Ordered direct implicational basis of a finite closure system

Kira Adaricheva
Yeshiva University, New York

J.B.Nation
University of Hawaii, Honolulu

Robert Rand
Yeshiva University, New York

Abstract

Closure system on a finite set is a unifying concept in logic programming, relational data bases and knowledge systems. It can also be presented in the terms of finite lattices, and the tools of economic description of a finite lattice have long existed in lattice theory. We present this approach by describing the so-called D-basis and introducing the concept of *ordered direct basis* of an implicational system. A direct basis of a closure operator, or an implicational system, is a set of implications that allows one to compute the closure of an arbitrary set by a single iteration. This property is preserved by D-basis at the cost of following a prescribed order in which implications will be attended. In particular, this allows us to optimize the *forward chaining procedure* in logic programming that uses the Horn fragment of propositional logic. It turns out that running the D-basis in one iteration is more efficient than running a shorter, but un-ordered, *canonical basis* of J. Guigues and V. Duquenne, and examples demonstrate that the canonical basis cannot always be ordered. We show that one can extract the D-basis from any direct unit basis Σ in time polynomial of $|\Sigma|$, and it takes only linear time of the size of the D-basis to put it into a proper order.

1 Introduction

In (Bertet and Monjardet 2010), it is shown that five implicational bases for a closure operator on a finite set, found in various contexts in the literature, are actually the same. The goal of this paper is to demonstrate that standard lattice-theoretic results about the "most economical way" to describe the structure of a finite lattice may be transformed into a basis for a closure system naturally associated with that lattice. We may refer to (Nation 1990), where the coding of a finite lattice in the form of so-called OD-graph was first suggested.

We will call the basis directly following from this OD-graph a *D-basis*, since it is closely associated with the *D-relation* on the set of join-irreducibles of a lattice (not necessarily finite) that was crucial in the studies of free and lower bounded lattices, see (Freese, Jezek, and Nation 1995).

The D-basis is a subset of a so-called *dependence relation basis* (Definition 6 in (Bertet and Monjardet 2010)). Thus, it is also a subset of *a canonical direct unit basis* that unifies the five bases discussed in (Bertet and Monjardet 2010). We give an example to demonstrate that the reverse inclusion does not hold, thus showing that this newly introduced D-basis is generally shorter than the existing ones.

Recall that the main desirable feature of bases from (Bertet and Monjardet 2010) is that they be *direct*, which means that the computation of the closure of any subset can be a single iteration of the basis.

While the D-basis is not direct in this meaning of the term, the closures can still be computed by a single iteration of the basis, provided the basis was put in a specific order prior to computation. Thus, applying the D-basis can be compared to the iteration known in artificial intelligence as the *forward chaining algorithm*. Moreover, there is a simple and effective linear time algorithm for ordering a D-basis appropriately.

We also discuss the so-called E-relation, introduced in (Freese, Jezek, and Nation 1995), which leads to the definition of the E-basis in closure systems *without cycles*. In general, the implications derived from the E-relation do not necessarily form a basis of a closure system, but in closure systems without cycles, the E-relation implications do form a basis which is ordered direct, is contained in the D-basis, and often shorter than D-basis. We define the polynomial time algorithm for ordering the E-basis.

While the canonical basis has the minimal number of implications among all the bases of a closure system, it does not have the feature of D-basis or E-basis discussed in this paper, namely, it cannot be turned into ordered direct basis: in the last section of our paper we present examples of closure systems on 6-element set, for which the canonical basis cannot be ordered. As a result, the canonical basis must typically be iterated at least twice to ensure the computation of the closure, whereas the D-basis requires only a single pass.

We also discuss the further polynomial-time optimizations of both the D-basis and the canonical basis.

The next two sections contain the required definitions and establish connections between finite lattices, closure operators and implicational systems. The reader may consult the survey (Caspard and Monjardet 2003) for the various aspects of closure systems on the finite sets. Most of the proofs are omitted in this article due to the lack of space, and will be provided in the journal edition.

2 Lattices and closure operators

By a lattice one means an algebra with two binary operations $\wedge, \vee$, called *meet* and *join*, respectively. A lattice is finite when the base set of this algebra is finite. The symbols $\bigwedge, \bigvee$ are used when more than two elements meet or join. We will use the notation 0 for the least element of a lattice, and 1 for its greatest element. Simultaneously, every lattice is a partially ordered set in which every two elements have a least upper bound (which coincides with the join of those elements), and a greatest lower bound (the meet). If $a \leq b$ in lattice L, then we denote by $[a, b]$ the interval in L, i.e., the set of all c satisfying $a \leq c \leq b$.

Recall the standard connection between a closure operator on a set and the lattice of its closed sets. Given a non-empty set S and the set $P(S) = 2^S$ of all its subsets, *a closure operator* is a map $\phi : P(S) \to P(S)$ that satisfies the following, for all $X, Y \in P(S)$:

(1) increasing: $X \subseteq \phi(X)$;

(2) isotone: $X \subseteq Y$ implies $\phi(X) \subseteq \phi(Y)$;

(3) idempotent: $\phi(\phi(X)) = \phi(X)$.

It would be convenient for us to refer to the pair $\langle S, \phi \rangle$ of a set S and a closure operator on it as *a closure system*.

A subset $X \subseteq S$ is called *closed* if $\phi(X) = X$. The collection of closed subsets of a closure operator ϕ on S forms a lattice, which is usually called a *closure lattice* of the closure system $\langle S, \phi \rangle$. This paper deals with only finite closure systems and finite lattices.

Conversely, we can associate with every finite lattice L a particular closure system $\langle S, \phi \rangle$ in such a way that L is isomorphic to a closure lattice of that closure system. Consider $J(L) \subseteq L$, a subset of *join-irreducible elements*. An element $j \in L$ is called join-irreducible, if $j \neq 0$, and $j = a \vee b$ implies $a = j$ or $b = j$. We define a closure system with $S = J(L)$ and the following closure operator:

$$\phi(X) = [0, \bigvee X] \cap J(L)$$

It is a straightforward argument to check that the closure lattice of ϕ is isomorphic to L.

Example 1

Consider a simple example illustrating a closure system built from the lattice $L = \{0, a, b, c, 1\}$, for which $0 < a < b < 1$, $0 < c < 1$, $a \vee c = b \vee c = 1$ and $a \wedge c = b \wedge c = 0$. Then $S = J(L) = \{a, b, c\}$. The closed subsets are $[0, x] \cap J(L)$ for $x \in L$, which are $\emptyset$, $\{a\}$, $\{c\}$,$\{a, b\}$ and $\{a, b, c\}$. Knowing all closed subsets, one can define a closure of X, or $\phi(X)$, as the smallest closed set containing X. For example, $\phi(\{b\}) = \{a, b\}$.

There are infinitely many sets and closure operators whose closure lattice is isomorphic to a given L. On the other hand, the one just described is the unique one with two additional properties:

(1) $\phi(\emptyset) = \emptyset$;

(2) $\phi(\{i\}) \setminus \{i\}$ is closed, for every $i \in S$.

Note that (1) is a special case of (2), and (2) implies property

(3) $\phi(\{i\}) = \phi(\{j\}) \to i = j$, for any $i, j \in S$.

We will call a closure system with properties (1),(2) above *a standard closure system*, and the closure system with (3) *reduced*.

If the closure system $\langle S, \phi \rangle$ is not reduced, one can modify it to produce an equivalent one that is reduced. Moreover, there is an effective algorithm for doing so. Thus, for all practical purposes, one can work with a reduced closure system $\langle U, \mu \rangle$ replacing a given one $\langle S, \phi \rangle$. Slightly more effort yields an equivalent standard closure system $\langle V, \nu \rangle$.

3 Closure systems, their bases and Horn formulas

If $y \in \phi(X)$, then this relation between an element $y \in S$ and a subset $X \subseteq S$ in a closure system can be written in the form of implication: $X \to y$. Thus, the closure system $\langle S, \phi \rangle$ can be replaced by the set of implications:

$$\Sigma_\phi = \{X \to y : y \in S, X \subseteq S \text{ and } y \in \phi(X)\}$$

Conversely, any set of implications Σ defines a closure system: the closed sets are exactly subsets $Y \subseteq S$ that respect the implications from Σ, i.e. if $X \to x$ is in Σ, and $X \subseteq Y$, then $x \in Y$.

Two sets of implications Σ and Σ' on the same set S are called *equivalent*, if they define the same closure system on S. The term *basis* is used for a set of implications Σ' satisfying some minimality condition; thus there may be different types of bases.

Note that, in general, one can consider implications of the form $X \to Y$, where Y is not necessarily a one-element subset of S. Following (Bertet and Monjardet 2010) we will call a basis Σ a *unit implicational basis* if $|Y| = 1$, for all implications $X \to Y$ in Σ. We will be mostly concerned with the unit implicational bases, except for the discussion of the canonical basis and its relation with the D-basis and E-basis. Given any unit basis, one can always collapse the implications with the same premise into one with all conclusions combined into a single set. This will be called an *aggregated* basis.

For a set of implications $\Sigma = \{X_1 \to Y_1, \dots, X_m \to Y_m\}$, define the *size* by $s(\Sigma) = \sum_{j=1}^{m}(|X_j| + |Y_j|)$. This is one convenient measure of the complexity of an implicational system.

In general, implications $X \to y$, $X \subseteq S, y \in S$, can be treated as the formulas of propositional logic over the set of variables S, equivalent to $y \vee \bigvee_{x \in X} \neg x$. The formulas of this form are also called *definite Horn-clauses*. More generally, Horn clauses are disjunctions of several negative literals and at most one positive literal. The presence of a positive literal makes a Horn clause *definite*.

What is called a *model* of a definite Horn clause in logic programming literature corresponds to a closed set of the closure operator defined by this clause. Indeed, by the definition, a model is simply $M \in 2^S$, i.e., a tuple of zeros and ones assigned to variables from S such that the formula is true (=1) on this assignment. If the formula is a definite

Horn clause $X \to y$, then M corresponds to a subset Y of S that is closed for a closure operator on S defined by $X \to y$. In fact, M is simply the characteristic function of Y.

There is also a direct correspondence between Horn formulas and Horn Boolean functions: a Boolean function $f : \{0,1\}^n \to \{0,1\}$ is called a (*pure* or definite) *Horn function*, if it has some CNF representation given by a (definite) Horn formula Σ. The dual definition is sometimes used in the literature, so that a Horn function is given by some formula in DNF, whose negation is a Horn formula (Crama and Hammer 2011). Using either definition, one can translate many results on Horn Boolean functions to the language of closure operators, see more details in (Bertet and Monjardet 2010).

Assume that we consider a set Σ of Horn clauses over some finite set of variables $S = \{x_1, \ldots, x_n\}$. If some Horn clause α in Σ is not definite and it does not use all variables from S, i.e., it is of the form $\bigvee_{x \in X} \neg x$ for some $X \subset S$, then we could define the set of definite clauses $\Sigma_\alpha = \{X \to y : y \in S \setminus X\}$. In case clause $\alpha \in \Sigma$ is not definite and uses all the variables from S, we define $\Sigma_\alpha = \emptyset$ (another possibility, $x_1 \to x_1$). It is easy to observe that the set of models of Σ_α consists of all models of $\bigvee_{x \in X} \neg x$ and one additional model, which is a tuple of all ones (representing the set S itself). If follows that the set of definite clauses Σ', where each non-definite clause α from Σ is replaced by set of clauses Σ_α, has the set of models that extends the set of models of Σ by a single tuple of all ones. This includes the case when Σ has no models, i.e., when it is *inconsistent*.

This observation allows to reduce the solution of various questions about sets of Horn clauses to the sets of *definite* Horn clauses. Thus, it emphasizes the importance of the study of closure operators on S.

One of the important questions in logic programming is whether one formula ϕ is *a consequence* of the set (or conjunction) of other formulas Σ. Denoted by $\Sigma \models \phi$, this means that every model of Σ is also a model of ϕ. If ϕ and formulas in Σ are Horn clauses, then, translating this question to the language of closure systems, one reduces it to checking whether every closed set of a closure system defined by Σ respects ϕ.

4 D-basis

In this section we are going to define a basis that translates to the language of closure systems the defining relations of a finite lattice developed in the lattice theory framework. One can consult (Freese, Jezek, and Nation 1995) for the corresponding notion of a minimal cover and D-relation used in the theory of free lattices and lower bounded lattices.

Given a *reduced* closure system $\langle S, \phi \rangle$, let us define two auxiliary relations. The first relation is between the subsets of S: we write $X \ll Y$, if for every $x \in X$ there is $y \in Y$ satisfying $x \in \phi(y)$. In Example 1, for instance, we have $\{a, c\} \ll \{b, c\}$. Note that $X \subseteq Y$ implies $X \ll Y$. We also write $X \sim_{\ll} Y$, if $X \ll Y$ and $Y \ll X$. This is true for $X = \{a, b, c\}$ and $Y = \{b, c\}$ in Example 1.

Several observations are easy.

Lemma 2 *The relation $\sim_{\ll}$ is an equivalence relation on $P(S)$.*

Denote the $\sim_{\ll}$-equivalence class containing X by $[X]$. Note that for any two members $X, Y \in [X]$, we have $\phi(X) = \phi(Y)$. Now we can define the relation $\leq_c$ on $\sim_{\ll}$-classes: $[X] \leq_c [Y]$, if $X \ll Y$.

Lemma 3 *The relation $\leq_c$ is a partial order on the set of $\sim_{\ll}$-equivalence classes.*

Each class $[X]$ is ordered itself with respect to relation of containment.

In Example 1, we have that $\{a, b, c\} \sim_{\ll} \{b, c\}$, thus, $[\{b, c\}]$ consists of two subsets, and $\{b, c\} \subseteq \{a, b, c\}$ is the minimal (with respect to the order of containment) subset in that equivalence class of $\sim_{\ll}$. We also have $\{a, c\} \ll \{b, c\}$, hence, $[\{a, c\}] \leq_c [\{b, c\}]$.

Lemma 4 *Each equivalence class $[X]$ has a unique minimal element with respect to the containment order.*

The second relation we want to introduce in this section, $\sim$, is between an element $x \in S$ and a subset $X \subseteq S$, which will be called a *cover* of x. We will write $x \sim X$, if $x \in \phi(X)$ and $x \notin \phi(x')$, for any $x' \in X$. Illustrating this notion in Example 1 gives us $b \sim \{a, c\}$. Note that it is not true that $a \sim \{b, c\}$, because $a < b$, whence $a \in \phi(b)$, for the corresponding standard closure operator.

Let us call a subset $Y \subseteq S$ a *minimal cover* of an element $x \in S$, if Y is a cover of x, and for every other cover Z of x, $Z \ll Y$ implies $Y \subseteq Z$.

To illustrate this notion, let us slightly modify Example 1. Rename element 0 as d and add a new 0 element: $0 < d$, resulting in lattice L_1. Thus, $J(L_1) = J(L) \cup \{d\}$. We will have $Y = \{a, c\}$ as a minimal cover for b. Indeed, the only other cover for b is $Z = \{a, c, d\}$, for which we have $Z \ll Y$ and $Y \subseteq Z$.

Lemma 5 *If $x \sim X$, then there exists Y such that $x \sim Y$, $Y \ll X$ and Y is a minimal cover for x. In other words, every cover can be $\ll$-reduced to a minimal cover.*

Definition 6 *Given a reduced closure system $\langle S, \phi \rangle$, define the D-basis Σ_D as a union of two subsets of implications:*

(1) $\{y \to x : x \in \phi(y) \setminus y, y \in S\}$;

(2) $\{X \to x : X$ *is a minimal cover for* $x\}$.

Lemma 7 *Σ_D generates $\langle S, \phi \rangle$.*

Proof We need to show that, for any $x \in S$ and $X \subseteq S$ such that $x \in \phi(X)$, the implication $X \to x$ follows from implications in Σ_D.

If $x \in \phi(x')$, for some $x' \in X$, then $X \to x$ follows from $x' \to x$ that is in Σ_D. So assume that $x \notin \phi(x')$, for any $x' \in X$. Then $x \sim X$. According to Lemma 5, there exists $Y \ll X$ such that $x \sim Y$, and Y is a minimal cover for x. Then $Y \to x$ is in Σ_D. Besides, for each $y \in Y$ there exists $x_y \in X$ such that $y \in \phi(x_y)$. Therefore, $x_y \to y$ is in Σ_D as well. Evidently, $X \to x$ is a consequence of $Y \to x$ and $\{x_y \to y : y \in Y\}$.

5 Comparison of the D-basis and the dependence relation basis

One of the bases discussed in (Bertet and Monjardet 2010) is the *dependence relation basis*. For a closure system $\langle S, \phi \rangle$, not necessarily reduced, the dependence relation basis is

$$\Sigma_\delta = \{X \to y : y \in \phi(X) \backslash X \text{ and } y \notin \phi(Z) \text{ for all } Z \subset X\}.$$

Since $Z \subseteq X$ implies $Z \ll X$, a minimal cover (as defined above) is automatically minimal with respect to containment. Thus we have the following connection.

Lemma 8 $\Sigma_D \subseteq \Sigma_\delta$.

The following example shows that the reverse inclusion does not always hold. Thus, Σ_D, in general, has fewer implications than Σ_δ.

Whenever there is no confusion, we will omit the braces in notations of subsets of some set S: $\{x\}, \{a, b, c\}$ etc. will be denoted simply x, abc etc.

Example 9

This example is based on Example 5 from (Bertet and Monjardet 2010). Consider the closure system on $S = \{1, 2, 3, 4, 5\}$ with the set of closed subsets $F = \{\emptyset, 1, 2, 3, 4, 12, 13, 234, 45, 12345\}$. Then $\Sigma_\delta = \{5 \to 4, 23 \to 4, 24 \to 3, 34 \to 2, 14 \to 2, 14 \to 3, 14 \to 5, 25 \to 1, 35 \to 1, 15 \to 2, 35 \to 2, 15 \to 3, 25 \to 3, 123 \to 5\}$.

All implication except $5 \to 4$ are of the form $X \to x$, where $x \sim X$. On the other hand, not all covers X are minimal covers of x. We can check that each of implications $15 \to 2, 35 \to 2, 15 \to 3, 25 \to 3$ does not represent a minimal cover. For example, $2 \sim 15$, but $14 \ll 15$ and $2 \sim 14$ is the minimal cover. In particular, D-basis consists of all implications from Σ_δ except the four indicated: $\Sigma_D = \{5 \to 4, 23 \to 4, 24 \to 3, 34 \to 2, 14 \to 2, 14 \to 3, 14 \to 5, 25 \to 1, 35 \to 1, 123 \to 5\}$.

As this example demonstrates, the D-basis can be obtained from Σ_δ simply by removing some unnecessary implications. It turns out that the same can be done for the big range of bases called *direct unit bases*. Moreover, it can be done in polynomial time in the size of the given basis. See Proposition 16 in the next section.

6 Direct basis versus ordered direct basis

The bases discussed in (Bertet and Monjardet 2010) are, in general, redundant: a proper subset of such a basis would generate the same closure system. For example, as we saw in previous section, Σ_δ from Example 5 was reduced to a smaller basis Σ_D.

While the desire to keep the basis as small as possible might be a plausible task, there is another property of a basis that could be better appreciated in a programming setting. Here we recall the definition of a so-called *direct basis*.

If Σ is some set of implications, let $\pi_\Sigma(X) = X \cup \{b : A \subseteq X \text{ and } (A \to b) \in \Sigma\}$. In order to obtain $\phi_\Sigma(X)$, for any $X \subseteq S$, one would need to repeat several iterations of π: $\phi(X) = \pi(X) \cup \pi^2(X) \cup \pi^3(X) \ldots$. Here $\pi^{k+1}(X) = \pi(\pi^k(X))$, $k \geq 1$. The bases for which one can obtain the closure of any set X performing only one iteration, i.e., $\phi(X) = \pi(X)$, are called *direct*.

It follows from Theorem 15 of (Bertet and Monjardet 2010) that the dependency relation basis Σ_δ is direct. Moreover, this basis is direct-optimal, meaning that no other direct basis for the same closure system can be found of smaller total size (sum of cardinalities of all sets participating in implications). In particular, any reduction of Σ_δ will cease to be direct. Thus, there is an apparent trade-off between the number of implications in the basis and the number of iterations one needs to compute the closures of subsets.

The goal of this section to implement a different approach to the concept of iteration. That would allow the same number of programming steps as with the iteration of π, while allowing us to reduce the bases to the smaller size. We note that this approach is used extensively in artificial intelligence in the *forward chaining procedure*, or equivalently, in computations connected to the closure operator associated with a set of implications, see (Kleine Buning and Lettman 1999).

Definition 10 *Suppose the set of implications Σ is equipped with some linear order $<$, or, equivalently, the implications are indexed by the natural numbers $\Sigma = \{s_1, s_2, \ldots, s_n\}$.*

Define a mapping $\rho_\Sigma : P(S) \to P(S)$ associated with this ordering as follows. For any $X \subseteq S$, let $X_0 = X$. If X_k is computed and implication s_{k+1} is $A \to B$, then

$$X_{k+1} = \begin{cases} X_k \cup B, & \text{if } A \subseteq X_k, \\ X_k, & \text{otherwise.} \end{cases}$$

Finally, $\rho_\Sigma(X) = X_n$. We will call ρ_Σ an ordered iteration *of Σ.*

Apparently, $\pi_\Sigma(X) \subseteq \rho_\Sigma(X)$, because all implications from Σ are applied to original subset X, while they are applied to potentially bigger subsets X_k in construction for $\rho_\Sigma(X)$. We note though that assuming the order on Σ is established, the number of computational steps to produce $\rho_\Sigma(X)$ is the same as for $\pi_\Sigma(X)$.

Definition 11 *The set of implications with some linear ordering on it, $\langle \Sigma, < \rangle$, is called* an ordered direct basis*, if, with respect to this ordering $\phi_\Sigma(X) = \rho_\Sigma(X)$, for all $X \subseteq S$.*

Proposition 12 *Let $\langle \Sigma, < \rangle = \{s_1, \ldots, s_n\}$ be the set of implications representing the ordered direct basis of a closure operator ϕ on finite set S. For every implication $X \to x$, $X \subseteq S, x \in S$, it can be answered in time $O(n)$ whether this implication is a consequence of Σ.*

Our next goal is to demonstrate that Σ_D is, in fact, an ordered direct basis. Moreover, it does not take much computational effort to impose a proper ordering on Σ_D.

Lemma 13 *Let $<$ be any linear ordering on Σ_D such that all implications of the form $y \to x$ precede implications $X \to x$, where X is a minimal cover of x. Then, with respect to this ordering, Σ_D is an ordered direct basis.*

Note in this regard, that the D-basis is ordered direct in both forms: in its original unit form, and in the aggregated form. Indeed, it follows from the fact that the only restriction on the order of the D-basis is to have its binary part prior to the rest of the basis.

Example 14

Consider the closure system on the set $S = \{1,2,3,4,5,6\}$ and the family of closed sets $F = \{1, 12, 13, 4, 45, 134, 136, 1362, 1346, 13456, 123456\}$. Then the D-basis of this system is $\Sigma_D = \{5 \to 4, 14 \to 3, 23 \to 6, 6 \to 3, 15 \to 6, 24 \to 6, 24 \to 5, 3 \to 1, 2 \to 1\}$. According to Lemma 13, the proper ordering that turns this basis into ordered direct can be defined, for example, as: (1) $5 \to 4$, (2) $6 \to 3$, (3) $3 \to 1$, (4) $2 \to 1$, (5) $14 \to 3$, (6) $23 \to 6$, (7) $15 \to 6$, (8) $24 \to 6$, (9) $24 \to 5$.

Corollary 15 *If $\Sigma_D = \{s_1, \dots, s_n\}$ is the D-basis of an implicational system Σ, then it requires time $O(n)$ to turn it into a direct ordered basis of Σ.*

The next statement shows that, given any direct unit basis, one can extract the D-basis from it in a polynomial time procedure.

Proposition 16 *Let $\langle S, \phi\rangle$ be a reduced closure system. If the direct unit basis Σ for this system has m implications, and $|S| = n$, then it requires time $O((nm)^2) \sim O(s(\Sigma)^2)$ to build the D-basis Σ_D equivalent to Σ.*

As follows from the procedure of Proposition 16, the D-basis is obtained from any direct unit basis by removing implications $X \to x$, for which X is not a minimal cover of x and $|X| > 1$. In particular, the binary part of the direct basis, i.e., implications of the form $y \to x$, remain to be in the D-basis.

We want to discuss the further optimization of the D-basis, as well as the any other basis that has the same binary part as the D-basis. As was noted in section 2, the join-irreducible elements of the lattice L of closed sets of $\langle S, \phi\rangle$ have the form $\phi(x), x \in S$. Besides, implication $y \to x$ belongs to the D-basis iff $x \in \phi(y)$ iff $\phi(x) \leq \phi(y)$. Thus, the binary part of the D-basis describes the partially ordered set $(J(L), \leq)$ of join-irreducible elements of L.

One can shorten the binary part of D-basis leaving only those implications $y \to x$, for which $\phi(y)$ covers $\phi(x)$ in $(J(L), \leq)$. This will come at the cost of the need to order the remaining implications. For example, if $y \to x$, $x \to z$, $y \to z$ are three implications from the binary part of some D-basis, then the last implication can be removed, under condition that the first two will be placed in that particular order into the ordered D-basis. Recall that if some set of implications Σ' is ordered, then $\rho_{\Sigma'}(X)$, the ordered iteration of Σ', is defined for every $X \subseteq S$, see Definition 10.

Proposition 17 *If Σ_1 is the binary part of the D-basis that has n implications, of a closure system on S and $|S| = m$, then there is an $O(mn + m^2)$ time algorithm that extracts $\Sigma' \subseteq \Sigma$ describing the cover relation of join-irreducible elements of closure lattice, and places implications of Σ' into a proper order. Under this order, $\rho_{\Sigma'}(y) = \rho_{\Sigma_1}(y)$, for every $y \in S$.*

Now we want to deviate slightly from the notion of ordered direct basis to the notion of *ordered direct sequence of implications*. Suppose Σ is some basis of a closure system $\langle S, \phi\rangle$. The ordered sequence $\sigma = \langle s_1, \dots, s_t\rangle$ of implications from Σ, not all necessarily different, is called *an ordered direct sequence from* Σ, if $\rho_\sigma(X) = \phi(X)$ for every $X \subseteq S$.

The idea of ordered direct sequencing allows some further optimization of the D-basis. If $Z = \langle z_1, \dots, z_k\rangle$ and $T = \langle t_1, \dots, t_s\rangle$ are two ordered sequences, then $Z^\frown T$ denotes their concatenation (the attachment of T at the end of Z).

Lemma 18 *Suppose $\sigma = \Sigma_1^\frown \Sigma_2^\frown \Sigma_3$ is an ordered direct sequence from some basis Σ, where Σ_1, Σ_3 consist of binary implications in proper order of Proposition 17, Σ_2 consists of non-binary implications, and Σ_2 can be put into arbitrary order without changing the ordered direct status. If $(A \to y), (A \to x) \in \Sigma_2$ and $(y \to x) \in \Sigma_1$, then $A \to x$ can be dropped from Σ_2 and replaced by an additional $y \to x$ in Σ_3.*

Corollary 19 *Suppose Σ_D is the D-basis of some closure system. Consider $\Sigma_D^+ \subseteq \Sigma_D$ obtained from Σ_D by performing the following reductions:*

(a) Remove $A \to x$, if $A \to y$ and $y \to x$ are also in Σ_D.

(b) Remove $z \to x$, if $z \to y$ and $y \to x$ are also in Σ_D.

Let Σ_1 be a the proper ordering of binary part of Σ_D^+ given in Proposition 17, and let Σ_3 be a subordering of this proper ordering on implications $y \to x$ that appear in triples of $A \to x, A \to y, y \to x$ of (a). Finally, let Σ_2 be some ordering of non-binary implications of Σ_D^+. Then $\sigma = \Sigma_1^\frown \Sigma_2^\frown \Sigma_3$ is the ordered direct sequence for the basis Σ_D^+. In particular, the length of this sequence is no longer than the length of the D-basis.

7 Closure systems without cycles and the E-basis

It turns out that the D-basis can be further reduced, when an additional property holds in a closure system $\langle S, \phi\rangle$. The results of this section follow closely exposition given in (Freese, Jezek, and Nation 1995), section 2.4.

We will write xDy, for $x, y \in S$, if $y \in Y$, for some minimal cover Y of x.

Definition 20 *A sequence $x_1, x_2, \dots, x_n$, where $n > 1$, is called a* D-cycle, *if $x_1 D x_2 D \dots x_n D x_1$. A finite closure system $\langle S, \phi\rangle$ is said to be* without D-cycles *if it has no D-cycles.*

We note that the closure systems without D-cycles is the weakening of the notion of *acyclic Horn formulas* well-known in the studies of Boolean functions, see, for example, (Hammer and Kogan 1995). In lattice-theoretical literature, the lattices without D-cycles are known as *lower bounded*.

For every $x \in S$, consider $M(x) = \{Y \subseteq S$:Y is a minimal cover of $x\}$. The family $\phi(M(x)) = \{\phi(Y) : Y \in M(x)\}$ is ordered by containment, thus, we can identify its minimal elements. Let $M^*(x) = \{Y \in M(x) : \phi(Y)$ is minimal in $\phi(M(x))\}$.

We will write xEy, for $x, y \in S$, if $y \in Y$ for some $Y \in M^*(x)$. According to the definition, if xEy then xDy. On the other hand, converse is not always true.

Now define two sequences of subsets of S, based on covers from $M(x)$ and $M^*(x)$, correspondingly.

Let $D_0 = E_0 = \{p \in S : p \in \phi(p_1, \dots, p_k)$ implies $p \in \phi(p_i)$, for some $i \leq k\}$. If D_k and E_k are defined, then $D_{k+1} = D_k \cup \{s \in S :$ if $s \sim Y$ then $s \sim Z$ for some $Z \subseteq D_k, Z \ll Y$ and $Z \in M(s)\}$. Similarly, $E_{k+1} = E_k \cup \{s \in S :$ if $s \sim Y$ then $s \sim Z$ for some $Z \subseteq E_k, Z \ll Y$ and $Z \in M^*(s)\}$. Apparently, $E_k \subseteq D_k$, for any k. The following result is proved in (Freese, Jezek, and Nation 1995), Theorem 2.51.

Lemma 21 *If $\langle S, \phi\rangle$ is a closure system without cycles, then, for some k, $S = E_k = D_k$.*

As a consequence, we can reduce D-basis to obtain a shorter basis for a closure system without cycles. We will say that $s \in S$ has D-rank $k = 0$, if $s \in D_0$, and $k > 0$, if $s \in D_k \setminus D_{k-1}$. According to Lemma 21, every $s \in S$ in a closure system without cycles has a D-rank.

Theorem 22 *Let $\langle S, \phi\rangle$ be a reduced closure system without D-cycles. Consider a subset Σ_E of the D-basis that is the union of two sets of implications:*

(1) $\{y \to x : x \in \phi(y)\}$,

(2) $\{X \to x : X \in M^*(x)\}$.

Then

(a) Σ_E is a basis for $\langle S, \phi\rangle$.

(b) Σ_E is ordered direct.

(c) The aggregated form of Σ_E is ordered direct.

Proof To begin with, it is not true that every cover of an element $x \in S$ refines to a cover in $M^*(x)$, so Σ_E must be ordered more carefully than Σ_D. Nonetheless, mimicking the proof of Theorem 2.50 of (Freese, Jezek, and Nation 1995), we can construct an order on Σ_E that makes it an ordered direct basis. This will be done for the aggregated E-basis, proving parts (a) and (c) simultaneously; part (b) then follows.

Consider the aggregated form of Σ_E. Given an implication $X \to Y$ in this basis, let $D^*(X \to Y)$ be the maximal D-rank of elements in X, and $D_*(X \to Y)$ be the minimal D-rank of elements in Y. Then $D^*(X \to Y) < D_*(X \to Y)$.

Order the implications following the rule: put the implications $x \to Y$ first (aggregated form of binary part of Σ_E), and for the rest, if $D^*(X_1 \to Y_1) < D^*(X_2 \to Y_2)$ then $X_1 \to Y_1$ precedes $X_2 \to Y_2$ in the order.

Claim. If $X_1 \to Y_1$ and $X_2 \to Y_2$ are in the aggregated E-basis, and $x \in Y_1 \cap X_2$, then $X_1 \to Y_1$ precedes $X_2 \to Y_2$.

Indeed, if the D-rank of x is k, then $D^*(X_2 \to Y_2) \geq k \geq D_*(X_1 \to Y_1) > D^*(X_1 \to Y_1)$. Hence, $X_1 \to Y_1$ will appear in the order before $X_2 \to Y_2$.

Now take any input set Z. We want to show that $\phi(Z)$ can be obtained when applying the aggregated basis in the described order. We argue by induction on the rank of an element $z \in \phi(Z) \setminus Z$.

If $z \in D_0$, then it only can be obtained via some implication $x \to Y$, for some $x \in Z$, and $z \in Y$, and implications $x \to Y$ form an initial segment in the ordered sequence of the basis. Now assume that it is already proved that all elements of $\phi(Z) \setminus Z$ of rank at most k can be obtained in some initial segment of the sequence for the basis. If we have now element z of rank $k+1$, then it can be obtained via an implication $X \to Y$ with $X \subseteq \phi(Z)$, $z \in Y$, and $D^*(X) < k+1$. By the induction hypothesis, all elements in $X \subseteq \phi(Z) \setminus Z$ can be obtained via implications located in some initial segment of the sequence, and by the Claim above, all those implications precede $X \to Y$. Thus, all implications producing elements of rank $k+1$ from $\phi(Z)$ will be located after the segment of the sequence producing all rank k elements.

Proposition 23 *Suppose $\Sigma_D = \{s_1, s_2, \dots, s_n\}$ is a D-basis of some closure system $\langle S, \phi\rangle$ and $|S| = m$. It requires time $O(mn^2)$ to determine whether the closure system is without cycles, and if it is, to build its ordered direct basis Σ_E.*

Proof Since the D-relation is a subset of S^2, it will contain at most m^2 pairs. On the other hand, it is built from implications $X \to x$, so the other upper bound for pairs in D-relation is mn. Evidently, the closure system is without D-cycles iff its D-relation can be extended to a linear order. There exists an algorithm that can decide whether $\langle S, D\rangle$ can be extended to a partial order on S in time $O(m + |D|)$, see Theorem 11.1 in (Freese, Jezek, and Nation 1995). We will see below that the rest of the algorithm will take time $O(mn^2)$, which makes the total time also $O(mn^2)$.

Assuming the first part of algorithm provides a positive answer and there are no D-cycles, we proceed by finding the ranks of all elements. It will take at most n operations to find set D_0: include p into D_0, if it does not appear as a conclusion in any (non-binary) implication $X \to x$ of the D-basis, where $x \triangleleft X$. If the system is without D-cycles, then $\pi_\Sigma(D_0) \setminus D_0$ gives elements of rank 1, $\pi^2_\Sigma(D_0) \setminus \pi(D_0)$ elements of rank 2, etc. Note that $\pi_\Sigma(X)$ is defined in the beginning of section 6. Computation of $\pi_\Sigma(X)$ requires n steps, since $\Sigma = \Sigma_D$ in our case has n implications. After at most m iterations of π, one would obtain the whole S, whence, $O(mn)$ operations are needed to obtain the ranks of all elements from S.

It remains to decide which implications from the D-basis should remain in the E-basis. To that end, for each element $x \in S$ we need to compare the closures $\phi(X)$ of subsets X, for which $X \to x$ is in the D-basis, and choose for the E-basis those that are minimal. There is at most n implications $X \to x$, for a given $x \in S$, and the closure $\phi(X)$, for each such X, can be found in $O(s(\Sigma_D))$ steps. It will take time $O(n^2)$ to determine all minimal subsets among $O(n)$ given subsets $\phi(X)$, associated with fixed $x \in S$. Hence, it will require time $O(mn^2)$ for all $x \in S$.

The size of the E-basis will be at most n, and it will take time $O(n^2)$ to order it with respect to the rank of elements, per Corollary 22.

When a closure system has cycles, subset Σ_E of Σ_D, defined in Corollary 22, may not form a basis. Further results about closure systems without D-cycles, and more generally systems whose closure lattice is join semidistributive, will be presented in (Adaricheva and Nation 2011).

8 D-basis versus Duquenne-Guigues canonical basis

We recall the definition of the canonical basis introduced iin (Guigues and Duquenne 1986), see also (Bertet and Monjardet 2010).

Definition 24 *The canonical basis of a closure system (S, ϕ) consists of implications $X \to Y$ for $X, Y \subseteq S$, that satisfy the following properties:*

(1) $X \subset \phi(X) = Y$;

(2) for any ϕ-closed set Z, either $X \subseteq Z$ or $Z \cap X$ is ϕ-closed;

(3) if $W \subseteq X$, $\phi(W) = Y$ and W satisfies (2) in place of X, then $W = X$.

The subsets $X \subset S$ with properties (1) and (2) are usually called *quasi-closed*, see (Caspard and Monjardet 2003). The meaning of (2) is that adding X to the family of closed sets of ϕ produces the family of closed sets of another closure operator. Property (3) indicates that among all quasi-closed subsets with the same closure one needs to choose the minimal ones. This basis is called *canonical*, since it is minimum, in that no implication can be removed from it without altering ϕ, and every other minimum implicational basis for ϕ can be obtained from it. In particular, no other basis can have a smaller number of implications. Note that here the implications are of the form $X \to Y$, where Y is not necessarily one-element set.

To bring this basis in comparison with other bases discussed in this paper, each implication $X \to Y$ may be replaced by set of implications $X \to y$, $y \in Y \setminus X$. We will call this modification of the canonical basis the *unit expansion*, or the *D-G unit basis*.

It turns out that, in general, the canonical basis cannot be ordered; thus, it is not ordered direct. Running a computer program that checked about a million of various closure systems on 5- and 6-element sets revealed several examples of closure systems on a 6-element set, for which the canonical basis cannot be ordered. In some of them, the unit expansion of the basis does admit an ordering to make it direct, but others show that some canonical bases cannot be ordered in either form.

Proposition 25 *There exists no algorithm to order the canonical basis of an arbitrary closure system turning it into an ordered direct basis.*

Example 26

Let (S, ϕ) be a closure system on $S = \{1, 2, 3, 4, 5, 6\}$, given by the family of closed sets:
$\{\emptyset, 1, 2, 3, 5, 6, 12, 13, 14, 16, 23, 123, 124, 135, 256, 1346, S\}$.

The canonical basis has 9 implications:
$4 \to 1, 15 \to 3, 35 \to 1, 25 \to 6, 56 \to 2, 26 \to 5, 36 \to 14, 134 \to 6, 146 \to 3$.

There is a single implication $36 \to 14$ that can be expanded to two unit implications $36 \to 1$ and $36 \to 4$.

The proof that the unit expansion of canonical basis cannot be ordered to make it direct, follows from consideration of the next three closures:

- $45 \to 145 \to 1345 \to 13456 \to S$, hence, $134 \to 6$ should be placed later than $15 \to 3$.
- $1234 \to 12346 \to S$, hence $26 \to 5$ should be placed later than $134 \to 6$.
- $126 \to 1256 \to 12356 \to S$, hence $15 \to 3$ should be placed later than $26 \to 5$, which contradicts the previous two constraints.

For comparison, the aggregated D-basis has 15 implications:
$4 \to 1, 45 \to 26, 36 \to 14, 34 \to 6, 15 \to 3, 46 \to 3, 35 \to 1, 25 \to 6, 26 \to 5, 56 \to 2, 126 \to 34, 235 \to 4, 156 \to 4, 234 \to 5, 125 \to 4$.

Thus, one run of aggregated D-basis (15 implications) wins over two runs (18 implications) of canonical basis.

In this example, the D-basis is 4 implications shorter than the canonical direct unit basis that has 22 implications.

The performance of D-basis in comparison with the D-G unit basis and canonical direct unit basis was tested on $300,000$ randomly generated closure systems on base sets of 6 and 7 elements.

The primary advantage of the D-basis is its ordered directness. By contrast, computing the closure of a non-closed set using the canonical basis will always take at least two iterations: the final pass produces nothing and exists solely to determine that the ability of the basis to expand the given set has been exhausted.

In the testing on domain length 6, with inputs sets of length 3, the D-G unit basis cycled through, on average, 22.9 implications before returning the closure. By comparison, the direct canonical basis took 15.8 such steps and the D-basis took only 12.7 checks on average. Due to their ordered directness, the number of implications checked in the direct optimal and D-basis was equivalent to the number of implications they contained.

9 Processing of ordered basis versus forward chaining algorithm

In this section we look more closely at the algorithmic aspects of finding the closures of input sets using the *ordered basis* approach in comparison to the well-known *forward chaining algorithm*, introduced in (Dowling and Gallier 1984) for checking the satisfiability of Horn formulas. As we pointed out earlier, these seemingly different tasks can be seen equivalent, when interpreting Horn formulas as implications. The ordered basis algorithm can be applied to any basis that is ordered direct. In our case, we did the comparisons using the D-basis as an input basis for both algorithms.

We will assume that the base set is $S = \{x_1, \dots, x_n\}$, which can be interpreted as propositional variables, and the closure system is given by a unit basis Σ with m implications.

The forward chaining procedure effectively requires two runs of the given basis. In the first (setup) pass, it constructs the *ClauseList*, *Propositions*, and *Consequent* arrays, along with the queue of *True* elements thought of as an input set.

Here *ClauseList* is the set of arrays X_i, for each propositional variable x_i, that keep the indices of all clauses (implications) in which x_i appears as a negative literal (or, equivalently, appears on the left-hand side of the implication). *Propositions* is an array of size m: for every implication s_i, it has the number of propositional variables in its left-hand side that remain to be evaluated to true. Also, *Consequent* is an m-element array that, for every clause s_i, points to a proposition x_j that appears on its right-hand side.

In the second run, to actually compute the closure, it pops a True element, say, x_i from the queue, addresses the corresponding X_i in *ClauseList*, and, for each pointer to a clause/implication there, it decrements by 1 the corresponding entry of *Propositions*. Whenever the entry in *Propositions* reaches 0, the corresponding implication is ready to be processed, which means taking the corresponding variable in *Consequent* and putting it into queue of *True* elements.

Since every entry of *Propositions* will, in the worst case, be reduced to zero, the number of steps in computing the closure is bounded by the size $s(\Sigma)$, i.e., the combined length of the implications in the basis. Including the pre-processing steps, the forward chaining algorithm should require $O(s(\Sigma))$ operations to compute the closure. If the closures of multiple sets are to be performed, of course, the setup steps can be abbreviated: only *Propositions* and *True* need be updated for subsequent runs.

Finding the closure of a given set from the D-basis is similarly linear in the basis length. Each implication s_i is recorded as an array X_i where the zero index contains the positive literal and the following indices contain the negative literals. We check each literal against a boolean array of length n which stores a 1 at the index i corresponding to a true literal x_i. If any negative literal corresponds to a 0 (false) in the array, we move on to the following implication. If all the negative literals in a given implication are found to be true, we set the value in the boolean array that corresponds to its positive literal to true. Since in the worst case, we process each literal of every clause exactly once, this procedure takes $O(s(\Sigma))$ operations to compute the closure.

In our testing of ordered-basis versus forward-chaining performance, we evaluated the time spent on finding the closures of random subsets of the domain by computing the number of elementary operations (setting a variable, comparing Boolean values, etc.) performed during the processing. The inputs for both algorithms were identical.

Test data bears out both complexity estimates. In 100,000 runs of randomly generated closure systems and their D-bases from the domain $S = \{1, 2, 3, 4, 5\}$, the average length $|\Sigma_D|$ was 18.99. It took an average of 59.82 simple operations (33.71 when not counting the pre-processing time) to find the closure of an arbitrary input set in forward chaining procedure compared to 34.35 simple operations to compute the closure using the linear processing of the D-basis, with similar values for larger domains (see the table in Figure 3). Taking all computing overhead into account then, it takes very close to $2|\Sigma_D|$ and $3|\Sigma_D|$ operations for the ordered-basis and forward-chaining respectively.

Noticeably, the ordered-basis approach does not actually require the representation of propositions as $S = \{x_1, \ldots, x_n\}$ and implications as $\Sigma = \{s_1, \ldots, s_m\}$, where each proposition has an associated integer value, necessary for indexing and traversing *ClauseList*, *Propositions*, and *Consequent*. Though we take advantage of the integer value in constructing the array of true values, we could easily substitute that array with a simple set of satisfied propositions. By contrast, to use the forward chaining method on a basis without this representation would require significant overhead in hashing each proposition to its corresponding integer.

Length	Basis Size	Forward Chaining	Chaining (pre-processed)	D-basis
4	8.02	29.39	17.38	16.14
5	18.99	59.82	33.71	34.35
6	35.04	105.47	58.60	60.62
7	56.91	169.34	93.30	95.86

Figure 1: Comparison

Additionally, the ordered-basis approach eliminates the need for pre-processing of the basis to store it in the form of *ClauseList* and *Consequent*. This is particularly important when the basis may not fit into main memory. Instead of having to individually access each array X_i in *ClauseList* when the propositional variable x_i appears at the head of the queue, the ordered-basis approach allows us to parse the basis in conveniently sized pieces.

There is at least one observation how the idea of the ordered basis may improve the performance of existing forward chaining algorithm. Indexing the implications according to the proper order of the D-basis, whenever we add a positive literal to the *True* queue we may additionally maintain the index i of the implication from which it was derived. Then, when we process the literal, we only need to update j-entries of *Propositions* where $j \geq i$, saving us significant processing time for very large sets.

Acknowledgments. We are grateful to several people who read the draft of this paper at different stages of its preparation and made valuable comments: György Turán, Marina Langlois, Ralph Freese, Marcel Wild and Karell Bertet, also students of Yeshiva College Joshua Blumenkopf and Jeremy Jaffe. Major advancements of this paper were done during the first author's visits to University of Hawaii in 2010-2011, supported by the AWM-NSF Mentor Travel grant. The welcoming atmosphere at the Department of Mathematics at UofH is greatly appreciated.

References

Adaricheva, K., and Nation, J. B. 2011. On bases of closure systems without d-cycles. Forthcoming.

Bertet, K., and Monjardet, B. 2010. The multiple facets of the canonical direct unit implicational basis. *Theoretical Computer Science* 411(4):2155–2166.

Caspard, N., and Monjardet, B. 2003. The lattices of closure systems, closure operators, and implicational systems on a finite set: a survey. *Disc. Appl. Math.* 127:241–269.

Crama, Y., and Hammer, P. 2011. *Boolean functions: Theorey, Algorithms and Applications*. Cambridge University Press.

Dowling, W., and Gallier, J. 1984. Linear-time algorithms for testing the satisfiability of propositional horn formulae. *J. Logic Programming* 3:267–284.

Freese, R.; Jezek, J.; and Nation, J. B. 1995. *Free lattices*, volume 42 of *Mathematical Surveys and Monographs*. Providence, RI: American Mathematical Society.

Guigues, J. L., and Duquenne, V. 1986. Familles minimales d'implications informatives résultant d'un tableau de données binaires. *Math. Sci. Humaines* (95):5–18, 83.

Hammer, P. L., and Kogan, A. 1995. Quasi-acyclic propositional horn knowledge bases: optimal compression. *IEEE Transactions on knowledge and data engineering* 7(5):751–762.

Kleine Buning, H., and Lettman, T. 1999. *Propositional logic: deduction and algorithms*. Cambridge, Massachusetts: Cambridge University Press.

Nation, J. B. 1990. An approach to lattice varieties of finite height. *Algebra Universalis* 27(4):521–543.