Non-monotone Dualization via Monotone Dualization

Yoshitaka Yamamoto[1], Koji Iwanuma[1], Katsumi Inoue[2]

[1] University of Yamanashi
4-3-11 Takeda, Kofu-shi, Yamanashi 400-8511, Japan.
[2] National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, Japan.

1 Introduction

The *non-monotone dualization* (NMD) is one of the most essential tasks required for finding hypotheses in various ILP settings, like *learning from entailment* [1, 2] and *learning from interpretations* [3]. Its task is to generate an irredundant prime CNF formula ψ of the dual f^d where f is a *general* Boolean function represented by CNF [4]. The DNF formula ϕ of f^d is easily obtained by De Morgan's laws interchanging the connectives of the CNF formula. Hence, the main task of NMD is to translate the DNF ϕ to an equivalent CNF ψ. This translation is used to find an alternative representation of the input form. For instance, given a set of models, it is desirable to seek underlying structure behind the models.

Example 1. Suppose that a set of models M are given as follows:

$$M = \{\ (bird \wedge normal \wedge flies),\ (\neg flies \wedge \neg normal),\ (\neg flies \wedge \neg bird)\ \}.$$

By treating M as the DNF formula, we translate it into CNF with NMD:

$$H = (bird \vee \neg flies) \wedge (normal \vee \neg flies) \wedge (flies \vee \neg normal \vee \neg bird).$$

In fact, H is regarded as a hypothesis in learning from interpretations [3].

In contrast, by converting a given CNF formula into DNF, we obtain the models satisfying the CNF formula. This fact shows an easy reduction from SAT problems to NMD, and then gives NP-hardness of it [5]. In this context, the research has been much focused on some restricted classes of NMD.

Monotone dualization (MD) is one such class that deals with *monotone* Boolean functions for which CNF formulas are negation-free [6, 7]. MD is one of the few problems whose tractability status with respect to polynomial-total time is still unknown. Besides, it is known that MD has many equivalent problems in discrete mathematics, such as the minimal hitting set enumeration and the transversal hypergraph computation [6]. Thus, this class has received much attention that yields remarkable algorithms: in terms of complexity, Fredman and Khachiyan [8] show that MD is solvable in a quasi-polynomial-total time (i.e.,

$(n+m)^{O(log(n+m))}$ where n and m denote the input and output size, respectively). Uno [9] shows a practically fast algorithm whose average computation time is experimentally $O(n)$ per output, for randomly generated instances.

This paper aims at clarifying whether or not NMD can be solved using these techniques of MD, and if it can be then how it is realized. In general, it is not straightforward to use them because of the following two problems in NMD:

- NMD has to treat *redundant clauses* like resolvents and tautologies.

  Example 2. Let a CNF formula ϕ be $(x_1 \vee x_2) \wedge (\overline{x_2} \vee x_3)$. If we treat negated variables as regular variables, we can apply MD to ϕ and obtain the CNF formula $\psi = (x_1 \vee \overline{x_2}) \wedge (x_1 \vee x_3) \wedge (x_2 \vee \overline{x_2}) \wedge (x_2 \vee x_3)$. However, ψ contains the tautology $x_2 \vee \overline{x_2}$ and the resolvent $x_1 \vee x_3$ of $x_1 \vee \overline{x_2}$ and $x_2 \vee x_3$.

- Unlike MD, the output of NMD is *not* necessarily *unique*. It is known that the output of MD uniquely corresponds to the set of all the prime implicates of f^d [10]. In contrast, some prime implicates can be redundant in NMD problems. Thus, the output of NMD corresponds to an irredundant subset of the prime implicates. However, such a subset is not unique in general.

For the first problem, this paper shows a technique to prohibit any resolvents from being generated in MD. This is done by simply adding some tautologies to the input CNF formula ϕ in advance. We denote by ϕ_t and ψ_t the extended input formula and its output by MD, respectively. Then, ψ_t contains no resolvents.

Example 3. Recall Example 2. We consider the CNF formula $\phi_t = (x_1 \vee x_2) \wedge (\overline{x_2} \vee x_3) \wedge (x_2 \vee \overline{x_2})$ obtained by adding one tautology $x_2 \vee \overline{x_2}$. Then, MD generates the CNF formula $\psi_t = (x_1 \vee \overline{x_2}) \wedge (x_2 \vee \overline{x_2}) \wedge (x_2 \vee x_3)$. Indeed, ψ_t does not contain the resolvent $x_1 \vee x_3$, unlike ψ in Example 2.

By removing all tautologies from ψ_t, we obtain an irredundant CNF formula, denoted by ψ_{ir}. Note that ψ_{ir} is $(x_1 \vee \overline{x_2}) \wedge (x_2 \vee x_3)$ in Example 3.

We next address the second problem using a good property of ψ_{ir}: a subset of the prime implicates is irredundant (i.e., an output of NMD) if and only if it subsumes ψ_{ir} but never subsumes ψ_{ir} if any clause is removed from it. This particular relation is called *minimal subsumption*. We then show that every subset satisfying the minimal subsumption is generated by MD. In this way, we reduce an original NMD problem into two MD problems: the one for computing ψ_{ir}, and the other for generating a subset corresponding to an output of NMD. Due to space limitations, full proofs are omitted.

2 Background

2.1 Preliminaries

A *Boolean function* is a mapping $f : \{0,1\}^n \rightarrow \{0,1\}$. We write $g \models f$ if f and g satisfy $g(v) \leq f(v)$ for all $v \in \{0,1\}^n$. g is (*logically*) *equivalent* to f, denoted by $g \equiv f$, if $g \models f$ and $f \models g$. A function f is *monotone* if $v \leq w$

implies $f(v) \leq f(w)$ for all $v, w \in \{0,1\}^n$; otherwise it is *non-monotone*. Boolean variables $x_1, x_2, \ldots, x_n$ and their negations $\overline{x_1}, \overline{x_2}, \ldots, \overline{x_n}$ are called *literals*. The *dual* of a function f, denoted by f^d, is defined as $\overline{f}(\overline{x})$ where $\overline{f}$ and $\overline{x}$ is the negation of f and x, respectively.

A *clause* (*resp. term*) is a disjunction (*resp.* conjunction) of literals which is often identified with the set of its literals. It is known that a clause is *tautology* if it contains complementary literals. A clause C is an *implicate* of a function f if $f \models C$. An implicate C is *prime* if there is no implicate C' such that $C' \subset C$.

A *conjunctive normal form* (CNF) (*resp. disjunctive normal form* (DNF)) formula is a conjunction of clauses (*resp.* disjunction of terms) which is often identified with the set of clauses in it. A CNF formula ϕ is *irredundant* if $\phi \not\equiv \phi - \{C\}$ for every clause C in ϕ; otherwise it is *redundant*. ϕ is *prime* if every clause in ϕ is a prime implicate of ϕ; otherwise it is *non-prime*. Let ϕ_1 and ϕ_2 be two CNF formulas. ϕ_1 *subsumes* ϕ_2, denoted by $\phi_1 \succeq \phi_2$, if there is a clause $C \in \phi_1$ such that $C \subseteq D$ for every clause $D \in \phi_2$. In turn, ϕ_1 *minimally subsumes* ϕ_2, denoted by $\phi_1 \succeq^{\natural} \phi_2$, if ϕ_1 subsumes ϕ_2 but $\phi_1 - \{C\}$ does not subsume ϕ_2 for every clause $C \in \phi_1$.

Let ϕ be a CNF formula. $\tau(\phi)$ denotes the CNF formula obtained by removing all tautologies from ϕ. We say ϕ is *tautology-free* if $\phi = \tau(\phi)$. Now, we formally define the dualization problem as follows.

Definition 1 (Dualization problem).

Input:	A tautology-free CNF formula ϕ
Output:	An irredundant prime CNF formula ψ such that ψ is logically equivalent to ϕ^d

We call it *monotone dualization* (MD) if ϕ is negation-free; otherwise it is called *non-monotone dualization* (NMD). As well known [6], the task of MD is equivalent to enumerating the *minimal hitting sets* (MHSs) of a family of sets.

2.2 MD as MHS enumeration

Definition 2 ((Minimal) Hitting set). Let Π be a finite set and $\mathcal{F}$ be a subset family of Π. A finite set E is a *hitting set* of $\mathcal{F}$ if for every $F \in \mathcal{F}$, $E \cap F \neq \emptyset$. A finite set E is a *minimal hitting set* (MHS) of $\mathcal{F}$ if E satisfies that

1. E is a hitting set of $\mathcal{F}$;
2. For every subset $E' \subseteq E$, if E' is a hitting set of $\mathcal{F}$, then $E' = E$.

Note that any CNF formula ϕ can be identified with the family of clauses in ϕ. Now, we consider the CNF formula, denoted by $M(\phi)$, which is the conjunction of all the MHSs of the family ϕ. Then, the following holds.

Theorem 1. [10] Let ϕ be a tautology-free CNF formula. A clause C is in $\tau(M(\phi))$ if and only if C is a non-tautological prime implicate of ϕ^d.

Hence, the output of MD for ϕ uniquely corresponds to $\tau(M(\phi))$: the set of all MHSs of the family ϕ by Theorem 1.

2.3 NMD as MHS enumeration

Our motivation is to clarify whether or not any NMD problem can be reduced into some MD problems. While MD is done by the state-of-the-art algorithms to compute MHSs [9], it is not straightforward to use them for NMD. Here, we review the two problems explained before in the context of MHS enumeration.

1. *Appearance of redundant clauses*: $\tau(M(\phi))$ is prime but not irredundant.

Example 4. Recall the input CNF formula $\phi_2 = \{\{x_1,\ x_2\}, \{\overline{x_2},\ x_3\}\}$ of Example 2. Then, $\tau(M(\phi_2)) = \{\{x_1,\ \overline{x_2}\},\ \{x_1,\ x_3\},\ \{x_2,\ x_3\}\}$. Indeed, this contains the redundant clause $\{x_1,\ x_3\}$.

2. *Non-uniqueness of NMD solutions*: there are many subsets of $\tau(M(\phi))$ that are prime and irredundant.

Example 5. Let the input CNF formula ϕ be $\{\{x_1,\ \overline{x_2},\ \overline{x_3}\},\ \{\overline{x_1},\ x_2,\ x_3\}\}$. $\tau(M(\phi))$ consists of the non-tautological prime implicates as follows:

$$\tau(M(\phi)) = \{\{x_1,\ x_2\},\ \{\overline{x_1},\ \overline{x_3}\},\ \{\overline{x_2},\ x_3\},\ \{x_1,\ x_3\},\ \{\overline{x_1},\ \overline{x_2}\},\ \{\overline{x_3},\ x_2\}\}.$$

Then, we may notice that there are at least two irredundant subsets of $\tau(M(\phi))$:

$$\psi_1 = \{\{x_1,\ x_2\},\ \{\overline{x_1},\ \overline{x_3}\},\ \{\overline{x_2},\ x_3\}\}.\ \psi_2 = \{\{x_1,\ x_3\},\ \{\overline{x_1},\ \overline{x_2}\},\ \{\overline{x_3},\ x_2\}\}.$$

Indeed, ψ_1 is equivalent to ψ_2, and thus both are equivalent to $\tau(M(\phi))$.

To address the two problems, this paper focuses on the following CNF formula.

Definition 3 (Bottom formula). Let ϕ be a tautology-free CNF formula and $Taut(\phi)$ be $\{\ x \vee \overline{x}\ \mid\ \phi \textit{ contains both } x \textit{ and } \overline{x}\ \}$. Then, the *bottom formula wrt* ϕ (in short, bottom formula) is defined as the CNF formula $\tau(M(\phi \cup Taut(\phi)))$.

3 Properties of bottom formulas

Now, we show two properties of bottom formulas.

Theorem 2. Let ϕ be a tautology-free CNF formula. Then, the bottom formula wrt ϕ is irredundant.

Example 6. Recall ϕ_2 in Example 4. Then, $Taut(\phi_2)$ is $\{x_2 \vee \overline{x_2}\}$, and the bottom formula wrt ϕ_2 is $\{\{x_1,\ \overline{x_2}\},\ \{x_2,\ x_3\}\}$. Indeed, it is irredundant, since it does not contain the resolvent $\{x_1,\ x_3\}$, unlike Example 4.

While any bottom formula is irredundant, it is not necessarily prime.

Example 7. Recall the CNF formula ϕ in Example 5. Since $Taut(\phi)$ is the set $\{\{x_1,\ \overline{x_1}\},\ \{x_2,\ \overline{x_2}\},\ \{x_3,\ \overline{x_3}\}\}$, the bottom formula is as follows:

$$\{\{x_1, x_2, x_3\},\ \{\overline{x_3}, x_2, x_1\},\ \{\overline{x_3}, x_2, \overline{x_1}\},\ \{\overline{x_2}, x_3, x_1\},\ \{\overline{x_2}, x_3, \overline{x_1}\},\ \{\overline{x_2}, \overline{x_3}, \overline{x_1}\}\}.$$

We write $C_1, C_2, \ldots, C_6$ for the above clauses in turn (i.e., C_4 is $\{\overline{x_2},\ x_3,\ x_1\}$). We then notice that the bottom formula is non-prime, because it contains a non-prime implicate C_1 whose subset $\{x_1,\ x_2\}$ is an implicate of ϕ^d.

As shown in Example 7, the bottom formula itself is not necessarily an output of NMD. However, every NMD output is logically described with this formula.

Theorem 3. Let ϕ be a tautology-free CNF formula. ψ is an output of NMD for ϕ iff $\psi \subseteq \tau(M(\phi))$ and ψ minimally subsumes the bottom formula wrt ϕ.

Example 8. Recall Example 5 and Example 7. Fig. 1 describes the subsumption lattice bounded by two irredundant prime outputs ψ_1 and ψ_2 as well as the bottom formula $\{C_1, C_2, \ldots, C_6\}$. The solid (*resp.* dotted) lines show the subsumption relation between ψ_1 (*resp.* ψ_2) and the bottom formula. We then notice that both outputs ψ_1 and ψ_2 minimally subsume the bottom formula.

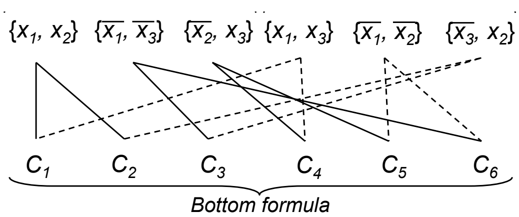

Fig. 1. Subsumption lattice bounded by NMD outputs and the bottom formula

4 Reconstructing NMD into MD

Theorem 3 shows that every NMD output ψ can be generated by selecting a subset ψ of $\tau(M(\phi))$ that minimally subsumes the bottom formula. Now, we show that the task of this selection is done by MD computation.

Let the bottom formula be $\{C_1,\ C_2,\ \ldots,\ C_n\}$. We then denote by S_i ($1 \leq i \leq n$) the set of clauses in $\tau(M(\phi))$ each of which is a subset of C_i. $\mathcal{F}_\phi$ denotes the family of those sets $\{S_1, S_2, \ldots, S_n\}$.

Theorem 4. Let ϕ be a tautology-free CNF formula. ψ is an output of NMD for ϕ if and only if ψ is an MHS of $\mathcal{F}_\phi$.

Example 9. Recall Example 8. We denote each clause of ψ_1 and ψ_2 in Fig. 4 by $D_1, \ldots, D_6$, starting from left to right (i.e., D_4 is $\{x_1,\ x_3\}$). $\mathcal{F}_\phi$ is as follows:

$$\mathcal{F}_\phi = \{\{D_1,\ D_4\},\ \{D_1,\ D_6\},\ \{D_2,\ D_6\},\ \{D_3,\ D_4\},\ \{D_3,\ D_5\},\ \{D_2,\ D_5\}\}.$$

By MHS computation, we have the five solutions, which contain $\{D_1,\ D_2,\ D_3\}$ and $\{D_4,\ D_5,\ D_6\}$ that correspond to ψ_1 and ψ_2, respectively.

5 Concluding remarks

This paper has presented a reduction technique from arbitrary NMD problem to two equivalent MD problems. Whereas algorithms and computation on MD has been extensively studied, it was not clarified whether or not NMD can be solved using the state-of-the-art MD computation. For this open problem, we give a solution how it is to be realized.

Our result can be used to investigate the complexity of NMD from the viewpoint of MD computation. For instance, the complexity of generating one NMD output can be described as follows:

$$(n+t+x)^{O(log(n+t+x))} + (x+1)^{O(log(x+1))}, \qquad (1)$$

where n, t and x are the sizes of the input formula ϕ, the tautologies $Taut(\phi)$ and the bottom formula $\tau(M(\phi \cup Taut(\phi)))$, respectively. This is simply derived from the result on the complexity of MD computation [8]. Note that the right-hand term in Formula (1) comes from the complexity of the *incremental* MHS generation problem [11]. In contrast, the complexity of MD for computing the prime implicates of ϕ is described as $(n+m)^{O(log(n+m))}$ where m is the size of $\tau(M(\phi))$. Hence, the difference of their complexities lies in how big the blow-up in size can be when the bottom formula is derived. In other words, our proposal is not a polynomial reduction to MD, and does not establish NP-completeness of MD. However, if x is equal to m, like Example 4 and Example 7, there is no difference between NMD and MD in terms of the complexity. In this sense, it is an important future work to characterize those subclasses of NMD with only polynomial increase in the problem size x.

References

1. Inoue, K.: Induction as consequence finding. Machine Learning **55(2)** (2004) 109–135
2. Yamamoto, Y., Inoue, K., Iwanuma, K.: Inverse subsumption for complete explanatory induction. Machine Learning **86(1)** (2011) 115–139
3. DeRaedt, L.: Logical settings for concept-learning. Artificial Intelligence **95** (1997) 187–201
4. Eiter, T., Makino, K.: Abduction and the dualization problem. In: Proceedings of the 6th Int. Conf. on Discovery Science. Volume 2843 of LNCS. (2003) 1–20
5. Eiter, T., Ibaraki, T., Makino, K.: Recognition and dualization of disguised bidual Horn functions. Information Processing Letters **82** (2002) 283–291
6. Eiter, T., Makino, K., Gottlob, G.: Computational aspects of monotone dualization: A brief survey. Discrete Applied Mathematics **156** (2008) 2035–2049
7. Hagen, M.: Algorithmic and computational complexity issues of MONET. PhD thesis, Friedrich Schiller University Jena (2008)
8. Fredman, M., Khachiyan, L.: On the complexity of dualization of monotone disjunctive normal forms. Algorithms **21** (1996) 618–628
9. Uno, T.: A practical fast algorithm for enumerating minimal set coverings. In: Proceedings of the 83rd SIGAL Conf. of the Information Processing Society of Japan. (2002) 9–16
10. Rymon, R.: An SE-tree based prime implicant generation algorithm. Annals of Mathematics and Artificial Intelligence (1994) 351–366
11. Elbassioni, K.M.: On the complexity of monotone dualization and generating minimal hypergraph transversals. Discrete Applied Mathematics **156** (2008) 2109–2123