The Reverse Doubling Construction

Jean-François Viaud[1], Karell Bertet[1], Christophe Demko[1] and Rokia Missaoui[2]
[1]*Laboratory L3i, University of La Rochelle, La Rochelle, France*
[2]*University of Québec in Outaouais, Gatineau, Canada*

Keywords: Concept Lattice, Congruence Relation, Factor Lattice, Arrow Relation, Arrow Closed Subcontext, Compatible Subcontext, Doubling Convex.

Abstract: It is well known inside the Formal Concept Analysis (FCA) community that a concept lattice could have an exponential size in the data. Hence, the size of concept lattices is a critical issue in the presence of large real-life data sets. In this paper, we propose to investigate factor lattices as a tool to get meaningful parts of the whole lattice. These factor lattices have been widely studied from the early theory of lattices to more recent work in the FCA field. This paper contains two parts. The first one gives background about lattice theory and formal concept analysis, and mainly compatible sub-contexts, arrow-closed sub-contexts and congruence relations. The second part presents a new decomposition called "reverse doubling construction" that exploits the above three notions used for the doubling convex construction investigated by Day. Theoretical results and their proofs are given as well as an illustrative example.

1 INTRODUCTION

During the last decade, the computation capabilities have promoted Formal Concept Analysis (FCA) with new methods based on concept lattices. Though they are exponential in space/time in worst case, concept lattices of a reasonable size enable an intuitive representation of data expressed by a formal context that links objects to attributes through a binary relation. Methods based on concept lattices have been developed in various domains such as knowledge discovery and management, databases or information retrieval where some relevant concepts, *i.e.* possible correspondences between objects and attributes are considered either as classifiers, clusters or representative object/attribute subsets.

With the increasing size of data, a set of methods have been proposed in order to either generate a subset (rather than the whole set) of concepts and their neighborhood in an on-line and interactive way (Ferré, 2014; Visani et al., 2011) or better display lattices using nested line diagrams (Ganter and Wille, 1999). Such approaches become inefficient when contexts are huge. However, the main idea of lattice/context decomposition into smaller ones is still relevant when the classification property of the initial lattice is maintained. Many lattice decompositions have been defined and studied, either from an algebraic point of view (Demel, 1982; Mihók and Semanis̃in, 2008) or from an FCA point of view (Ganter and Wille, 1999; Funk et al., 1995). We can cite the Unique Factorisation Theorem (Mihók and Semanis̃in, 2008), the matrix decomposition (Belohlavek and Vychodil, 2010), the Atlas decomposition (Ganter and Wille, 1999), the subtensorial decomposition (Ganter and Wille, 1999), the subdirect decomposition (Demel, 1982; Freese, 2008; Freese, 1997; Freese, 1999; Wille, 1969; Wille, 1976; Wille, 1983; Wille, 1987; Funk et al., 1995), or the doubling convex construction. The doubling convex construction has also been widely studied (Day, 1994; Nation, 1995; Geyer, 1994; Bertet and Caspard, 2002), mainly from a theoretical point of view in order to characterize lattices that can be obtained by such decomposition.

In this paper, we investigate a new method named reverse doubling construction to reduce the size of data. In other words, we propose a new method to construct a smaller lattice from a given one. It is based on the previous work of Day about the doubling convex construction (Day, 1977; Day, 1994). Such a method has then be generalized (Geyer, 1994) and further widely studied (Day et al., 1989; Nation, 1995; Bertet and Caspard, 2002). Intuitively, this construction consists in doubling into a lattice L a convex subset C of nodes of L. In this paper, we propose a "reverse doubling construction" which aims at removing from a lattice L a doubled convex set until no du-

Viaud, J., Bertet, K., Demko, C. and Missaoui, R..
The Reverse Doubling Construction.
In *Proceedings of the 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2015) - Volume 1: KDIR*, pages 350-357
ISBN: 978-989-758-158-8

plicated convex set exists. However, there may be no convex to remove and hence no changes in the initial lattice structure. When the reverse doubling construction successively applied to an assumed *finite lattice L* and a convex set C, and following the Day's doubling construction, the initial lattice L is recovered without any loss of information.

Studies about the doubling convex construction can be organized according to the following chronological sequence of events:

- The first one corresponds to the original work of Day (Day, 1977; Day, 1994; Day et al., 1989; Nation, 1995), who introduced the construction. At the very beginning, only intervals were doubled.
- Further generalizations were developed that lead to the general doubling convex method (Geyer, 1994).
- In parallel, characterizations of lattices obtained by iterating the doubling convex construction were investigated (Day, 1977; Day, 1994; Day et al., 1989; Nation, 1995; Bertet and Caspard, 2002).
- In this paper, we define a reverse construction which removes a convex from a lattice whenever it is possible.

Being able to recover the full lattice from the smaller one and a convex inside, means that all the information is contained in the small lattice. Thus we only need to consider the sub-context that defines the small lattice. In other words, only a part of the data is relevant. Consequently, one needs only to access or even keep a smaller part of the data. This is obviously an interesting way to manage big data that are nowadays ubiquitous in many real-life applications.

This paper is organized as follows. The next section as well as the appendix give the background needed to understand the rest of the article. Then, Section 3 presents the decomposition method while an example serves as an illustration in Section 4. Conclusion and future work are given in Section 5.

2 STRUCTURAL FRAMEWORK

Throughout this paper all sets (and thus lattices) are considered to be finite.

This work takes place in the framework of Formal Concept Analysis (FCA) (Ganter and Wille, 1999) which deals with formal contexts, i.e. data represented as binary matrices. From a formal context, a concept lattice is deduced. An example of a formal context is given in Table 1 and its associated concept lattice is given in Figure 1.

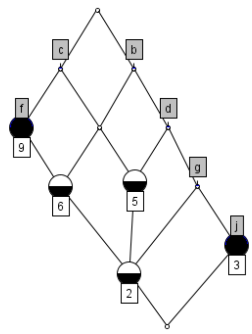

Figure 1: A lattice with its irreducible nodes.

2.1 Lattices and Formal Concept Analysis

2.1.1 Algebraic Lattice

Let us first recall that a *lattice* $(L,\leq)$ is an ordered set in which every pair (x,y) of elements has a *least upper bound*, called *join* $x \vee y$, and a *greatest lower bound*, called *meet* $x \wedge y$. As we are only considering finite structures, every subset $A \subset L$ has a join and meet (e. g. finite lattices are complete).

2.1.2 Concept or Galois Lattice

A (formal) *context* (O,A,R) is defined by a set O of objects, a set A of attributes, and a binary relation $R \subset O \times A$, between O and A. Two operators are derived:

- for each subset $X \subset O$, we define $X' = \{m \in A, j\,R\,m\ \forall j \in X\}$ and dually,
- for each subset $Y \subset A$, we define $Y' = \{j \in O, j\,R\,m\ \forall m \in Y\}$.

Table 1 is an example of a formal context where X in a cell (j,m) means that object j has the attribute m.

A(formal) *concept* represents a maximal objects-attributes correspondence by a pair (X,Y) such that $X' = Y$ and $Y' = X$. The sets X and Y are respectively called *extent* and *intent* of the concept. The set of concepts derived from a context is ordered as follows:

$$(X_1,Y_1) \leq (X_2,Y_2) \Longleftrightarrow X_1 \subseteq X_2 \Longleftrightarrow Y_2 \subseteq Y_1 \quad (1)$$

The whole set of formal concepts together with this order relation form a complete lattice, called the *concept lattice* of the context (O,A,R).

Different formal contexts can provide isomorphic concept lattices, and there exists a unique one, named the *reduced context*, defined by the two sets O and A of the smallest size.

This particular context is introduced by means of

Table 1: The reduced context of the lattice in Figure 1.

	b	c	d	f	g	j
2	x	x	x	x	x	
3	x		x		x	x
5	x	x	x			
6	x	x		x		
9		x		x		

special concepts or elements of the lattice L, namely irreducible elements.

An element $j \in L$ is *join-irreducible* if it is not a least upper bound of a subset not containing it. The set of join irreducible elements is noted J_L. *Meet-irreducible* elements are defined dually and their set is M_L. As a direct consequence, an element $j \in L$ is join-irreducible if and only if it has only one immediate predecessor denoted j^-. Dually, an element $m \in L$ is meet-irreducible if and only if it has only one immediate successor denoted m^+.

In Figure 3, nodes with with a number above are meet-irreducible while nodes with a number below are join-irreducible.

2.1.3 Fundamental Bijection

A fundamental result (Barbut and Monjardet, 1970) establishes that any lattice $(L, \leq)$ is isomorphic to the concept lattice of the context $(J_L, M_L, \leq)$, where J_L and M_L are the join and meet irreducible concepts of L, respectively. Moreover, this context is a reduced one.

As a direct consequence, there is a bijection between lattices and reduced contexts where objects of the context are associated with join-irreducible concepts of the lattice, and attributes are associated with meet-irreducible concepts.

Table 1 in the appendix shows the reduced context of the lattice in Figure 1.

When needed, operators $\bullet'$ introduced in subsection 2.1.2 will be written $\bullet^L$ to stress the fact that they are used with the reduced context of the lattice L.

A sub-context consists of a binary relation between a subset of objects and a subset of attributes in the initial data. In this paper we consider particular sub-contexts for which three different equivalent definitions can be given. More details can be found in (Viaud et al., 2015).

- Compatible sub-contexts,
- Arrow-closed sub-contexts,
- Sub-contexts defining factor lattices and congruence relations.

In conjunction with Day's doubling construction (Day, 1977; Day, 1994), these particular sub-contexts enable us to introduce a new construction, namely the reverse doubling construction.

3 REVERSE DOUBLING CONVEX

In this section, we describe the reverse doubling construction which itself uses congruence relations.

Given a lattice L, we recall that we are looking for a lattice L_C containing a convex C such that $L = L_C[C]$.

3.1 Main Steps

Before giving details about this decomposition, let us provide the main steps of our construction.

3.1.1 Main Steps of the Proof

Given a finite lattice L, we are searching a lattice L_C and a convex set $C \subset L_C$ such that L is isomorphic to $L_C[C]$.

1. We first use following result of Day (Day, 1994):

   Theorem 1. *Let L be a congruence normal lattice and Θ a congruence relation which is an atom, i.e. a successor of the bottom element. Then, there exists a convex set C such that L is isomorphic to $L/\Theta[C]$.*

   From this theorem 1, we know that good candidates can be obtained from factor lattices, spanned by a congruence relation which is an atom.
   It is important to note that we only get good candidates because we do not use the congruence normality hypothesis.

2. We next use the following equivalence whose proof can be found in (Ganter and Wille, 1999):

   Theorem 2. *Given a lattice L, the set of congruence relations on L corresponds bijectively with the set of arrow-closed subcontexts of the reduced context of L.*

From this Theorem 2, we know that these congruence relations correspond to arrow-closed sub-contexts.

3. Then the following theorem 3 of Geyer is used:

   Theorem 3. *Let $L = L_C[C]$ be a lattice obtained by doubling the convex set C of the lattice L_C. Note $K = (O,A,R)$ the reduced context of L, then the reduced context $K_C = (J,M,R \cap J \times M)$ is an arrow-closed subcontext of K such that:*

   $$R \cap ((O \backslash J) \times (A \backslash M)) = \emptyset \qquad (2)$$

   We only need to check if the arrow-closed sub-contexts that are just good candidates satisfy the Geyer's condition 2.

4. If none of these sub-contexts is valid, the decomposition is not possible. Otherwise, each one of these sub-contexts generates a lattice L_C that is appropriate.

3.1.2 Main Steps of the Construction

The principal steps are as follows:

1. The first step of our construction is to compute atoms of the lattice of congruence relations.
2. Then we select the arrow-closed sub-contexts satisfying the Geyer's condition. Each one of these sub-contexts generates a lattice L_C that is appropriate.
3. The last step consists in identifying the convex set C in L_C.

3.2 The Lattice of Congruence Relations

A reduced context (O,A,R) of a lattice L is supposed to be given.

Recall that congruence relations are equivalence relations compatible with the lattice structure. As a particular kind of binary relation, the set of congruence relations inherits a structure of lattice: it is a sublattice of the lattice of binary relations which is ordered by inclusion. Using Theorem 2, this lattice is also the lattice of arrow-closed sub-contexts. As explained previously, we aim at finding the atoms of that lattice. To that end, we introduce a new context, namely the context of arrow-closed subcontexts, which is defined and computed as follows. We first compute one-generated arrow-closed sub-contexts.

Definition 1. *A context (J,M,R) is one-generated if it can be obtained by arrow-closing a context with only one $j \in J$. Thus (J,M,R) is the smallest arrow-closed subcontext containing $j \in J$.*

The set of such one-generated sub-contexts contains all join-irreducible arrow-closed sub-contexts of the lattice of arrow-closed sub-contexts. Thus, the set of one-generated sub-contexts generates the lattice of arrow-closed sub-contexts. The one-generated sub-contexts are stored as observations of a new context in the following way: an arrow-closed is characterized by its set of attributes, since it is closed. Thus the newly generated context has the same set of attributes, namely A. Since one-generated sub-contexts are computed by closing one $j \in O$, they can be identified with that j. The set of observations of the newly generated context is still O. Finally, there is a cross in the new context between object j and attribute a if and only if the arrow-closed sub-context generated by j contains the attribute a.

From this context, we can deduce the lattice of arrow-closed sub-contexts and in particular its atoms.

3.3 The Factor Lattice

We are now given a list of arrow-closed sub-contexts, which correspond to atom congruence relations.

The second step is to remove from this list the following contexts:

- The empty context. In the particular case where the initial lattice has only two congruence relations, there is only one atom, namely the bottom element. This case is excluded since it gives rise to a trivial decomposition.
- Any context that does not satisfy Geyer's Condition.

After this removal process, our list could be empty. Since the Geyer's condition is necessary and sufficient, we can conclude that the original lattice can not be processed through the reverse doubling construction.

On the opposite side, any remaining context generates a lattice L_C such that there exists a convex set C so that the initial lattice L satisfies $L = L_C[C]$.

3.4 Finding the Convex

Given a lattice L_C satisfying the previous conditions of Subsection 3.3, the last step is to identify in L_C nodes of a convex set C such that $L = L_C[C]$.

Using Theorem 3 which gives a necessary and sufficient condition, we already know that L is obtained by the doubling construction from L_C. Thus, each concept of L_C gives rise to one concept in L if and only if it is not in C and two concepts in L if and only if it is in C.

Let us be more precise about these two concepts, and first let us recall some notations: (O,A,R) is the reduced context of the lattice L. We consider $(J,M,R\cap J\times M)$ an atom in the lattice of arrow-closed sub-contexts which satifies the Geyer's condition. From this context, we can deduce L_C the lattice from which a convex has been removed. Let c be a concept of L_C. Recall from (Ganter and Wille, 1999) that arrow-closed sub-contexts are also compatible ones. Since $(J,M,R\cap J\times M)$ is a compatible sub-context, using this result, c can be written as $c=(H\cap J,N\cap M)$ with (H,N) a concept of L. Since c is a concept, we have $(H\cap J)'=N\cap M$ and $(N\cap M)'=H\cap J$. Recall, from the end of Subsection 2.1.3, that these operations are written $\bullet^L$ in the reduced context (O,A,R) of L; thus we also have $H^L=N$ and $N^L=H$. Now, we can conclude that, if $c=(H\cap J,N\cap M)$ is not in the convex, it comes from the unique concept of L, namely (H,N) and thus we must have $((H\cap J)^{LL},(H\cap J)^L)=(H,N)=((N\cap M)^L,(N\cap M)^{LL})$. In the other case, we have $((H\cap J)^{LL},(H\cap J)^L)\neq((N\cap M)^L,(N\cap M)^{LL})$.

From this point, we can state a simpler condition for a concept to be in the convex set:

Theorem 4. *A concept $c=(H\cap J,N\cap M)$ is in the convex if and only if*

$$(H\cap J)^L\subseteq(N\cap M) \text{ or } (N\cap M)^L\subseteq(H\cap J) \quad (3)$$

Proof. We have previously seen that if c is in the convex then:

$$((H\cap J)^{LL},(H\cap J)^L)=(H,N) \quad (4)$$
$$=((N\cap M)^L,(N\cap M)^{LL}) \quad (5)$$

So both inclusions are true and actually are equalities.

Suppose, reciprocally, that we have the first one:

$$(H\cap J)^L\subset(N\cap M) \quad (6)$$

We obtain the same result dually in the second case.

First we prove that the inclusion is an equality. Indeed, since $(J,M,R\cap J\times M)$ is a subcontext of (O,A,R), we have:

$$(H\cap J)'\subset(H\cap J)^L \quad (7)$$

However $(H\cap J)'=(N\cap M)$, so:

$$(N\cap M)\subset(H\cap J)^L \quad (8)$$

and dually

$$(H\cap J)\subset(N\cap M)^L \quad (9)$$

Therefore, we deduce that $(H\cap J)^L=(N\cap M)$, and then $(N\cap M)^L=(H\cap J)^{LL}$.

Moreover:

$$(H\cap J)\subset(N\cap M)^L\Longrightarrow(N\cap M)^{LL}\subset(H\cap J)^L \quad (10)$$

and with $(N\cap M)=(H\cap J)^L$, we deduce that:

$$(N\cap M)=(N\cap M)^{LL} \quad (11)$$

since $(N\cap M)\subset(N\cap M)^{LL}$.

Now from $(H\cap J)^L=(N\cap M)=(N\cap M)^{LL}$ and $(N\cap M)^L=(H\cap J)^{LL}$, we get:

$$((H\cap J)^L,(H\cap J)^{LL})=((N\cap M)^{LL},(N\cap M)^L) \quad (12)$$

This means that a concept $((H\cap J),(N\cap M))$ in the small lattice L_C that satisfies the condition $(H\cap J)^L\subset(N\cap M)$ or $(N\cap M)^L\subset(H\cap J)$ comes from a unique concept in the lattice L, namely the concept:

$$((H\cap J)^{LL},(H\cap J)^L)=(H,N) \quad (13)$$
$$=((N\cap M)^L,(N\cap M)^{LL}) \quad (14)$$

Conversely, if $((H\cap J),(N\cap M))$ does not satisfy the previous condition, it comes from two concepts in L:

$$((H\cap J)^{LL},(H\cap J)^L)\neq((N\cap M)^L,(N\cap M)^{LL}) \quad (15)$$

Thus to get L from L_C, one needs to double these concepts and then these concepts are exactly the ones of the convex. □

4 EXAMPLE

To illustrate the reverse doubling convex construction, we will use the lattice given in Figure 2 and its reduced context given by Table 2 already filled with arrows. Nodes in light grey and dark grey correspond to the two occurrences of the doubled convex.

From the lattice shown in Figure 2, we compute Table 3 of one-generated arrow-closed sub-contexts and then we get the lattice of congruence relations given in Figure 4. Notice that Table 3 has been reduced after computation. In this particular case, there are two atoms.

Since the two atoms identified in Figure 4 satisfy Geyer's condition, the lattice given in Figure 2 has two decompositions.

Using only one of these atoms, namely the one containing observation 25, we get the arrow-closed

Table 2: The reduced context of the lattice in Figure 2 filled with arrows.

	9	10	14	19	33	15	28	7	26	13
2	↓	↕	↓	×	×	×	×	↓	↓	×
4	×	×	×	↕	↓	↓	↓	↓	↓	↓
8	×	×	↕	×	↕	×	×	×	×	×
12	↕	×	↑	×	○	↓	↓	×	×	↕
14	↕	×	×	↑	○	○	○	○	○	○
25	○	↑	○	×	×	×	↕	○	○	×
20	○	↑	○	×	×	↕	×	○	○	×
27	×	×	↑	×	↑	×	↕	×	↕	×
16	×	×	↑	×	↑	↕	×	↕	×	×

Table 3: The *reduced* context computed to get the lattice of congruence relations (Figure 4).

	9	10	19	15	28
2		×			
4			×		
8		×	×		
25	×	×	×		×
20	×	×	×	×	

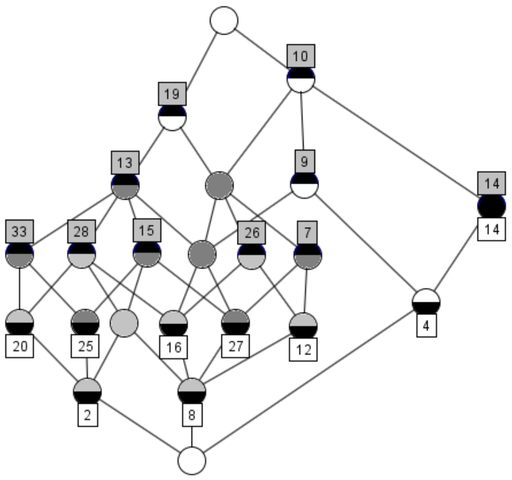

Figure 2: The lattice of Figure 3 with the convex set doubled. Nodes of each convex are colored in light grey and dark grey.

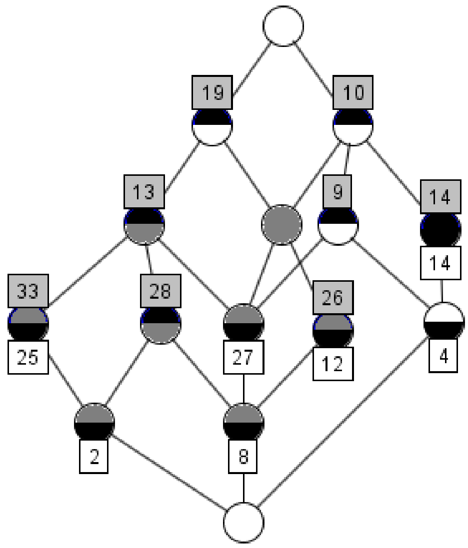

Figure 3: A lattice with its irreducible nodes. Nodes with a number above are meet-irreducible while nodes with a number below are join-irreducible.

subcontext of Table 4, which satisfies Geyer's condition. Then, its concept lattice, which is also the factor lattice of the selected congruence relation, can be computed and visualized on Figure 3.

In Figure 3, nodes of the convex C are in grey, and when the doubling construction is applied on that lattice, with this convex set, the lattice from Figure 2 is obtained. Nodes in light grey and dark grey correspond to the two occurrences of the doubled convex. This lattice is obtained from the reduced context given in Table 4, which is one of the two atoms of the lattice in Figure 4.

In order to test our decomposition method with respect to its ability to reduce the lattice size, a first experiment has been done. A series of 500 random contexts with 20 observations, 10 attributes and a density of 75% was generated. Only 42 of them (8.4%) had a non trivial decomposition. However, 17 of these contexts (40%) led to reduced lattices with a ratio of 50% while the rest (60%) of contexts led to a lattice size reduction higher than 50%. Table 5 reflects such ratios where the first column represents the lattice size reduction (*i.e.*, the number of nodes of the reduced lattice over the number of nodes in the initial lattice)

Table 4: The reduced context of the factor lattice given in Figure 3.

	10	13	14	19	26	28	33	9
12	×			×	×			
14	×		×					
2		×		×		×	×	
25		×		×			×	
27	×	×		×				×
4	×		×					×
8	×			×	×	×		×

Table 5: Lattice size reduction ratio and proportion of reduced lattices.

Ratio of the numbers of nodes	Number and proportion of contexts
50%	17 - 40%
50% to 60%	9 - 21%
60% to 70%	6 - 14%
70% to 80%	5 - 12%
80% to 90%	4 - 10%
90% to 100%	1 - 3%

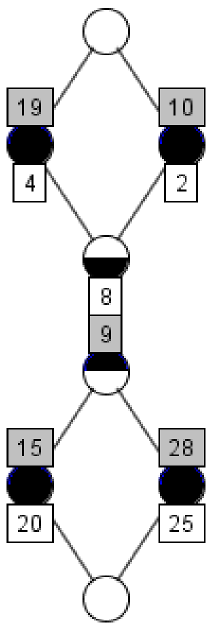

Figure 4: The lattice of congruence relations of the lattice given in Figure 2.

while the second column indicates the number and proportion of contexts for which the reduction ratio is in the interval given in the first column. We can then conclude that the decomposition is rarely possible, but when it occurs, it often leads to a significant reduction in the lattice size.

All computations were done using the Galactic project available at http://thegalactic.github.io/ while the lattices were drawn with ConExp (see http://conexp.sourceforge.net/).

5 CONCLUSION AND FUTURE WORK

In this paper, a new decomposition based on Day's construction was given which uses congruence relations. Although Day's doubling construction has been widely studied, the reverse doubling construction we are investigating is, to the best of our knowledge, a new decomposition method that could be helpful in dealing with large contexts and their visualization.

To further investigate the reverse doubling construction, it would be interesting to conduct large-scale experiments on very large real-life data, not only to test the potential of decomposition for lattice reduction but also its performance and scalability for such large data sets. Since noisy data occur frequently in a process of data analysis, we plan to carefully perform a preprocessing step in which efficient data cleaning techniques will be identified and used.

Since the empirical study in (Snelting, 2005) shows that many real-life contexts do not contain non-trivial congruence relations, we plan to formally identify cases in which the reverse doubling construction can either be used or not.

Our future work consists to study, compare and combine other decompositions with the one described in this paper. In particular the Fratini congruence (Duquenne, 2010) which exploits again a congruence relation, and Atlas decomposition (Ganter and Wille, 1999) which uses tolerance relations.

REFERENCES

Barbut, M. and Monjardet, B., editors (1970). *L'ordre et la classification*. Algèbre et combinatoire, tome II. Hachette.

Belohlavek, R. and Vychodil, V. (2010). Discovery of optimal factors in binary data via a novel method of matrix decomposition. *Journal of Computer and System Sciences*, 76(1):3–20.

Bertet, K. and Caspard, N. (2002). Doubling convec sets in lattices: characterizations and recognition algorithms. Technical Report TR-LACL-2002-08, LACL (Laboratory of Algorithms, Complexity and Logic), University of Paris-Est (Paris 12).

Day, A. (1977). Splitting lattices generate all lattices. *algebra universalis*, 7(1):163–169.

Day, A. (1994). Congruence normality: The characterization of the doubling class of convex sets. *algebra universalis*, 31(3):397–406.

Day, A., Nation, J., and Tschantz, S. (1989). Doubling convex sets in lattices and a generalized semidistributivity condition. *Order*, 6(2):175–180.

Demel, J. (1982). Fast algorithms for finding a subdirect decomposition and interesting congruences of finite algebras. *Kybernetika (Prague)*, 18(2):121–130.

Duquenne, V. (2010). Lattice drawings and morphisms. In *Formal Concept Analysis, 8th International Conference, ICFCA 2010, Agadir, Morocco, March 15-18, 2010. Proceedings*, pages 88–103.

Ferré, S. (2014). Reconciling expressivity and usability in information access from file systems to the semantic web. Rapport hdr, University Rennes 1.

Freese, R. (1997). Computing congruence lattices of finite lattices. *Proceedings of the American Mathematical Society*, 125(12):3457–3463.

Freese, R. (1999). Algorithms in finite, finitely presented and free lattices. *Preprint, July*, 22:159–178.

Freese, R. (2008). Computing congruences efficiently. *Algebra universalis*, 59(3-4):337–343.

Funk, P., Lewien, A., and Snelting, G. (1995). Algorithms for concept lattice decomposition and their applications. Technical report, TU Braunschweig.

Ganter, B. and Wille, R. (1999). *Formal concept analysis - mathematical foundations.* Springer.

Geyer, W. (1994). The generalized doubling construction and formal concept analysis. *algebra universalis*, 32(3):341–367.

Mihók, P. and Semanišin, G. (2008). Unique factorization theorem and formal concept analysis. In Yahia, S., Nguifo, E., and Belohlavek, R., editors, *Concept Lattices and Their Applications*, volume 4923 of *Lecture Notes in Computer Science*, pages 232–239. Springer Berlin Heidelberg.

Nation, J. (1995). Alan day's doubling construction. *algebra universalis*, 34(1):24–34.

Snelting, G. (2005). Concept lattices in software analysis. In *Formal Concept Analysis, Foundations and Applications*, pages 272–287.

Viaud, J.-F., Bertet, K., Demko, C., and Missaoui, R. (2015). Subdirect decomposition of contexts into subdirectly irreducible factors. *Formal Concept Analysis and Applications FCA&A 2015*, page 49.

Visani, M., Bertet, K., and Ogier, J.-M. (2011). Navigala: an Original Symbol Classifier Based on Navigation through a Galois Lattice. *International Journal on Pattern Recognition and Artificial Intelligence (IJPRAI)*.

Wille, R. (1969). Subdirekte produkte und konjunkte summen. *Journal für die reine und angewandte Mathematik*, 0239_0240:333–338.

Wille, R. (1976). Subdirekte Produkte vollständiger Verbände. *J. reine angew. Math.*, 283/284:53–70.

Wille, R. (1983). Subdirect decomposition of concept lattices. *Algebra Universalis*, 17:275–287.

Wille, R. (1987). Subdirect product construction of concept lattices. *Discrete Mathematics*, 63(2-3):305–313.