Applied Mathematics & Information Sciences
*An International Journal*

http://dx.doi.org/10.12785/amis/092L05

# From Implicational Systems to Direct-Optimal bases: A Logic-based Approach.

*E. Rodríguez-Lorenzo*[1,*], *K. Bertet*[2], *P. Cordero*[1], *M. Enciso* [1], *A. Mora*[1] *and M. Ojeda-Aciego*[1]

[1] Universidad de Málaga, Málaga, Spain
[2] Laboratoire 3I, Université de La Rochelle, France

Received: 5 Jul. 2014, Revised: 6 Oct. 2014, Accepted: 7 Oct. 2014
Published online: 1 Apr. 2015

**Abstract:** Due to its solid mathematical foundations, Formal Concept Analysis (FCA) has become an emergent topic in the area of data analysis and knowledge discovering. Information is represented in a binary table defining a relation between a set of objects and a set of attributes—the formal context. The knowledge extracted from the formal context allows to identify useful patterns in data in different forms. One very useful knowledge representation in FCA are implications among attributes which are validated over the objects. The most outstanding feature of implications is that they can be managed by means of inference systems. Equivalent sets of implications can be obtained using different logic-based transformations. The aim of these transformations is to turn the original set of implications into an equivalent one fulfilling some desired properties. Among them, the directness and optimality are very popular targets because getting a direct-optimal basis ensures that the closure of a set of attributes may be computed with lower cost (time and resources). In this work, we introduce a new method to compute the direct-optimal basis which improves the existing ones. The new method reduces the input in a first stage and is guided by the idea of limiting the growth of the intermediate sets of implications as a way to improve the performance. We illustrate the good features of the new method with both a detailed example and by experimental evaluation.

**Keywords:** Formal Concept Analysis, Implications, Basis, Logic

## 1 Introduction

From the mid 1980s, Formal Concept Analysis (FCA) has become a useful tool for data analysis. One of the reasons of its success is the power of lattice theory which underlies FCA. Information is represented by a binary table defining a relation between a set of objects and a set of attributes, and a well established set of methods and techniques allows to extract knowledge by means of automated tools. The main goal of FCA is to infer *concepts* from the data set, i.e. to deduce (in an automated way) a set of objects that may be precisely characterized by a set of attributes. Such concepts inherit an order relation induced by attribute set inclusion, providing a lattice structure of the concept set.

The concept lattice allows to identify patterns in data, which is one of the main issues in Knowledge Discovery in databases. Moreover, FCA has been used in different areas: Artificial Intelligence, Databases, Software Engineering, Data Mining, and recently it is becoming to be a suitable tool in the Semantic Web.

One of the most relevant patterns which can be extracted from a formal context is an implication, representing a relation between subsets of attributes. Implications are strongly connected with the concept lattice, providing an alternative representation of the underlying information.

Belohlavek et al. [17] focus on the importance of implication when stating “A distinguishing feature of FCA is an inherent integration of three components: discovery of clusters (so-called formal concepts) in data, discovery of data dependencies (so-called attribute implications) in data, and visualization of formal concepts and attribute implications by a single hierarchical diagram (so-called concept lattice)”.

* Corresponding author e-mail: estrellarodlor@ctima.uma.es

© 2015 NSP
Natural Sciences Publishing Cor.

We recall here an example from [10, pp. 30 and 84] which will be used later as a running example[1]. Six attributes are considered representing supranational organizations: Gr77 (Group of 77), NA (Non-aligned), LLDC (Least Developed Countries), MASC (Most Seriously Affected Countries), OPEC (Organization of Petrol Exporting Countries) and ACP (African, Caribbean and Pacific Countries).

In FCA, when all the objects satisfying a subset of attributes also satisfy another subset of attributes, there exists a dependence between both sets, which is called an *implication*. In Table 1, one notices that every country belonging to the OPEC, also belongs to both, to the NA organization and to the Gr77. This knowledge is captured by an attribute implication written in the form:

$$\text{OPEC} \rightarrow \text{NA, Gr77}$$

One of the most relevant issues related with attribute implications is that it may be managed by using an inference system. It opens the doors to the design of automated deduction methods to deal with sets of implications: thus, an interesting research topic is to find methods which transform a set of implications into another one, semantically equivalent but more suitable for further treatment. In this framework, the present paper focuses on the design of logic-based transformations for sets of implications which render an equivalent sets but fulfilling desired properties, *directness* and *optimality*.

In FCA many of the problems are solved by intensively computing the closure of a set of attributes. Although classical closure algorithms have linear complexity, a reduction in the execution cost is relevant when a huge number of closures is necessary. Rudolph states in [20] that "one central task when dealing with closure operators is to represent them in a succinct way while still allowing for their efficient computational usage".

Particularly, it would be useful to have minimal representations of implicational theories. Duquenne et al. [15] stress the need "to put data in canonical forms to speed up access to information and of extracting classifications and rules providing some explanation".

One can find in the literature several interesting contributions to the problem of the minimal generation of a closure operator by an implicational system: the first one was introduced independently by Maier [6] and Duquenne-Guigues [11], leading to the currently so-called *Duquenne-Guigues canonical basis* or *stem basis*. Bertet and Monjardet [4] surveyed five of the results about minimal implicational systems, and proved that all of them were equivalent, introducing the term *direct-optimal implicational basis*. Adaricheva et al. [14] proposed the so-called *ordered direct basis* in order to improve the efficiency of the direct-optimal implicational basis.

The notion of direct-optimal implicational basis has two very interesting properties: its size can be proved to be the least (*optimality*), and the closure of any subset of attributes can be computed in just one traversal of the implicational set (*directness*).

Working with arbitrary implicational systems, allows for considering inputs of smaller size, but the method given in [3] to compute direct-optimal basis can be exponentially hard, whereas considering unitary implicational systems enhances the performance of the method [5] but the inconvenient is that the inputs can be much bigger.

In this paper, after summarizing the methods to compute the direct-optimal basis proposed in [3] for arbitrary implicational systems, and in [5] for unitary implicational systems, we propose a new alternative method which combines the good properties of both approaches.

The paper is organized as follows, Section 2 summarizes the required background for the rest of the paper. A brief summary of previous methods to compute the direct-optimal basis are given in Section 3. Later, Section 4 deals with the main contribution of the paper: we propose a new method based on the idea of removing superfluous attributes to efficiently deal with arbitrary implications. Experimental evaluation has been performed by using a Prolog implementation and the results are outlined in Section 5. Some conclusions and future works are given in Section 6.

## 2 Background

In this section we present a brief introduction of preliminaries of FCA, the logic for implications, and implicational systems in three separated subsections.

### *2.1 Formal Concept Analysis*

In Formal Concept Analysis [10], the notion of formal context is defined to be a triplet $\mathbb{K} := (G, M, I)$ where $G$ is a set of objects, $M$ is a set of attributes and $I$ is a binary relation between $G$ and $M$. For $g \in G$ and $m \in M$, we write $\langle g, m \rangle \in I$ if the object $g$ has the attribute $m$.

*Example 1.*Table 1 from the previous section depicts the binary relation $I$ of a formal context $\mathbb{K}_0$ where the set of objects and the set of attributes are, respectively, $G = \{$Afghanistan, Algeria, Benin, Botswana, Cameroon, Gabon, Haiti, Kiribati$\}$ and $M = \{$Gr77, NA, LLDC, MASC, OPEC, ACP$\}$. □

From this triple, two mappings $\uparrow\colon 2^G \rightarrow 2^M$ and $\downarrow\colon 2^M \rightarrow 2^G$, named concept-forming operators, are defined as follows: for any $X \subseteq G$ and $Y \subseteq M$,

$$X^{\uparrow} = \{m \in M \mid \langle g, m \rangle \in I \text{ for all } g \in X\}$$

[1] In [10], 130 countries were considered but we have reduced it to an equivalent 8 object table, in the sense that the resulting concept lattice is isomorphic to the original one.

© 2015 NSP
Natural Sciences Publishing Cor.

**Table 1:** Membership of countries in supranational groups.

| $I$ | Gr77 | NA | LLDC | MASC | OPEC | ACP |
|---|---|---|---|---|---|---|
| Afghanistan | × | × | × | × | | |
| Algeria | × | × | | | × | |
| Benin | × | × | × | × | | × |
| Botswana | × | × | × | | | × |
| Cameroon | × | × | | × | | × |
| Gabon | × | × | | | × | × |
| Haiti | × | | × | × | | × |
| Kiribati | | | × | | | × |

$$Y^{\downarrow} = \{g \in G \mid \langle g, m\rangle \in I \text{ for all } m \in Y\}$$

These mappings establish a Galois connection and the composition of the two concept-forming operators gives us two closure operators, i.e $X^{\uparrow\downarrow}$ and $Y^{\downarrow\uparrow}$ are the closures of $X$ and $Y$, respectively. And, the closed sets of these two mappings, that is, the fixpoints of the closure operators, define the so-called *formal concepts*. As we shall see, formal concept is a key point in FCA which formally describes an *idea* of the model and it allows us to characterize a set of objects by means of the attributes they share, and vice versa.

*Example 2.*There are 26 concepts associated with the formal context $\mathbb{K}_0$ introduced in Table 1. For instance, the set of attributes {LLDC, ACP} and the set of objects {Benin, Botswana, Haiti, Kiribati} are two closed sets which describe the notion of the least developed countries in a certain region providing its properties and characterizing its countries; i.e.:

$$\{\text{Afghanistan, Benin, Botswana, Haiti}\}^{\uparrow\downarrow} = \\ = \{\text{Afghanistan, Benin, Botswana, Haiti}\}$$

$$\{\text{LLDC, ACP}\}^{\downarrow\uparrow} = \{\text{LLCD, ACP}\}$$

□

A formal concept is a pair $\langle X, Y\rangle$ such that $X \subseteq G$, $Y \subseteq M$, $X^{\uparrow} = Y$ and $Y^{\downarrow} = X$ where $X$, named the *extent*, and $Y$, named the *intent*, are closed sets of objects and attributes. The set of formal concepts is known to be a complete lattice, the concept lattice associated to the context with the following partial ordering:

$$\langle X_1, Y_1\rangle \leq \langle X_2, Y_2\rangle \text{ if and only if } X_1 \subseteq X_2$$

$$\text{(or equivalently } Y_1 \supseteq Y_2\text{)}$$

*Example 3.*The concept lattice associated with the formal context $\mathbb{K}_0$ in Table 1 is depicted in Figure 1. □

In FCA, a concept lattice can be defined dually using *attribute implications*, which can be deduced from the concept lattice or using mining techniques from the context as well. An attribute implication is an expression $A \to B$ where $A$ and $B$ are subsets of attributes, i.e. $A, B \subseteq M$. A context satisfies $A \to B$ if every object that has all the attributes in $A$ also has all the attributes in $B$.

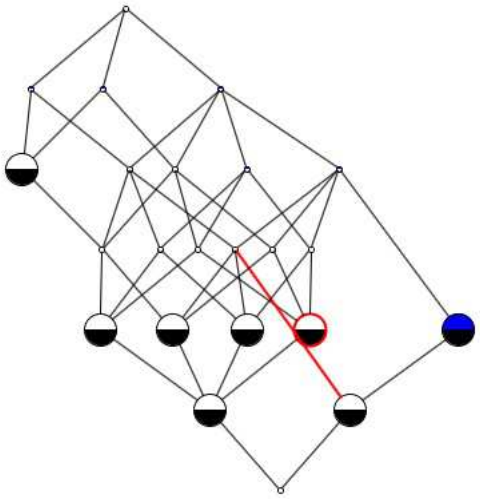

**Fig. 1:** Concept lattice of $\mathbb{K}_0$ formal context.

**Definition 1.***Let $\mathbb{K}$ be a formal context and $A, B \subseteq M$ subsets of attributes, an implication is an expression as $A \to B$ and it is said that it holds (is valid) in $\mathbb{K}$ whenever $A^{\downarrow} \subseteq B^{\downarrow}$.*

*Example 4.*The implication

$$OPEC \to Group\,77, Non\text{-}Aligned$$

holds in $\mathbb{K}_0$ because every object that belongs to the OPEC also belongs to Gr77 and NA supranational organizations (see Table 1). □

As we have mentioned above, although attribute implications and concept lattices may be considered dual expressions of the knowledge system, the former has an outstanding property since they can be managed syntactically by means of logic. The pioneering logic-based approach of Armstrong's Axioms to deal with dependencies was originally defined for functional dependencies. Though funcional dependencies and attribute implications have different interpretations, they share the same notion of semantic entailment [2].

© 2015 NSP
Natural Sciences Publishing Cor.

## 2.2 Logic for attribute implications

We begin this section with a formal description of Armstrong's Axioms, the first sound and complete inference system to deal with attribute implications [1]. The language for attribute implications, $\mathscr{L}$, is defined as follows

**Definition 2(Language).** *Given a non-empty finite alphabet M (named attribute set), the language is the following set of implications or formulas*

$$\mathscr{L} = \{A \to B \mid A, B \subseteq M\}$$

In order to distinguish between language and metalanguage, inside implications, $AB$ means $A \cup B$ and $A$-$B$ denotes the set difference $A \setminus B$. We will use capital letters to denote subsets of attributes and lower-case letters for singleton sets of attributes. Moreover, when no confusion arises, we omit the brackets, e.g. $abc$ denotes the set $\{a,b,c\}$.

As usual, we introduce now an interpretation of these formulas and their corresponding models. So, a formal context $\mathbb{K}$ is a *model* of a formula $A \to B$ if it is valid in $\mathbb{K}$ as per Definition 1. A *theory* is a set of formulas and, if $\Sigma$ is a theory, $\Sigma \models A \to B$ denotes that every model of all formulas in $\Sigma$ is also a model of $A \to B$.

Implications can be syntactically managed by means of the following inference system:

**Definition 3(Armstrong's Axiom System).** *Given the language $\mathscr{L}$, the inference system $\mathscr{S}_{Ar}$ has one axiom and two inference rules:*

$$\texttt{[Ax]}\ \frac{A \supseteq B}{A \to B} \qquad \texttt{[Augm]}\ \frac{A \to B}{AC \to BC} \qquad \texttt{[Trans]}\ \frac{A \to B,\ B \to C}{A \to C}$$

We use the standard notation for syntactic derivation. Given a theory $\Sigma$ and a formula $A \to B$, $\Sigma \vdash A \to B$ denotes that $A \to B$ can be derived from $\Sigma$ by using the axiom system.

Simplification Logic, $\mathbf{SL}_{\text{FD}}$ [7], has shown to be an alternative axiom system to Armstrong's. $\mathbf{SL}_{\text{FD}}$ is an executable and useful tool to manipulate implications guided by the idea of simplifying the set of implications by efficiently removing redundant attributes.

**Definition 4(Simplification Axiom System).** $\mathbf{SL}_{\text{FD}}$ *considers reflexivity as axiom scheme and the following inference rules named fragmentation, composition and simplification respectively.*

$$\texttt{[Ref]}\ \frac{}{A \to A}$$

$$\texttt{[Frag]}\ \frac{A \to BC}{A \to B}$$

$$\texttt{[Comp]}\ \frac{A \to B,\ C \to D}{AC \to BD}$$

$$\texttt{[Simp]}\ \frac{A \to B,\ C \to D}{C\text{-}B \to D\text{-}B}, (A \subseteq C \text{ and } A \cap B \neq \emptyset)$$

Both axiom systems are sound and complete and, therefore, equivalent [7].

The rules of $\mathbf{SL}_{\text{FD}}$ have been used as the engine of a set of automated reasoning methodd to manipulate implications developed in [8, 9, 12, 13]. One of the most important problems related with implications is the computation of the closure of a set of attributes. Thus, given a set of implications $\Sigma$, valid in a formal context, and a subset of attributes $A \in 2^M$, the problem is the computation of the biggest subset $A^+ \in 2^M$ such that $A \to A^+$ holds in the formal context. This problem was tackled in [8] developing a logic-based method by means of a set of equivalence rules derived from the above inference rules.

We emphasize that $\mathbf{SL}_{\text{FD}}$ inference rules can be considered equivalence rules and are sufficient to compute all the derivations [12].

**Theorem 1( [12]).** *In $\mathbf{SL}_{\text{FD}}$, the following equivalencies hold:*

1. *Fragmentation Equivalency* **[FrEq]***:*

$$\{A \to B\} \equiv \{A \to B\text{-}A\}$$

2. *Composition Equivalency* **[CoEq]***:*

$$\{A \to B, A \to C\} \equiv \{A \to BC\}$$

3. *Simplification Equivalency* **[SiEq]***: If $A \cap B = \emptyset$ and $A \subseteq C$ then*

$$\{A \to B, C \to D\} \equiv \{A \to B, C\text{-}B \to D\text{-}B\}$$

The reading from left to right of these rules gives the essence of $\mathbf{SL}_{\text{FD}}$. In this direction, these equivalencies remove redundant information: redundant attributes (**[FrEq]**, **[SIEq]** or redundant implications (**[CoEq]**). This was the core of $\mathbf{SL}_{\text{FD}}$ when it was conceived.

We emphasize that $\mathbf{SL}_{\text{FD}}$ adequately deals with arbitrary set of implications, particularly with non-unitary formulas.

A set of implications is said to be an *Implicational System* (IS), which is defined as a binary relation between $2^M$ and $2^M$. An IS in which any implication has a singleton set of attributes in its right-hand side ($A \to b$) is named *Unit Implicational System* (UIS). That is, an UIS is a binary relation between $2^M$ and $M$. Obviously, by using **[CoEq]**, any IS can be transformed into an equivalent UIS and vice versa. Most of the existing methods in the literature assume that inputs are UIS, which induces a significant growth of the original set, worsening its further treatment. However, methods based on $\mathbf{SL}_{\text{FD}}$ do not require any normalization preprocessing. The method introduced in this paper is a new example of this good behavior.

## 2.3 Implicational Systems

In this section we present particular implicational systems, satisfying some properties, which are very

© 2015 NSP
Natural Sciences Publishing Cor.

interesting for applications because its shape eases its automated management. Among the most outstanding properties presented in [4], the authors emphasize *minimality*. An IS $\Sigma$ is *minimal* if no implication may be removed from $\Sigma$ without losing equivalence.

Notice that there exist several minimal ISs which may be equivalent. It is interesting to consider those ones having the least number of implications: the so-called *minimum* IS.

On the other hand, another interesting property can be based on the total number of attributes in the IS, and this leads to the notion of optimality, that is, an IS $\Sigma$ is said to be *optimal* if there does not exist another equivalent IS with smaller number of attributes.

Thus, in this direction, one of the main trend topics in FCA is the following: given a formal context $\mathbb{K}$, to obtain an IS $\Sigma$ such that the following conditions are fulfilled:

–all the implications in $\Sigma$ hold in $\mathbb{K}$ (*correctness*)
–every valid implication in $\mathbb{K}$ can be deduced from $\Sigma$ by using a sound and complete inference system (*completeness*)
–if any implication is removed from $\Sigma$ then the new IS $\Sigma'$ is not equivalent to $\Sigma$ (*minimality*)

An IS $\Sigma$ with these properties is called a *basis* of $\mathbb{K}$. When a property $P$ is added to a basis $\Sigma$, then the IS $\Sigma$ is called a *P basis*.

*Example 5.*Let $\mathbb{K}_0$ be the formal context presented in Table 1. A basis associated to $\mathbb{K}_0$ is the following:

$\Sigma = \{$ OPEC $\rightarrow$ Gr77 NA
MASC $\rightarrow$ Gr77
NA $\rightarrow$ Gr77
Gr77 NA MASC OPEC $\rightarrow$ LLDC ACP
Gr77 NA LLDC OPEC $\rightarrow$ MASC ACP$\}$ □

The above basis was provided in [10] and it corresponds to the so-called Duquenne-Guigues (or stem) basis [11], strongly based on the notion of pseudo-intent. As shown in [8], redundant attributes may appear in this kind of minimal basis and it is possible to define another equivalent basis with smaller size, although the new basis loses the property for Duquenne-Guigues basis. Thus, for different applications, it could be interesting to consider different properties. For instance, in [9] a basis with minimal size in the left-hand side of the implications was proposed.

In this work our target is the notion of directness (see Section 3), strongly related with the closure problem. In FCA, the computation of the closure of a set of attributes is an important topic. We emphasize the proposals in [3, 5, 12] for this well-known problem. Bertet et al. [4] summarized the works around the notion of direct bases presented in the literature. In that paper they relate directness with another interesting property: optimality. Furthermore they established the uniqueness of the direct-optimal basis. They also studied this notion of basis for both, IS and UIS.

In the next section, we summarize the works of [3, 5] to obtain the direct-optimal basis associated to a formal context.

## 3 Methods to compute the Direct Optimal basis

This section establishes the good properties of a direct-optimal basis, i.e. the computation of the closure can be done only in just one iteration and, due to its minimal size, the number of visited implications is reduced to the minimum. This situation makes the problem of building a direct-optimal basis one of the outstanding problems in FCA. Now, we will formally introduce these notions:

**Definition 5(Direct IS).** *An IS $\Sigma$ is said to be* direct *if, for all $X \subseteq M$:*

$$X^+ = X \cup \{b \in B | A \subseteq X \text{ and } A \rightarrow B \in \Sigma\}$$

It is worth noticing that if an IS $\Sigma$ is direct for a set of attributes $M$, the closure is obtained at cost $O(|\Sigma|)$ instead of $O(|\Sigma| \cdot |M|)$ ,which is the complexity in the worst case of classical closure algorithms.

**Definition 6(Direct-optimal IS).** *A direct IS $\Sigma$ is said to be* direct-optimal *if, for any direct IS $\Sigma'$ equivalent to $\Sigma$ we have that $\|\Sigma\| \leq \|\Sigma'\|$ where*

$$\|\Sigma\| = \sum_{A \rightarrow B \in \Sigma} (|A| + |B|).$$

In the same way as in other fields, the use of formulas in a given normal form allows the design of simpler methods with a better performance than those working with arbitrary expressions (e.g. the use of Horn clauses in Logic Programming). In FCA, the normal form usually chosen to improve the methods to get the direct-optimal basis is the unitary implication.

As a first step, Bertet et al. proposed a method to build the direct-optimal basis from an arbitrary IS [3]. Later, they proposed a second method which works with UISs [4]. The main advantage in the use of general IS is the smaller size of the input implication set whereas the use of UIS allows a better performance of the second method.

In the rest of this section, we summarize the methods proposed in [3] (for IS) and in [4] (for UIS) and the main differences between them will be highlighted.

### *3.1 Non-Unitary Direct-Optimal basis*

Given an arbitrary IS $\Sigma$, the way to proceed is the following: first, a direct IS $\Sigma_d$ is built by adding new

© 2015 NSP
Natural Sciences Publishing Cor.

implications to $\Sigma$. Then the direct-optimal $\Sigma_{do}$ is obtained by removing from $\Sigma_d$ those implications that are redundant whose elimination preserves directness. In the first step, we apply exhaustively the following rule:

$$\texttt{[Overlap]} \quad \frac{A \to B, C \to D}{A(C\text{-}B) \to D}, \; B \cap C \neq \emptyset$$

The second step is mainly based on the following rule:

$$\texttt{[Optimization]} \quad \frac{A \to B, C \to D}{A \to B\text{-}AD}, \quad C \subset A$$

Note that optimization rule is close to the above simplification rule. This rule induces an equivalence named Optimization Equivalency:

**[OpEq]**: If $C \subset A$ then

$$\{A \to B, C \to D\} \equiv \{C \to D, A \to B\text{-}AD\}$$

To compute the unique direct-optimal IS equivalent to $\Sigma_d$, we apply exhaustively **[OpEq]**+**[CoEq]**+**[FrEq]**. If the application of **[OpEq]** renders a trivial implication $A \to \emptyset$, it is removed from the output. We introduce now an illustrative example:

*Example 6.*Let $\Sigma$ be the IS considered in Example 5. The first step builds the following direct IS with 31 implications:

$\Sigma_d = \{$ NA $\to$ Gr77
OPEC Gr77 NA LLDC $\to$ MASC ACP
MASC $\to$ Gr77
OPEC Gr77 NA MASC $\to$ LLDC ACP
OPEC $\to$ Gr77 NA
OPEC NA LLDC $\to$ MASC ACP
OPEC $\to$ Gr77
OPEC NA MASC $\to$ LLDC ACP
OPEC MASC $\to$ LLDC ACP
OPEC Gr77 NA LLDC $\to$ Gr77
OPEC MASC $\to$ Gr77
OPEC Gr77 NA MASC $\to$ MASC ACP
OPEC LLDC $\to$ MASC ACP
OPEC NA MASC LLDC $\to$ MASC ACP
OPEC NA LLDC $\to$ Gr77
OPEC NA LLDC $\to$ LLDC ACP
OPEC MASC LLDC $\to$ LLDC ACP
OPEC NA MASC LLDC $\to$ Gr77
OPEC NA MASC $\to$ Gr77
OPEC Gr77 NA MASC $\to$ Gr77
OPEC LLDC $\to$ Gr77
OPEC Gr77 NA MASC LLDC $\to$ LLDC ACP
OPEC MASC LLDC $\to$ MASC ACP
OPEC Gr77 NA MASC LLDC $\to$ MASC ACP
OPEC MASC $\to$ MASC ACP
OPEC NA MASC LLDC $\to$ LLDC ACP
OPEC MASC LLDC $\to$ Gr77
OPEC Gr77 NA LLDC $\to$ LLDC ACP
OPEC LLDC $\to$ LLDC ACP
OPEC Gr77 NA MASC LLDC $\to$ LLDC ACP
OPEC NA MASC $\to$ MASC ACP$\}$

After this, the second step renders the direct-optimal basis with 5 implications:

$\Sigma = \{$ OPEC $\to$ NA Gr77
NA $\to$ Gr77
MASC $\to$ Gr77
LLDC OPEC $\to$ MASC ACP
MASC OPEC $\to$ ACP LLDC$\}$

□

## 3.2 Unit Direct-Optimal basis

We recall here the method proposed in [5] to obtain the direct-optimal basis equivalent to a given Unit Implicational System .

Firstly, the method computes a direct UIS $\Sigma_d$ by exhaustively applying the following rules, which are particularizations of `[Overlap]` and `[Optimization]` to UIS:

$$\texttt{[UnitOverlap]} \quad \frac{A \to b, Cb \to d}{AC \to d}, \text{ where } d \neq b \text{ and } d \notin A$$

$$\texttt{[UnitOptimization]} \quad \frac{C \to b}{A \to b}, \quad C \subset A$$

The second rule above is indeed used to *narrow* the implications, using the following equivalence:

**[NarrEq]**: If $C \subset A$ then $\{A \to b, C \to b\} \equiv \{C \to b\}$

Notice that the use of UIS turns the method into an easier one. We illustrate the method in the following example:

*Example 7.*The UIS equivalent to the IS given in example 5 is:

$\Sigma = \{$ OPEC $\to$ Gr77
OPEC $\to$ NA
NA $\to$ Gr77
MASC $\to$ Gr77
Gr77 NA MASC OPEC $\to$ LLDC
Gr77 NA MASC OPEC $\to$ ACP
Gr77 NA LLDC OPEC $\to$ MASC
Gr77 NA LLDC OPEC $\to$ ACP$\}$

Firstly, from this set with 5 unitary implications, the following direct UIS with 26 implications is generated (using the `[UnitOverlap]` rule):

© 2015 NSP
Natural Sciences Publishing Cor.

$$\begin{aligned}\Sigma_d = \{\ & \text{OPEC} \to \text{Gr77}\\ & \text{OPEC} \to \text{NA}\\ & \text{NA} \to \text{Gr77}\\ & \text{MASC} \to \text{Gr77}\\ & \text{OPEC Gr77 NA MASC} \to \text{LLDC}\\ & \text{OPEC Gr77 NA MASC} \to \text{ACP}\\ & \text{OPEC Gr77 NA LLDC} \to \text{MASC}\\ & \text{OPEC Gr77 NA LLDC} \to \text{ACP}\\ & \text{OPEC NA LLDC} \to \text{ACP}\\ & \text{OPEC NA MASC LLDC} \to \text{ACP}\\ & \text{OPEC Gr77 LLDC} \to \text{ACP}\\ & \text{OPEC NA LLDC} \to \text{MASC}\\ & \text{OPEC Gr77 LLDC} \to \text{MASC}\\ & \text{OPEC NA MASC} \to \text{ACP}\\ & \text{OPEC Gr77 MASC} \to \text{ACP}\\ & \text{OPEC NA MASC} \to \text{LLDC}\\ & \text{OPEC Gr77 MASC} \to \text{LLDC}\\ & \text{OPEC MASC LLDC} \to \text{ACP}\\ & \text{OPEC NA LLDC} \to \text{Gr77}\\ & \text{OPEC MASC} \to \text{ACP}\\ & \text{OPEC MASC} \to \text{LLDC}\\ & \text{OPEC LLDC} \to \text{ACP}\\ & \text{OPEC LLDC} \to \text{MASC}\\ & \text{OPEC LLDC} \to \text{Gr77}\\ & \text{OPEC NA MASC} \to \text{Gr77}\\ & \text{OPEC MASC} \to \text{Gr77}\}\end{aligned}$$

Notice that in this case, the intermediate direct basis is smaller than that presented in Example 6 for non-unit IS.

Now, `[NaEq]` is applied, rendering the unit direct-optimal basis with 7 unitary implications.

$$\begin{aligned}\Sigma_{do} = \{\ & \text{OPEC} \to \text{Gr77}\\ & \text{OPEC} \to \text{NA}\\ & \text{NA} \to \text{Gr77}\\ & \text{MASC} \to \text{Gr77}\\ & \text{OPEC, MASC} \to \text{LLDC}\\ & \text{OPEC, MASC} \to \text{ACP}\\ & \text{OPEC, LLDC} \to \text{MASC}\\ & \text{OPEC, LLDC} \to \text{ACP}\}\end{aligned}$$ □

An analysis of the previous approaches provides a set of interesting conclusions which guide the design of the new method proposed in this paper. Although the use of UIS causes a non trivial growth of the input set of implications with respect to IS (from 5 to 8 in the cardinal and from 19 to 28 in the size, for the case of $\mathbb{K}_0$), the method based on UIS shows a better performance. One reason is that the intermediate direct basis built after the first step is smaller in the UIS method than in the IS one (26 vs 31 in cardinal and 95 vs 144 in size). This is a key point in the efficiency of the method because the size of the implicational set directly influences the performance, in terms of the number of applications of the rules and equivalences. Thus, the total number of applications is 57 in the case of IS and 36 in the UIS method. This significant difference is due to the fact that unitary implications have a smaller *possibility* to fit the conditions imposed in the equivalences, which are based on the set inclusion and intersection operators.

Nevertheless, an interesting direction to improve the efficiency of these methods can be to reduce the cardinal/size of the intermediate direct basis, which influences the cost of the second stage. Thus, our aim is to design a new method which combines the best of these two approaches: to work with IS so that we limit the cardinal and size of the set of implications at any time and to define new rules which reduce the number of applications, avoiding a growth in the first step that have to be narrowed in the second stage with extra computations.

In Table 2 we summarize the performance of the methods presented in [3] (for IS) and in [5] (for UIS) over Example 5. This table indicates the cardinal and the size of the implicational set at each stage of the method and the number of applications of the rules throughout its execution.

## 4 A new method to compute direct optimal basis.

As stated above, working with UIS causes a growth in the size of the implication sets. The design of new methods admitting arbitrary IS would provide a more compact representation of these sets. Up to now, this way has shown to be unsuccessful because of its bigger number of applications of inference rules, which usually produce redundant attributes in both sides of the new implication. These superfluous attributes can be safely removed from the right-hand side avoiding, in this way, extra applications of the inference rules in the future. Let us see an example to illustrate this situation:

*Example 8.*Consider $\mathbb{K}_0$ from Example 5. The second and fifth implications are respectively:

$$\text{MASC} \to \text{Gr77}$$

$$\text{Gr77 NA LLDC OPEC} \to \text{MASC ACP}$$

These implications satisfy the precondition of the `[Overlap]` rule, after its application we obtain:

$$\text{NA LLDC OPEC MASC} \to \text{MASC ACP}$$

Notice that the MASC attribute is redundant and it will cause new applications of the `[Overlap]` rule when the method continues. □

Our goal here is to design a new method that applies the paradigm of *reduction* in order to achieve a method working with (not necessarily unitary) IS which, moreover, reduces the extra coupling of implications. To this end, we propose to work with reduced implications, which do not have redundant attributes in their right-hand sides: Algorithm 1 has four main stages, each one consisting of the transformation of a previous IS into an equivalent one fulfilling some additional property.

© 2015 NSP
Natural Sciences Publishing Cor.

**Table 2:** Comparison of IS and UIS methods.

Algorithm from [3] to IS

| $\mid \Sigma \mid$ | $\mid\mid \Sigma \mid\mid$ | $\mid \Sigma_d \mid$ | $\mid\mid \Sigma_d \mid\mid$ | Num. Rules Applied | $\mid \Sigma_{do} \mid$ | $\mid\mid \Sigma_{do} \mid\mid$ |
|---|---|---|---|---|---|---|
| 5 | 19 | 31 | 144 | 57 | 5 | 15 |

Algorithm from [5] to UIS

| $\mid \Sigma \mid$ | $\mid\mid \Sigma \mid\mid$ | $\mid \Sigma_d \mid$ | $\mid\mid \Sigma_d \mid\mid$ | Num. Rules Applied | $\mid \Sigma_{do} \mid$ | $\mid\mid \Sigma_{do} \mid\mid$ |
|---|---|---|---|---|---|---|
| 8 | 28 | 26 | 95 | 36 | 8 | 20 |

**Algorithm 1: DirectOptimalBasis**

**input** : An implicational system $\Sigma$ on $S$
**output**: The direct-optimal IS $\Sigma_{do}$ on $S$

1 **begin**
2 /* Stage 1: Generation of $\Sigma_r$ by reduction of $\Sigma$*/
3 $\Sigma_r = \emptyset$
4 **foreach** $A \rightarrow_\Sigma B$ **do**
5 **if** $B \not\subseteq A$ **then** add $A \rightarrow B$-$A$ to $\Sigma_r$;
6 /* Stage 2: Generation of $\Sigma_{sr}$ by simplification of $\Sigma_r$*/
7 $\Sigma_{sr} = \Sigma_r$
8 **repeat**
9 **foreach** $A \rightarrow B \in \Sigma_{sr}$ **do**
10 **foreach** $C \rightarrow D \in \Sigma_{sr}$ **do**
11 **if** $A \subseteq C$ **then**
12 **if** $C \subseteq A \cup B$ **then**
13 replace $A \rightarrow B$ and $C \rightarrow D$ by $A \rightarrow BD$ in $\Sigma_{sr}$;
14 **else if** $D \subseteq B$ **then**
15 remove $C \rightarrow D$ from $\Sigma_{sr}$;
16 **else** replace $C \rightarrow D$ by $C$-$B \rightarrow D$-$B$ in $\Sigma_{sr}$;
17 **until** $\Sigma_{sr}$ *becomes a fix point*;
18 /* Stage 3: Generation of $\Sigma_{dsr}$ by completion of $\Sigma_{sr}$*/
19 $\Sigma_{dsr} = \Sigma_{sr}$
20 **foreach** $A \rightarrow B \in \Sigma_{dsr}$ *and* $C \rightarrow D \in \Sigma_{dsr}$ **do**
21 **if** $B \cap C \neq \emptyset \neq D \setminus (A \cup B)$ **then**
22 add $AC$-$B \rightarrow D$-$(AB)$ to $\Sigma_{dsr}$
23 */Stage 4: Generation of $\Sigma_{do}$ by optimization of $\Sigma_{dsr}$*/
24 $\Sigma_{do} = \emptyset$
25 **foreach** $A \rightarrow B \in \Sigma_{dsr}$ **do**
26 **foreach** $C \rightarrow D \in \Sigma_{dsr}$ **do**
27 **if** $C = A$ **then** $B = B \cup D$;
28 **if** $C \subsetneq A$ **then** $B = B \setminus D$;
29 **if** $B \neq \emptyset$ **then** add $A \rightarrow B$ to $\Sigma_{do}$;
30 **return** $\Sigma_{do}$

**Definition 7(Reduced IS).** *An IS $\Sigma$ is* reduced *iff it does not have any trivial implication and for all $A \rightarrow B \in \Sigma$ we have that $A \cap B = \emptyset$.*

Notice that this condition (i.e. being reduced) is a natural one in the original specification of the problem, but the inspiration of the method proposed in this paper is to maintain intermediate ISs reduced at any time by means of the application of inference rules which always produce reduced implications.

The first stage of Algorithm 1 (lines 2 to 5) transforms an arbitrary IS into an equivalent and reduced one, just by applying **[FrEq]**. The following definition introduces the target IS for the second stage (see lines 6–17).

**Definition 8(Simplified IS).** *An IS $\Sigma_r$ is* ***simplified*** *if the following conditions hold: for all $A,B,C,D \subseteq M$,*

*1.$A \rightarrow B, A \rightarrow C \in \Sigma$ implies $B = C$.*
*2.$A \rightarrow B, C \rightarrow D \in \Sigma$ and $A \subsetneq C$ imply $C \cap B = \emptyset = D \cap B$.*

Theorem 1 provides three equivalencies which allow to remove redundant information when reading from left to right. The simplified implicational system is obtained by applying these equivalences to remove redundant information. More specifically, any IS can be transformed into a simplified equivalent one by systematically applying the simplification and composition equivalences based on $\mathbf{SL}_{\mathrm{FD}}$ (**[SiEq]**+**[CoEq]**). If the application of **[SiEq]** renders an implication $A \rightarrow \emptyset$, it is removed from the output. Notice that the fixpoint referenced in the second stage of Algorithm 1 is reached because the IS are finite and in each loop *at least* one attribute is removed; furthermore, as in Algorithm 1 this transformation is applied to reduced IS, it always renders a simplified reduced IS.

**Theorem 2.***Let $\Sigma_{sr}$ be the IS obtained by using lines 6 to 17 of Algorithm 1 on a reduced IS $\Sigma_r$. Then $\Sigma_{sr}$ is a simplified reduced IS equivalent to $\Sigma_r$.*

*Proof.*The loop in lines 8–17 always finishes because the size of $\Sigma_r$ is finite and, in each step of the bucle, this size decreases. Theorem 1 ensures that $\Sigma_{sr} \equiv \Sigma_r$. Moreover, $\Sigma_{sr}$ is simplified (Definition 8) because it is a fixpoint for the loop. Finally, $\Sigma_{sr}$ is reduced because these equivalence rules preserve this feature. □

In the third stage of Algorithm 1, $\Sigma_{sr}$ is transformed into an equivalent direct reduced IS by exhaustively

© 2015 NSP
Natural Sciences Publishing Cor.

applying the following rule [2], called *strong Simplification rule*,

$$\texttt{[sSimp]} \quad \frac{A \to B, C \to D}{AC\text{-}B \to D\text{-}(AB)},\ B \cap C \neq \emptyset \neq D \setminus (A \cup B)$$

summarizing, lines 19–22 compute the smallest IS $\Sigma_{dsr}$ fulfilling:

1. $\Sigma_{sr} \subseteq \Sigma_{dsr}$ and
2. for all pair of implications $A \to B,\ C \to D \in \Sigma_{dsr}$ where $B \cap C \neq \emptyset \neq D \setminus (A \cup B)$ we have that $AC\text{-}B \to D\text{-}(AB) \in \Sigma_{dsr}$.

In order to prove that $\Sigma_{dsr}$ is a direct reduced IS which is equivalent to $\Sigma_{sr}$, firstly the following lemma ensures the soundness of the strong Simplification rule.

**Lemma 1.***The* strong Simplification rule,

$$\texttt{[sSimp]} \quad \frac{A \to B, C \to D}{AC\text{-}B \to D\text{-}(AB)},\ B \cap C \neq \emptyset \neq D \setminus (A \cup B)$$

*can be derived from Armstrong's axioms.*

*Proof.*Assume $B \cap C \neq \emptyset \neq D \setminus (A \cup B)$. The following sequence proves the soundness of [sSimp]:

1. $A \to B$ ..... by hypothesis.
2. $B \to B \cap C$ ..... by [Ax]
3. $A \to B \cap C$ ..... by 1., 2. and [Trans]
4. $C\text{-}B \to C\text{-}B$ ..... by [Ax]
5. $A(C\text{-}B) \to (B \cap C)(C\text{-}B)$ .. by 3., 4. and [Comp] $= A(C\text{-}B) \to C$
6. $C \to D$ ..... by hypothesis
7. $A(C\text{-}B) \to D$ ..... by 5., 6. and [Trans]
8. $D \to D\text{-}(AB)$ ..... by [Ax]
9. $A(C\text{-}B) \to D\text{-}(AB)$ .. by 7., 8. and [Trans] □

The following theorem ensures that the IS $\Sigma_{dsr}$ computed at stage 3 (after line 22) has the desired properties.

**Theorem 3.***Given a reduced IS, $\Sigma_{sr}$, the implicational system $\Sigma_{dsr}$ computed by lines 19–22 is a direct reduced IS which is equivalent to $\Sigma_{sr}$.*

*Proof.*Because of Lemma 1, we already have that $\Sigma_{dsr} \equiv \Sigma_{sr}$. Moreover, $\Sigma_{dsr}$ is reduced since [sSimp] preserves this property. In order to prove the directness, we will prove that, for all attribute set $X$, if $y \in X^+ \setminus X$ then there exists $X' \to Y' \in \Sigma_{dsr}$ such that $X' \subseteq X$ and $y \in Y'$.

Let $\Sigma_f$ be the set of all implications that can be derived from $\Sigma_{sr}$ by using the Armstrong's axioms (the so-called full implicational system) and let $\Sigma_i$ with $0 \leq i \leq p$ such that

– $\Sigma_0 = \Sigma_{dsr}$,
– for each $1 \leq i \leq p$, $\Sigma_i = \Sigma_{i-1} \cup \{X_i \to Y_i\}$ where $X_i \to Y_i$ is directly obtained from $\Sigma_{i-1}$ by [Ax], [Augm] or [Trans]
– and $\Sigma_p = \Sigma_f$.

Note that $y \in X^+ \setminus X$ if and only if there exists $X \to Y \in \Sigma_p$ with $y \in Y \setminus X$.

We will prove inductively that, for all $0 \leq i \leq p$, if $X \to Y \in \Sigma_i$ with $y \in Y \setminus X$ then there exists $X' \to Y' \in \Sigma_{dsr}$ such that $X' \subseteq X$ and $y \in Y'$. The *base case* is straightforward because $\Sigma_0 = \Sigma_{dsr}$.

*Inductive step:* For $i \geq 1$, assume the property is true for $\Sigma_{i-1}$, i.e. for all $X \to Y \in \Sigma_{i-1}$, for all $y \in Y \setminus X$, there exists $X' \to Y' \in \Sigma_{dr}$ such that $X' \subseteq X$ and $y \in Y'$. Let us prove that the property is also true for $X_i \to Y_i$:

– *Case* [Ax]: If $X_i \to Y_i$ is obtained by [Ax] then $Y_i \subseteq X_i$, i.e. $Y_i \setminus X_i = \emptyset$, and the property is trivially satisfied.
– *Case* [Augm]: in this case, there exists $A \to B \in \Sigma_{i-1}$ such that $X_i = A \cup C$ and $Y_i = B \cup C$. If $y \in Y_i \setminus X_i$ then $y \in B \setminus A$ and, since $A \to B \in \Sigma_{i-1}$, by induction hypothesis, there exists $A' \to B' \in \Sigma_{dsr}$ such that $A' \subseteq A \subseteq A \cup C = X_i$ and $y \in B'$.
– *Case* [Trans]: There exist $A \to B, B \to C \in \Sigma_{i-1}$ such that $X_i = A, Y_i = C$ and $y \in C \setminus A$. Let us consider the two sub-cases $y \in B$ and $y \notin B$.
  – $y \in B$ implies $y \in B \setminus A$, and since $A \to B \in \Sigma_{i-1}$: by induction hypothesis there exists $A' \to B' \in \Sigma_{dsr}$ such that $A' \subseteq A = X_i$ and $y \in B'$.
  – $y \notin B$ implies $y \in C \setminus B$, and since $B \to C \in \Sigma_{i-1}$: by induction hypothesis there exists $B' \to C' \in \Sigma_{dsr}$ such that $B' \subseteq B$ and $y \in C'$. If $B' \subseteq A = X_i$ the property trivially holds. Otherwise, if $B' \not\subseteq A$, let us write $B' \setminus A = \{y_k \mid 1 \leq k \leq q\}$. Since $B' \subseteq B$ : $y_k \in B \setminus A$, and since $A \to B \in \Sigma_{i-1}$: by induction hypothesis there exist $q$ implications $A_k \to B_k \in \Sigma_{dsr}$ such that $A_k \subseteq A$ and $y_k \in B_k$. Therefore:

$$B' \setminus A \subseteq \bigcup_{1 \leq k \leq q} B_k \text{ and } B' \subseteq A \cup \bigcup_{1 \leq k \leq q} B_k$$

  The [sSimp] rule is now used to build an implication whose premise is included into $A$ and $y$ belongs to its conclusion. Let us consider the two sub-cases $y \in \bigcup_{1 \leq k \leq q} B_k$ and $y \notin \bigcup_{1 \leq k \leq q} B_k$.

  • If $y \in \bigcup_{1 \leq k \leq q} B_k$ then there exists $k \in [1, q]$ such that $y \in B_k$, so $A_k \to B_k \in \Sigma_{dsr}$ satisfies the property.
  • If $y \notin \bigcup_{1 \leq k \leq q} B_k$, then the implication we are searching is the last element of a sequence of $q$ implications $A'_k \to C'_k \in \Sigma_{dsr}$ obtained by iteratively applying [sSimp] to implications $A_k \to B_k \in \Sigma_{dsr}$.
  · *Base case:* we define $A'_1 \to C'_1 \in \Sigma_{dsr}$ as the result of [sSimp] applied to $A_1 \to B_1, B' \to C' \in \Sigma_d$ (note that $y_1 \subseteq B_1 \cap B'$ so $B_1 \cap B' \neq \emptyset$ and [sSimp] can be applied), so $A'_1 = A_1 \cup B' \setminus B_1$ and $C'_1 = C' \setminus (A_1 \cup B_1)$ (note that $y \in C'_1 : y \in C', y \notin A_1 \subseteq A, y \notin B_1$).

[2] Notice that it is not necessary to put brackets to define the order of the operations in *AC-B* because the corresponding IS is reduced: if $AC \cap B \neq \emptyset$ only $C \cap B \neq \emptyset$ holds, so $(AC)$-$B$ and $A(C\text{-}B)$ are equal.

© 2015 NSP
Natural Sciences Publishing Cor.

·*Inductive case:* for $i < k \leq q$, we define $A'_k \to C'_k \in \Sigma_{dr}$ as:

1. the result of `[sSimp]` applied to $A_k \to B_k \in \Sigma_d$ and $A'_{k-1} \to C'_{k-1} \in \Sigma_d$ if $B_k \cap A'_{k-1} \neq \emptyset$, so $A'_k = A_k \cup A'_{k-1} \setminus B_k$ and $C'_k = C'_{k-1} \setminus (B_k \cup A_k)$.
2. $A'_{k-1} \to C'_{k-1} \in \Sigma_d$ otherwise, so $A'_k = A'_{k-1}$ and $C'_k = C'_{k-1}$.

To prove that $A'_q \subseteq A$, we will prove, again by induction on $k \in [1,q]$, that

$$A'_{k-1} \subseteq A \cup \bigcup_{k<j\leq q} B_j$$

–*Base case:* For $k = 1$, we have $A'_1 = A_1 \cup (B' \setminus B_1)$ so $A'_1 \subseteq A \cup \bigcup_{1<j\leq qB_j}$ directly follows from $B' \subseteq A \cup \bigcup_{1\leq k\leq qB_k}$ and $A_1 \subseteq A$.

–*Inductive case:* For $k > 1$, the induction hypothesis is

$$A'_{k-1} \subseteq A \cup \bigcup_{k-1<j\leq q} B_j$$

moreover the computation of $A'_k$ depends on the emptiness of $B_k \cap A'_{k-1}$.

–If $B_k \cap A'_{k-1} = \emptyset$, then

$$A'_{k-1} \subseteq (A \cup \bigcup_{k-1<j\leq q} B_j \setminus B_k$$

moreover $A'_k = A'_{k-1}$. So we directly obtain $A'_k \subseteq A \cup \bigcup_{k<j\leq q} B_j$

–If $B_k \cap A'_{k-1} \neq \emptyset$, then $A'_{k-1} \setminus B_k \subseteq A \cup \bigcup_{k<j\leqq} B_j$. Moreover $A'_k = A_k \cup (A'_{k-1} \setminus B_k)$ and since $A_k \subseteq A$, qe also obtain $A'_k \subseteq A \cup \bigcup_{k<j\leq q} B_j$.

We finally obtain

$$A'_q \subseteq A \cup \bigcup_{q<j\leq q} B_j \subseteq A$$

and $A'_q \to C' \in \Sigma_{dr}$ satisfies the property (since $A = X_i$ and $y \in C'$). Thus the property is proved. □

This section concludes with results ensuring that the output of Algorithm 1 is an equivalent direct-optimal implicational system.

**Lemma 2.***Let $\Sigma_{dsr}$ be a direct-reduced IS and $\Sigma_{do}$ be the IS obtained from $\Sigma_{dsr}$ by using lines 24–29 in Algorithm 1. Then $\Sigma_{do}$ is a direct-reduced simplified IS equivalent to $\Sigma_{dsr}$.*

*Proof.*It is easy to check that the transformation given by lines 24–29 preserves equivalence, directness and being reduced. Since fragmentation and composition equivalences have been applied, then $\Sigma_{do}$ is also reduced, and $A \to B, A \to C \in \Sigma_{do}$ implies $B = C$ (case 1 of the definition of being simplified).

To prove that $A \to B, C \to D, A \subset C$ imply $C \cap B = \emptyset =$ D $\cap B$ (case 2 of the definition of being simplified), let us observe that:

1. $C \cap B = \emptyset$: trivial from $C \subset A$ and $A \cap B = \emptyset$ by the reduction property.
2. $D \cap B = \emptyset$: trivial from the optimization equivalence `[OpEq]`. If $D \cap B \neq \emptyset$, then $A \to B$ is replaced by $A \to B - AD$, thus a contradiction with $A \to B \in \Sigma_{do}$.
3. If $A \to B, C \to D \in \Sigma$ with $A \subset C$ and $C \cap B \neq \emptyset \neq D \cap B$ then $\Sigma \setminus \{C \to D\} \cup \{C\text{-}B \to D\text{-}B\}$ is also a direct-reduced IS equivalent to $\Sigma$ of smaller size. □

The following theorem provided in [3] allows us to conclude this section with Theorem 5 which ensures that Algorithm 1 returns the only direct-optimal base equivalent to the original one.

**Theorem 4( [3]).** *A direct IS $\Sigma$ is direct-optimal iff:*

–*(extensiveness): for all $A \to B \in \Sigma$, $A \cap B = \emptyset$*
–*(isotony): for all $A \to B, C \to D \in \Sigma$, if $C \subset A$ then $B \cap D = \emptyset$.*
–*(premise): for all $A \to B, A \to B' \in \Sigma$, $B = B'$.*
–*(non-empty conclusion): for all $A \in B \in \Sigma$, $B \neq \emptyset$.*

**Theorem 5.***Let $\Sigma$ be an implicational system on $M$ and let $\Sigma_{do}$ be the IS output by Algorithm 1. Then, $\Sigma_{do}$ is the direct-optimal implicational system equivalent to $\Sigma$.*

*Proof.*From Theorems 2 and 3 and Lemma 2, $\Sigma_{do}$ is a direct-reduced simplified IS equivalent to $\Sigma$. Since $\Sigma_{do}$ is reduced, extensiveness and non-emptiness of the conclusion hold. Moreover, since it is simplified, isotony and premise also hold. Finally, Theorem 4 ensures that $\Sigma_{do}$ is the direct-optimal base equivalent to $\Sigma$. □

## 5 The direct-optimal method in action

In this section, the execution of the method is shown on an illustrative example and, then, we present the initial results of our experimental evaluation in order to obtain the conclusions on its performance in practice.

*Example 9.*In this example, the execution of the method is carried out step by step rendering a direct-optimal basis for the IS in Example 5:

$$\begin{aligned}\Sigma = \{\ & \text{OPEC} \to \text{Gr77 NA},\\ & \text{MASC} \to \text{Gr77}\\ & \text{NA} \to \text{Gr77}\\ & \text{Gr77 NA MASC OPEC} \to \text{LLDC ACP}\\ & \text{Gr77 NA LLDC OPEC} \to \text{MASC ACP}\}\end{aligned}$$

In the first and second stages, Algorithm 1 calculates the following equivalent simplified-reduced IS $\Sigma_{sr}$:

$$\begin{aligned}\Sigma_{sr} = \{\ & \text{OPEC} \to \text{NA}\\ & \text{NA} \to \text{Gr77}\\ & \text{MASC} \to \text{Gr77}\\ & \text{LLDC OPEC} \to \text{MASC}\\ & \text{MASC OPEC} \to \text{ACP LLDC}\}\end{aligned}$$

© 2015 NSP
Natural Sciences Publishing Cor.

The third stage renders the following direct-simplified-reduced IS $\Sigma_{dsr}$ equivalent to $\Sigma_{sr}$. Note that this set of implications is smaller (in size) than those built with the previous methods. The cardinal of the direct IS is 8 whereas in the previous methods we obtained a cardinal of 26 and 31 for UIS and IS respectively (see Table 2).

$$\begin{aligned}\Sigma_{dsr} = \{ & \text{OPEC} \to \text{NA} \\ & \text{NA} \to \text{Gr77} \\ & \text{MASC} \to \text{Gr77} \\ & \text{LLDC OPEC} \to \text{MASC} \\ & \text{MASC OPEC} \to \text{ACP LLDC} \\ & \text{OPEC} \to \text{Gr77} \\ & \text{LLDC OPEC} \to \text{Gr77} \\ & \text{LLDC OPEC} \to \text{ACP}\}\end{aligned}$$

Although it is not strictly required by the method, by applying `[sSimp]` rule, we achieve an extra reduction in the size of the IS, $||\ \Sigma_{dsr}\ || = 21$. Notice that it is smaller than the size of the direct UIS and IS obtained with previous methods, which were 95 and 144 respectively.

In the last stage of our method the direct-optimal basis $\Sigma_{do}$ is obtained from $\Sigma_{dsr}$:

$$\begin{aligned}\Sigma_{do} = \{ & \text{OPEC} \to \text{NA Gr77} \\ & \text{NA} \to \text{Gr77} \\ & \text{MASC} \to \text{Gr77} \\ & \text{LLDC OPEC} \to \text{MASC ACP} \\ & \text{MASC OPEC} \to \text{ACP LLDC}\}\end{aligned}$$

Regarding the number of rules applied, it is worth to remark that the new method does not need to apply reduction at the end to *clean* the basis, because all the stages preserve reduction and the direct-optimal basis is already reduced. In the second stage, it applies four times the **[SiEq]** equivalence to get $\Sigma_{sr}$ and `[sSimp]` rule just once in the third step. In this example, it is not necessary to apply the last loop because in the previous one we already obtained the direct-optimal basis $\Sigma_{do}$. The total number of rules which have been applied is 5 whereas in the previous methods 57 and 36 rules were necessary for IS and UIS, respectively. □

In summary, the new method improves all the previously published ones and, moreover, the key point is the following: it begins by reducing the implications and ensures that reduction is preserved at any time dealing with smaller IS than any other method based on UIS. It also narrows the input by using **[SiEq]**, an equivalence based on $\mathbf{SL}_{\text{FD}}$, and by adding less implications to compute the intermediate direct IS by using `[sSimp]`.

In the rest of this section, we focus on the design and development of an empirical study to get some practical conclusions on the performance of the algorithm from a more exhaustive application. The methods of Bertet et al. [3, 4] and our newly proposed method have been implemented in SWI-Prolog.[3] The input are sets of implications randomly generated, increasing their size until the limits of the algorithms are reached.

Table 3 summarizes the results of a number of operations that makes the algorithm to compute the direct-optimal basis in these experiments.

For each method, there are three columns showing the following information: the first one represent the number of logical inferences per second (lips) often used to describe the performance of a logical reasoning system; the second one is the execution time in seconds; and the third column stores the number of couples of implications in which a rule is applied.

The name of each example encloses the number of implications and a serial number for unique identification ; e.g. E10_9 correspond to the experiment #9, having 10 implications. Table 3 shows that experiments with 15 implications saturate machine resources for the previous methods. In all three parameters, the method proposed in this paper obtains much better results.

## 6 Conclusions

In this work we have presented a new method to compute the direct-optimal basis in a way more efficient than previous methods published in the literature. The efficiency of the method is illustrated through a first experiment in which we compare the three methods.

Currently, we are conducting a more exhaustive comparison with a random generator of implications and formal contexts. Moreover, we will implement our method in Java using the lattice library provided by Bertet in the GitHub platform, in order to further compare the three methods.

We are working in new methods to compute the direct-optimal basis by trying to take extra advantages of the huge number of algebraic-logical results already known about implications. As future work, we are planning to use the $\mathbf{SL}_{\text{FD}}$ paradigm for the study of automated methods to compute other types of basis, for instance, Duquenne-Guigues, ordered-direct, etc.

## Acknowledgment

Partially supported by Grants TIN2011-28084 and TIN12-39353-C04-01of the Science and Innovation Ministry of Spain, co-funded by the European Regional Development Fund (ERDF).

## References

[1] W. W. Armstrong, Dependency structures of data base relationships, Proc. IFIP Congress. North Holland, Amsterdam: 580–583 (1974).

[3] Available at http://www.lcc.uma.es/~enciso/do2Simp.zip

© 2015 NSP
Natural Sciences Publishing Cor.

**Table 3:** Random Experiment

| lips/Time/Com | BertetNebutDO | | | BertetUnitDO | | | Simp | | |
|---|---|---|---|---|---|---|---|---|---|
| E10_1 | 575,956,176 | 535.668 | 33,242 | 91,936,721 | 36.676 | 2,778 | **7,120** | **0.001** | **15** |
| E10_2 | 150,267,712 | 100.864 | 27,568 | 151,138,149 | 55.520 | 2,781 | **13,779** | **0.002** | **47** |
| E10_3 | 104,840,358 | 63.646 | 22,929 | 22,179,197 | 6.199 | 1,471 | **17,962** | **0.003** | **54** |
| E10_4 | 1,146,989,926 | 1,741.837 | 62,901 | 528,138,643 | 318.767 | 5,210 | **24,335** | **0.003** | **62** |
| E10_5 | 479,587,856 | 428.081 | 32,658 | 87,930,508 | 30.263 | 2,348 | **55,305** | **0.009** | **166** |
| E10_6 | 478,876,258 | 412.915 | 31,280 | 252,840,734 | 102.644 | 4,022 | **15,064** | **0.003** | **51** |
| E10_7 | 2,202,209,028 | 4,458.851 | 74,664 | 354,897,316 | 224.053 | 5,009 | **120,748** | **0.017** | **358** |
| E10_8 | 4,574,831,622 | 11,606.545 | 90,602 | 1,741,975,406 | 1,113.769 | 7,925 | **144,109** | **0.020** | **442** |
| E10_9 | 735,501,082 | 755.756 | 38,970 | 404,815,847 | 167.039 | 4,100 | **51,846** | **0.010** | **118** |
| E10_10 | 305,690,377 | 281.431 | 34,805 | 90,314,741 | 30.703 | 2,538 | **64,858** | **0.011** | **194** |
| E15_1 | | | | | | | **59,293,068** | **7.920** | **6,484** |
| E15_2 | | | | | | | **504,968** | **0.067** | **1,068** |
| E15_3 | | | | | | | **1,563,330** | **0.194** | **1,965** |
| E15_4 | | | | | | | **15,839,947** | **2.191** | **2,838** |


[2] R. Belohlávek, V. Vychodil, Functional Dependencies of Data Tables Over Domains with Similarity Relations, Proc. of the 2nd Indian International Conference on Artificial Intelligence (2005).

[3] K. Bertet, M. Nebut, Efficient algorithms on the Moore family associated to an implicational system, DMTCS, **6**, 315–338 (2004).

[4] K. Bertet, B. Monjardet, The multiple facets of the canonical direct unit implicational basis, Theor. Comput. Sci., **411**, 2155–2166 (2010).

[5] K. Bertet, Some Algorithmical Aspects Using the Canonical Direct Implicationnal Basis, CLA 2006, 101–114 (2006).

[6] D. Maier, The theory of Relational Databases, Computer Science Press (1983).

[7] P. Cordero, A. Mora, M. Enciso, I. Pérez de Guzmán, SLFD Logic: Elimination of Data Redundancy in Knowledge Representation, Lecture Notes in Computer Science, **2527**, 141–150 (2002).

[8] P. Cordero, M. Enciso, A. Mora, M. Ojeda-Aciego, Computing Minimal Generators from Implications: a Logic-guided Approach, CLA 2012, 187–198 (2012).

[9] P. Cordero, M. Enciso, A. Mora, M. Ojeda-Aciego, Computing Left-Minimal Direct Basis of Implications, CLA 2013, 293–298 (2012).

[10] B. Ganter, Two basic algorithms in concept analysis, Technische Hochschule, Darmstadt (1984).

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

[12] A. Mora, M. Enciso, P. Cordero, I. Fortes, Closure via functional dependence simplification, International Journal of Computer Mathematics, **89**, 510–526 (2012).

[13] A. Mora, M. Enciso, P. Cordero, I. Pérez de Guzmán, An Efficient Preprocessing Transformation for Functional Dependencies Sets Based on the Substitution Paradigm, Lecture Notes in Computer Science, **3040**, 136–146 (2004).

[14] K. Adaricheva, J.B. Nation, R. Rand, Ordered Direct Implicational Basis of a finite closure system, Discrete Applied Mathematics, **161**, 707–723 (2013).

[15] V. Duquenne, C. Chabert, A. Cherfouh, A.L. Doyen, J.M. Delabar, D. Pickering , Structuration of Phenotypes and Genotypes through Galois Lattices and Implications, Applied Artificial Intelligence, **17**, 243–256 (2003).

[16] K.S. Qu, Y.H. Zhai, Generating complete set of implications for formal contexts, Knowledge Based Systems, **21**, 429–433 (2008).

[17] R. Belohlávek, V. Vychodil, Formal Concept Analysis with background knowledge: attributes priorities, IEEE Transactions on Systems, Man, and Cybernetics, Part C **39**, 399–409 (2009).

[18] E. Rodríguez, P. Cordero, M. Enciso. A. Mora, A logic-based approach to compute a direct basis from implications, Proceedings of the 14th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2014, 331–339 (2014).

[19] S. Rudolph, Some notes on Pseudo-closed sets, ICFCA 2007, 151–165 (2007).

[20] S. Rudolph, Some notes on managing closure operators, ICFCA 2012, 278–291 (2007).


**Estrella Rodríguez-Lorenzo** received the degree in Mathematics in University of Sevilla. Her research interests are in the areas of applied mathematics, logic, automated reasoning, Lattice theory and generalizations, Formal methods in databases and in Formal Concept Analysis. She is doing her PhD Thesis in the algebraic and logical foundations of Formal Concept Analysis and she has published research articles in reputed international applied mathematical and engineering sciences conferences.

© 2015 NSP
Natural Sciences Publishing Cor.

**Karell Bertet** is Associate Professor at University of La Rochelle and received the PhD degree in Computer Science at University of La Rochelle. Her research interests are in the areas of applied mathematica, automated reasoning, Lattice structures and algorithmic aspects, Image data analysis, Formal methods in databases, Formal Concept Analysis. She has published research articles in reputed international journals of applied mathematical and engineering sciences. She is referee of mathematical and computer science journals. She is member of the Francophone community trellises, and program committees of conferences (CLA, ICFCA, CIFED).

**Pablo Cordero** is Associate Professor of the Applied Mathematic Department at the University of Málaga (Spain). He received the MSc in Mathematics in 1992 from the Universidad Complutense de Madrid, and the PhD in Computer Science in 1999 from the University of Málaga. His research is devoted to Mathematics applied to Computer Sciences. Specifically, his research area is the information and knowledge treatment via logic and its algebraic foundations: Logic and Automated Reasoning, Fuzzy Logic, Formal Methods in Databases, Formal Concept Analysis, Hyperstructures and Multilattice Theory, Galois Connections and Applications, etc. He is coauthor of more than forty research articles in a variety of scientific journals (e.g. Fuzzy Set Syst., Inf. Sci., Appl. Math. Comput., Ann. Math. Artif. Intell., Logic Jnl IGPL , etc.) and seventy contributions to conferences and workshops (e.g. IJCAI, IPMU, CLA, ICFCA, ICSOFT, IWANN, etc.). He is member of the editorial board of the International Journal of Algebraic Hyperstructures and Applications.

**Manuel Enciso** is Associate Professor in the Languages and Computer Sciences Department at University of Málaga. In this university received the PhD degree in Computer Science. His research is focussed in Intelligent Information Systems, logic, fuzzy logic, methods of automated reasoning, Formal Concept Analysis, and Software Meta-models. He is a member of the research group SICUMA and has published his research articles in international journals and conferences.

**Angel Mora** is Associate Professor at University of Málaga and received the PhD degree in Computer Science at University of Málaga. His research interests are in the areas of applied mathematica, logic, modal logic, fuzzy logic, automated reasoning, Lattice theory and generalizations, Formal methods in databases, Formal Concept Analysis. He has published research articles in reputed international journals of applied mathematical and engineering sciences. He is referee of mathematical and computer science journals.

**Manuel Ojeda-Aciego** received the MSc in Mathematics in 1990, and the PhD in Computer Science in 1996, both from the University of Málaga, Spain. He is currently full professor in the Department of Applied Mathematics, University of Málaga, and has authored or coauthored more than 120 papers in scientific journals and proceedings of international conferences. He has co-edited the book Foundations of Reasoning under Uncertainty (Springer-Verlag, 2010), as well as several special issues in scientific journals on mathematical and logical foundations of non-classical reasoning. His current research interests include fuzzy answer set semantics, residuated and multi-adjoint logic programming, fuzzy formal concept analysis, and algebraic structures for computer science. He is the president of the Computer Science Committee of the Royal Spanish Mathematical Society, Area Editor of the Intl J on Uncertainty and Fuzziness in Knowledge-based Systems, member of the Editorial Board of the IEEE Tr on Fuzzy Systems, member of the Steering Committee of the Intl Conf on Concept Lattices and their Applications (CLA) and the Intl Conf on Information Processing and Management of Uncertainty in knowledge-based systems (IPMU), member of EUSFLAT, and senior member of the IEEE.

© 2015 NSP
Natural Sciences Publishing Cor.