbf{a}) \geq 2$ (for otherwise α would be a projection or a constant and hence not a generator of M_{c}). Then there exist indices $p, q \in \{1, \ldots, n\}$ with $p \neq q$ such that $a_p = a_q = 1$. For $i \in \{1, \ldots, n\}$, let

$$t_i := \begin{cases} a_i & \text{if } i \notin \{p, q\}, \\ x_1 & \text{if } i = p, \\ x_2 & \text{if } i = q. \end{cases}$$

Then $\alpha\{t_1/x_1, \ldots, t_n/x_n\}$ is a binary linear term in $T(\alpha)$ that is equivalent to $x_1 \wedge x_2$. □

Proposition 52. *If α is a connective such that $\mathcal{C}(\alpha) = M_{\mathrm{c}}$, then $T(\alpha\neg) \preceq \mathbf{M}$.*

Proof. Let α be an n-ary generator of the clone M_{c}. Consider the function α' of arity $n + 1$ defined by

$$\alpha' := [\alpha(x_1, \ldots, x_n) \wedge x_{n+1}].$$

Note that α' is monotone and constant-preserving, and for every true point $\mathbf{a}$ of α' we have $a_{n+1} = 1$; hence $\alpha' \in M_{\mathrm{c}}U_\infty$. Being a generator of M_{c}, the function α is not a member of Λ; consequently, we must have also that $\alpha' \notin \Lambda$. Therefore, α' is a generator of $M_{\mathrm{c}}U_\infty$, and it follows from Proposition 50 that $T(\alpha'\neg) \preceq \mathbf{M}$.

Let t be the binary linear term in $T(\alpha)$ equivalent to $x_1 \wedge x_2$ provided by Lemma 51, and let $t' := t\{\alpha(x_1, \ldots, x_n)/x_1, x_{n+1}/x_2\}$. Then t' is a linear term in $T(\alpha)$ and clearly $\alpha'(x_1, \ldots, x_{n+1}) \equiv t'$. Therefore, $T(\alpha'\neg) \sqsupseteq T(\alpha\neg)$, as witnessed by the term t'. By Proposition 33, we have $T(\alpha\neg) \preceq T(\alpha'\neg)$. It now follows from the transitivity of $\preceq$ that $T(\alpha\neg) \preceq \mathbf{M}$, as claimed. □

5.3.4. The clones U_k, $T_{\mathrm{c}}U_k$, W_k and $T_{\mathrm{c}}W_k$

Proposition 53. *If α is a pseudo-monotone function such that $\mathcal{C}(\alpha)$ equals U_k, $T_{\mathrm{c}}U_k$, W_k or $T_{\mathrm{c}}W_k$ for some $k \in \{2, \ldots, \infty\}$, then $T(\alpha\neg) \preceq \mathbf{M}$.*

Proof. Let α be an n-ary pseudo-monotone function, and assume first that $\mathcal{C}(\alpha) \in \{U_k, T_{\mathrm{c}}U_k\}$. As $\alpha \notin M$, Lemma 32 provides a unary linear term $u \in T(\alpha)$ with $|u|_\alpha = 1$ such that $u \equiv \neg x_1$.

Recall from Fact 1 that α is pseudo-monotone if and only if there exist a monotone function $g\colon \mathbb{B}^n \to \mathbb{B}$ and a subset $S \subseteq \{1, \ldots, n\}$ such that for

all $x_1, \ldots, x_n \in \mathbb{B}$, $\alpha(x_1, \ldots, x_n) = g(l_1, \ldots, l_n)$, where $l_i = x_i$ if $i \in S$ and $l_i = \neg(x_i)$ if $i \notin S$.

Then g is of the form $[\alpha(l'_1, \ldots, l'_n)]$, with $l'_j \in \{x_j, \neg x_j\}$, for all $j \in \{1, \ldots, n\}$. Therefore there exists a term $t \in T(\alpha)$ satisfying $|t|_{x_i} = 1$, for all i such that $1 \leq i \leq n$, and $g = [t]$.

Note that $g \notin V$. (Suppose, to the contrary, that $g \in V$. Then α is either a disjunction of negated variables or a disjunction of both negated and unnegated variables. By a suitable choice of $i_1, \ldots, i_n \in \{1, 2\}$, we obtain $\alpha(x_{i_1}, \ldots, x_{i_n}) \equiv \neg x_1 \vee \neg x_2 \equiv x_1 \uparrow x_2$ or $\alpha(x_{i_1}, \ldots, x_{i_n}) \equiv \neg x_1 \vee x_2$. Since $\mathcal{C}(\uparrow) = \Omega$ and $\mathcal{C}([\neg x_1 \vee x_2]) = W_\infty$, we have $W_\infty \subseteq \mathcal{C}(\alpha)$, which contradicts the initial assumption.) If g has at least 2 minimal true points, then one can see easily that there exist $i, j \in \{1, \ldots, n\}$, $i < j$, and constants $c_1, \ldots, c_n$ such that $x \vee y \equiv g(c_1, \ldots, c_{i-1}, x, c_{i+1}, \ldots, c_{j-1}, y, c_{j+1}, \ldots, c_n)$. If g has a unique minimal true point, then it is a conjunction of variables. Using De Morgan's laws and the above representations of $\neg$ and g in terms of α, we see that both $\wedge$ and $\vee$ can be obtained as the term function of some binary term $t \in T(\alpha)$ such that each variable occurs in t only once.

Let now $h := [g(x_1, \ldots, x_n) \vee x_{n+1}]$. Clearly, h is a generator of $M_{\mathrm{c}}W_\infty$ (because $g \notin V$) and, by construction, $T(h\neg)$ is linearly reducible to $T(\alpha\neg)$. By Proposition 50 it then follows that $T(\alpha\neg) \preceq \mathbf{M}$. The remaining cases when $\mathcal{C}(\alpha) \in \{W_k, T_{\mathrm{c}}W_k\}$ follow by a dual reasoning. □

5.3.5. The clones Ω, T_0, T_1, T_{c}, S and S_{c}

Lemma 54. *For any connective $\alpha \notin M \cup L$, there exists a binary linear term $t \in T(\alpha)$ such that $t \equiv x_1 \wedge x_2$.*

Proof. Let α be a connective such that $\alpha \notin M \cup L$. Let P_α be a polynomial normal form representation of α that is of smallest possible size. Then P_α is of the form $C_1 \oplus C_2 \oplus \ldots \oplus C_p$, where each subterm C_i is either a variable, constant, or a conjunction of variables. There must be at least one C_i that is a conjunction of at least two variables, or else α would be a linear function. Let C be the smallest (w.r.t. size) subterm among the C_i that is a conjunction of variables. Without loss of generality, we may assume that $C = x_1 \wedge x_2 \wedge \ldots \wedge x_k$.

Let now α' be the term obtained from $\alpha(x_1, \ldots, x_n)$ by substituting 1 for every occurrence of x_i, for $3 \leq i \leq k$, and by substituting 0 for every occurrence of x_j, for $j > k$. The resulting term α' is linear and equivalent to

one of the following:

$$\begin{aligned}
&x_1 \wedge x_2, \\
&(x_1 \wedge x_2) \oplus 1 && \equiv \neg x_1 \vee \neg x_2, \\
&(x_1 \wedge x_2) \oplus x_1 && \equiv x_1 \wedge \neg x_2, \\
&(x_1 \wedge x_2) \oplus (x_1 \oplus 1) && \equiv \neg x_1 \vee x_2, \\
&(x_1 \wedge x_2) \oplus x_2 && \equiv \neg x_1 \wedge x_2, \\
&(x_1 \wedge x_2) \oplus (x_2 \oplus 1) && \equiv x_1 \vee \neg x_2, \\
&(x_1 \wedge x_2) \oplus (x_1 \oplus x_2) && \equiv x_1 \vee x_2, \\
&(x_1 \wedge x_2) \oplus ((x_1 \oplus x_2) \oplus 1) && \equiv \neg x_1 \wedge \neg x_2.
\end{aligned}$$

It is now clear that the conjunction $x_1 \wedge x_2$ can be obtained from α' by negating variables or the entire term α'. Since α is not monotone, Lemma 32 provides a linear term $t \in T(\alpha)$ representing the negation. Using α' and t, we can now build a linear term in $T(\alpha)$ that is equivalent to $x_1 \wedge x_2$. □

Proposition 55. *If α is a pseudo-monotone connective such that $S_{\mathrm{c}} \subseteq \mathcal{C}(\alpha)$, then $T(\alpha\neg) \preceq \mathbf{M}$.*

Proof. Assume that α is an n-ary pseudo-monotone function such that $S_{\mathrm{c}} \subseteq \mathcal{C}(\alpha)$, and define $\alpha' := [\alpha(x_1, \ldots, x_n) \wedge x_{n+1}]$. The function α' is clearly 1-separating because x_{n+1} must take value 1 in every true point; hence $\alpha' \in U_\infty$. Being a generator of a clone containing S_{c}, the function α is not in $M \cup L$; it follows that $\alpha' \notin M$. Consequently, α' is a generator of U_∞ or of $T_{\mathrm{c}}U_\infty$. There exists a linear reduction from $T(\alpha'\neg)$ to $T(\alpha\neg)$, as witnessed by the linear term $t\{\alpha(x_1, \ldots, x_n)/x_1, x_{n+1}/x_2\} \in T(\alpha)$, where $t \in T(\alpha)$ is the binary linear term representing $\wedge$ that is provided by Lemma 54 (recall that $\alpha \notin M \cup L$). Now $T(\alpha\neg) \preceq T(\alpha'\neg)$ by Proposition 33, and our desired conclusion follows because α' is clearly pseudo-monotone and $T(\alpha'\neg) \preceq \mathbf{M}$ by Proposition 53. □

6. Other NFSs

In this subsection we consider the remaining NFSs, namely, those whose connectives are in $V \cup L \cup \Lambda$ and those generated by a function that is not pseudo-monotone.

6.1. NFSs whose connectives are in $V \cup L \cup \Lambda$

We look first into NFSs whose connectives are in $V \cup L \cup \Lambda$. In view of Proposition 20 we may focus on NFSs with at most 3 connectives. In [5] it was shown that $\mathbf{M} \prec \mathbf{C}, \mathbf{D}, \mathbf{P}, \mathbf{P}^{\mathrm{d}}$. However the connectives considered were those of minimal arity, i.e., the binary disjunction $\vee$ and conjunction $\wedge$, and the ternary sum $\oplus_3$. These results still hold for connectives of arbitrary arity. Given a set of terms of the form $T = T(\alpha\beta\neg)$, using the clone composition table in [5], it is not difficult to verify that if T is an NFS the only possibilities are $\alpha \in \Lambda$, $\beta \in V$ or $\alpha \in V$, $\beta \in \Lambda$. Similarly, for a set of terms $T(\alpha\beta)$ to be an NFS, the only possibilities are $\alpha \in L$, $\beta \in \Lambda$ or $\alpha \in L$, $\beta \in V$.

The following proposition shows that any NFS based on two connectives, possibly with negation, is equivalent to the conjunctive, disjunctive, polynomial, or dual polynomial NFS.

Proposition 56. *For any connectives α, β, γ such that $\mathcal{C}(\alpha) \circ I = L$, $\mathcal{C}(\beta) = \Lambda_{\mathrm{c}}$, $\mathcal{C}(\gamma) = V_{\mathrm{c}}$, we have $T(\alpha\beta) \sim \mathbf{P}$, $T(\alpha\gamma) \sim \mathbf{P}^{\mathrm{d}}$, $T(\beta\gamma\neg) \sim \mathbf{C}$, $T(\gamma\beta\neg) \sim \mathbf{D}$.*

Proof. Without loss of generality, we may assume that α, β, and γ are of arity ℓ, m, and n, respectively, and that they have no inessential arguments. Then $\ell \geq 2$, $m \geq 2$, $n \geq 2$, and

$$\begin{aligned}
&\alpha(x_1, \ldots, x_\ell) \equiv x_1 \oplus x_2 \oplus \cdots \oplus x_\ell \oplus c =: t_\alpha \in T(\oplus) \quad \text{for some } c \in \mathbb{B},\\
&\beta(x_1, \ldots, x_m) \equiv x_1 \wedge x_2 \wedge \cdots \wedge x_m =: t_\beta \in T(\wedge),\\
&\gamma(x_1, \ldots, x_n) \equiv x_1 \vee x_2 \vee \cdots \vee x_n =: t_\gamma \in T(\vee).
\end{aligned}$$

Moreover,

$$\begin{aligned}
x_1 \wedge x_2 &\equiv \beta(x_1, x_2, 1, \ldots, 1) =: t_\wedge \in T(\beta),\\
x_1 \vee x_2 &\equiv \gamma(x_1, x_2, 0, \ldots, 0) =: t_\vee \in T(\gamma).
\end{aligned}$$

In order to represent $\oplus$ as a term $t_\oplus \in T(\alpha)$, we need to consider different cases: let

$$t_\oplus := \begin{cases} \alpha(x_1, x_2, 0, \ldots, 0, c), & \text{if } \ell \geq 3,\\ \alpha(x_1, x_2), & \text{if } \ell = 2 \text{ and } c = 0,\\ \alpha(\alpha(x_1, x_2), 0), & \text{if } \ell = 2 \text{ and } c = 1. \end{cases}$$

In each case it holds that $x_1 \oplus x_2 \equiv t_\oplus$.

The terms t_α, t_β, t_γ, $t_\oplus$, $t_\wedge$, $t_\vee$ are all linear, and now we can make straightforward translations between $T(\alpha\beta)$ and $\mathbf{P}$, between $T(\alpha\gamma)$ and $\mathbf{P}^{\mathrm{d}}$, between $T(\beta\gamma\neg)$ and $\mathbf{C}$, and between $T(\gamma\beta\neg)$ and $\mathbf{D}$.

For example, given a term $t \in T(\alpha\beta)$, we obtain an equivalent term $t' \in \mathbf{P}$ by replacing each subterm $\beta t_1 \dots t_m$ of t by $t_\beta\{t_1/x_1, \dots, t_m/x_m\}$ and each subterm $\alpha t_1 \dots t_\ell$ by $t_\alpha\{t_1/x_1, \dots, t_\ell/x_\ell\}$. Since the terms t_α and t_β are linear and since $|t_\alpha|_\oplus = \ell$ and $|t_\beta|_\wedge = m-1$, we have $|t'| = |t'|_\oplus + |t'|_\wedge = \ell|t|_\alpha + (m-1)|t|_\beta \leq \max(\ell, m-1)|t|$. It follows that for every function f, $C_{\mathbf{P}}(f) \leq 2C_{T(\alpha\beta)}(f)$, i.e., $\mathbf{P} \preceq T(\alpha\beta)$.

Conversely, given a term $t \in \mathbf{P}$, we obtain an equivalent term $t' \in T(\alpha\beta)$ by replacing each subterm $\wedge(t_1, t_2)$ of t by $t_\wedge\{t_1/x_1, t_2/x_2\}$ and each subterm $\oplus(t_1, t_2)$ by $t_\oplus\{t_1/x_1, t_2/x_2\}$. Since the terms $t_\oplus$ and $t_\wedge$ are linear and since $|t_\oplus|_\alpha \leq 2$ and $|t_\wedge|_\beta = 1$, we have $|t'| = |t'|_\alpha + |t'|_\beta \leq 2|t|_\oplus + |t|_\wedge \leq 2|t|$. It follows that for every function f, $C_{T(\alpha\beta)}(f) \leq 2C_{\mathbf{P}}(f)$, i.e., $T(\alpha\beta) \preceq \mathbf{P}$.

The other claimed equivalences follow by similar arguments. □

Example 57. For example, the term $\vee_5(x_1, x_2, x_3, x_4, x_5)$ of size 1 involving the 5-ary disjunction $\vee_5$ is equivalent to the term $\vee(x_1, \vee(x_2, \vee(x_3, \vee(x_4, x_5))))$ of size 4. Similarly, the term $\vee(x_1, x_2)$ is equivalent to the term $\vee_5(x_1, x_2, 0, 0, 0)$. By Proposition 56, $\mathbf{D} = T(\vee\wedge\neg) \sim T(\vee_5\wedge\neg)$. ■

6.2. Non-monotonic NFSs

The case of non-monotonic NFSs still eludes us. However we conjecture that non-monotonic (quasi-)Sheffer NFSs are strictly more efficient than other NFSs.

To motivate our intuition, consider the set of terms $\mathbf{\Sigma} = T(\sigma)$ where $\sigma := [(x_1 \wedge x_2) \vee (\neg x_1 \wedge x_3)]$. Observe that σ is not pseudo-monotone since $\sigma(x_1, 1, 0) \equiv x_1$ and $\sigma(x_1, 0, 1) \equiv \neg x_1$. Also, from the equivalences $\sigma(x_1, 0, 1) \equiv \neg x_1$, and $\sigma(x_1, x_2, 0) \equiv x_1 \wedge x_2$, it follows that $\mathbf{\Sigma}$ is a Sheffer NFS. Moreover, $\mathbf{\Sigma}$ is at least as efficient as any other Sheffer or quasi-Sheffer NFS.

Lemma 58. *For any set of terms $T(\alpha\neg)$ (resp. $T(\alpha)$), $\mathbf{\Sigma} \preceq T(\alpha\neg)$ (resp. $T(\alpha)$).*

Proof. By Boole's expansion theorem (also known as Shannon's decomposition [18]), for any connective α of arity n:

$$\alpha(x_1, \dots, x_n) \equiv t_j = \sigma(x_j, \alpha(x_1, \dots, x_n)\{1/x_j\}, \alpha(x_1, \dots, x_n)\{0/x_j\}).$$

As $\forall j,\ |t_j|_{x_j} = 1$, $T(\alpha\neg) \sqsupseteq_\forall \mathbf{\Sigma}$. By Proposition 35, $\mathbf{\Sigma} \preceq T(\alpha\neg)$. □

However, the converse seems unlikely, due to the fact that σ is neither increasing nor decreasing in x_1. As it will see, this implies that x_1 must occur more than once in any term $t \in T(\alpha\neg)$ representing σ, whenever α is a monotone function, and hence that α occurs more than once in t.

Proposition 59. *Let α be a monotone function. If $t \in T(\alpha\neg)$ is a term in which the variable x_i occurs exactly once, then $[t]$ is either increasing or decreasing in the i-th argument.*

Proof. We prove the claim by induction on the structure of terms. If $t = x_i$, then $[t] = e_i^{(n)}$, a projection, which is clearly increasing in the i-th argument. If $t = \neg t'$ for some term t' such that $[t']$ is increasing (decreasing, resp.) in the i-th argument, then $[t]$ is decreasing (increasing, resp.) in the i-th argument. Assume now that $t = \alpha(t_1, \ldots, t_n)$ for some terms $t_1, \ldots, t_n \in T(\alpha\neg)$. Then x_i appears in exactly one of the subterms $t_1, \ldots, t_n$, say in t_p. By the induction hypothesis, $[t_p]$ is either increasing or decreasing in its i-th argument. Set $\mathbf{a} = (a_1, \ldots, a_n)$ and $\mathbf{b} = (a_1, \ldots, a_{i-1}, b_i, a_{i+1}, \ldots, a_n)$ with $a_i \leq b_i$. Since for all $j \in [n] \setminus \{p\}$, the variable x_i does not appear in t_j and hence $[t_j]$ does not depend on the i-th argument, we have $[t_j](\mathbf{a}) = [t_j](\mathbf{b})$. If $[t_p]$ is increasing in the i-th argument, then $[t_p](\mathbf{a}) \leq [t_p](\mathbf{b})$, and the monotonicity of α implies

$$\begin{aligned}[t](\mathbf{a}) &= \alpha([t_1](\mathbf{a}), \ldots, [t_p](\mathbf{a}), \ldots, [t_n](\mathbf{a})) \\ &\leq \alpha([t_1](\mathbf{b}), \ldots, [t_p](\mathbf{b}), \ldots, [t_n](\mathbf{b})) = [t](\mathbf{b}),\end{aligned}$$

so $[t]$ is increasing in the i-th argument. Similarly, if $[t_p]$ is decreasing in the i-th argument, then so is $[t]$. □

Corollary 60. *Let α and β be connectives. If α is not pseudo-monotone and β is monotone, then there is no universal quasi-linear reduction from $T(\alpha\neg)$ to $T(\beta\neg)$.*

Assume now that β is a monotone connective and α is an n-ary connective that is not pseudo-monotone. Then there is an index $i \in \{1, \ldots, n\}$ such that α is neither increasing nor decreasing in the i-th argument. Let t be a term in $T(\beta\neg)$ such that $[t] = \alpha$. By Proposition 59, we must have $|t|_{x_i} > 1$. Consider now the following terms in $T(\alpha)$: $s_1 := \alpha(x_1, \ldots, x_n)$, and for $k \geq 1$,

$$\begin{aligned}&s_{k+1} := \\ &\alpha(x_1, \ldots, x_{i-1}, s_k\{x_i/x_1, x_{i+1}/x_2, \ldots, x_{i+N(k)-1}/x_{N(k)}\}, x_{i+N(k)}, \ldots, x_{N(k+1)}),\end{aligned}$$

where $N(k) := k \cdot (n-1) + 1$. Let $f_k := [s_k]$, for $k \geq 1$. We clearly have $C_{T(\alpha)}(f_k) \leq |s_k|_\alpha = k$. We obtain an equivalent term in $s'_k \in T(\beta\neg)$ by replacing in s_k each subterm $\alpha(t_1, \ldots, t_n)$ by $t\{t_1/x_1, \ldots, t_n/x_n\}$. The size of the resulting terms s'_k grows exponentially in k due to the repeated variable x_i in t. Of course, this straightforward replacement of the connective α by the term t does not necessarily produce a term of minimal size in $T(\beta\neg)$ representing f_k, so we cannot really conclude anything about $C_{T(\beta\neg)}(f_k)$. We are nevertheless lead to the following conjecture.

Conjecture 61. *If* $\mathbf{A}$ *is a non-monotonic quasi-Sheffer NFS, then* $\mathbf{A} \prec \mathbf{M}$.

7. Conclusion

In this paper we have extended the framework of [5] by adapting the notion of NFSs to sets of terms, which allows several generalizations of results, e.g., of Theorem 27. In particular, we have shown that the results do not depend on the choice of connectives (in particular, on their arity), as long as they are pseudo-monotone: the optimal monotonic NFSs are exactly those of the form $T(\alpha)$ or $T(\alpha\neg)$, where α is a pseudo-monotone connective. Moreover, optimal monotonic NFSs are pairwise equivalent.

However, this contribution reveals several challenging issues that constitute topics of further research. We mention some topics of ongoing research below.

- Prove Conjecture 61 in order to shed light on the classification of all NFSs. Moreover, study the complexity of redundant systems and systems with unbounded stratification (or alternation [4]).

- Study stratified Boolean *circuits*, i.e., terms with sharing in addition to Boolean terms. It is noteworthy that in the case of circuits with variable sharing, there is no distinction between $B \sqsupseteq A$, $B \sqsupseteq_\forall A$, or $B \sqsupseteq_\exists A$. However, other measures can be considered, e.g., the number of wires between the gates of the circuit.

- Extend the current setting to multiple-valued operations, i.e., defined on a set of cardinality at least 3. Here, one of the main difficulties follows from the fact that the set of clones on a finite set with at least three elements has the cardinality of continuum (see, e.g., [12]) and there is no complete description of the corresponding clone lattice.

Acknowledgments

The authors would like to thank the anonymous referees for insightful comments and valuable suggestions that helped significantly improve the paper.

References

[1] F. Baader and T. Nipkow. *Term rewriting and all that*. Cambridge University Press, 1998.

[2] E. Böhler and H. Vollmer. Boolean functions and Post's lattice with applications to complexity theory. 2002.

[3] M. L. Bonet and S. R. Buss. Size-depth tradeoffs for Boolean formulae. *Information Processing Letters*, 49(3):151–155, 1994.

[4] A. K. Chandra, D. C. Kozen, and L. J. Stockmeyer. Alternation. *J. ACM*, 28(1):114–133, 1981.

[5] M. Couceiro, S. Foldes, and E. Lehtonen. Composition of Post classes and normal forms of Boolean functions. *Discrete Mathematics*, 306(24):3223–3243, 2006.

[6] M. Couceiro and E. Lehtonen. Galois theory for sets of operations closed under permutation, cylindrification and composition. *Algebra Universalis*, 67(3):273–297, 2012.

[7] M. Couceiro, E. Lehtonen, J.-L. Marichal, and T. Waldhauser. An algorithm for producing median formulas for Boolean functions. In *Proc. of the Reed Muller 2011 Workshop*, pages 49–54, 2011.

[8] M. Couceiro, E. Lehtonen, and T. Waldhauser. Decompositions of functions based on arity gap. *Discrete Mathematics*, 312(2):238–247, 2012.

[9] M. Couceiro, E. Lehtonen, and T. Waldhauser. A survey on the arity gap. *Multiple-Valued Logic and Soft Computing*, 24(1-4):223–249, 2015.

[10] Y. Crama and P. L. Hammer. *Boolean functions: Theory, algorithms, and applications*. Cambridge University Press, 2011.

[11] S. Foldes and G. R. Pogosyan. Post classes characterized by functional terms. *Discrete Applied Mathematics*, 142(1):35 – 51, 2004.

[12] I. I. Ianov and A. A. Muchnik. The existence of k-valued closed classes having no finite basis. *Doklady Akademii Nauk SSSR*, 127(1):44–46, 1959.

[13] D. Lau. *Function Algebras on Finite Sets: Basic Course on Many-Valued Logic and Clone Theory*. Springer, 2006.

[14] J.-L. Marichal. Weighted lattice polynomials. *Discrete Mathematics*, 309(4):814–820, 2009.

[15] E. L. Post. *The Two-Valued Iterative Systems of Mathematical Logic*, volume 5 of *Annals of Mathematical Studies*, pages 1–122. Princeton University Press, Princeton, 1941.

[16] A. Salomaa. On essential variables of functions, especially in the algebra of logic. *Annales Academiæ Scientiarum Fennicæ*, Series A I 339:11, 1963.

[17] T. J. Schaefer. The complexity of satisfiability problems. In *Proceedings of the tenth annual ACM symposium on Theory of computing*, pages 216–226. ACM, 1978.

[18] C. Shannon. The synthesis of two-terminal switching circuits. *Bell Labs Technical Journal*, 28(1):59–98, 1949.

[19] P. Spira. On time-hardware complexity tradeoffs for boolean functions. In *Proceedings of the 4th Hawaii Symposium on System Sciences, 1971*, pages 525–527, 1971.

[20] W. Wernick. Complete sets of logical functions. *Trans. Amer. Math. Soc.* 51:117–132, 1942.

[21] R. Willard. Essential arities of term operations in finite algebras. *Discrete Mathematics*, 149(1-3):239–259, 1996.