Fast Factorization of Concept Lattices by Similarity: Solution and an Open Problem

Radim Bělohlávek, Jiří Dvořák, and Jan Outrata

Department of Computer Science, Palacky University, Olomouc
Tomkova 40, CZ-779 00 Olomouc, Czech Republic
{radim.belohlavek, jiri.dvorak, jan.outrata}@upol.cz

Abstract. An important problem in applications of formal concept analysis is a possibly large number of clusters extracted from data. Factorization is one of the methods being used to cope with the number of clusters. We present an algorithm for computing a factor lattice of a concept lattice from the data and a user-specified similarity threshold a. The elements of the factor lattice are collections of clusters which are pairwise similar in degree at least a. The presented algorithm computes the factor lattice directly from the data, without first computing the whole concept lattice and then computing the collections of clusters. We present theoretical insight and examples for demonstration, and an open problem.

Keywords: formal concept analysis, fuzzy attributes, factorization, similarity

1 Problem Setting and Preliminaries

1.1 Problem Setting

The problem The present paper presents a solution to a problem formulated at CLA 2002 Workshop (Horí Bečva, Czech Republic, June 2002) concerning factorization of concept lattice over data with fuzzy attributes. Formulated briefly, the problem is as follows: Find a fast way to compute the factor concept lattice over data with fuzzy attributes as described in [2] (factorization is by similarity which is given by a user-specified threshold a). In addition to the solution, we formulate another open problem related to the present one. The present paper is a brief version of a more detailed paper [6] which is under preparation (due to the limited scope, we omit proofs and shorten demonstrating examples and comments in the present paper).

⋆ R. Bělohlávek gratefully acknowledges support by grant No. 201/02/P076 of the GAČR.

© V. Snášel, R. Bělohlávek (Eds.): CLA 2004, pp. 47–57, ISBN 80-248-0597-9.
VŠB – Technical University of Ostrava, Dept. of Computer Science, 2004.

The context of the problem We assume basic familiarity with formal concept analysis (FCA) [8], and with fuzzy logic and fuzzy sets [3, 10]. It is well-known that an important problem of FCA is the possible large number of formal concepts (clusters) in data. One of the ways to cope with this problem is factorization of concept lattices [8]. In [2], a method to factorize concept lattices over data with fuzzy attributes was proposed. Basically, a user specifies a similarity threshold a and the resulting factor lattice contains as its elements the maximal groupings of formal concepts (elements of the original "large" concept lattice over the data) which are pairwise similar in degree at least a. Parameter a controls the coarseness of the factorization and thus the factor of reduction (for a running from 0 over ... to 1 we obtain a one-element lattice over ... to a lattice which is isomorphic to the original concept lattice). The resulting factor concept lattice can be computed by definition as follows: (a) compute the "large" (the original, non-factorized concept lattice); (b) compute the factor concept lattice of the large concept lattice. Although polynomial time delay algorithms exist for both (a) and (b), it is interesting to ask whether there is a way to compute the factor lattice directly from the data, i.e. without the need to compute first the "large" concept lattice. In what follows, we show a positive answer and demonstrate its efficiency on examples.

1.2 Preliminaries

Fuzzy sets and fuzzy logic We assume basic familiarity with fuzzy logic and fuzzy sets [3, 10]. An element may belong to a fuzzy set in an intermediate degree not necessarily being 0 or 1. Formally, a fuzzy set A in a universe X is a mapping assigning to each $x \in X$ a truth degree $A(x) \in L$ where L is some partially ordered set of truth degrees containing at least 0 (full falsity) and 1 (full truth). L needs to be equipped with logical connectives, e.g. $\otimes$ (fuzzy conjunction), $\rightarrow$ (fuzzy implication), etc. L together with logical connectives forms a structure $\mathbf{L}$ of truth degrees. We assume that $\mathbf{L}$ forms a so-called residuated lattice in which arbitrary infima $\bigwedge$ and suprema $\bigvee$ exist.

The set of all fuzzy sets (or $\mathbf{L}$-sets) in X is denoted L^X. For fuzzy sets A, B in X we put $A \subseteq B$ (A is a subset of B) if for each $x \in X$ we have $A(x) \leq B(x)$. More generally, the degree $S(A, B)$ to which A is a subset of B is defined by $S(A, B) = \bigwedge_{x \in X} A(x) \rightarrow B(x)$. Then, $A \subseteq B$ means $S(A, B) = 1$.

Formal concept analysis of data with fuzzy attributes Let X and Y be sets of objects and attributes, respectively, I be a fuzzy relation between X and Y; $I(x, y) \in L$ is the degree to which object x has attribute y. The triplet $\langle X, Y, I \rangle$ is called a formal fuzzy context (a data table with fuzzy attributes).

For fuzzy sets $A \in L^X$ and $B \in L^Y$, define fuzzy sets $A^\uparrow \in L^Y$ and $B^\downarrow \in L^X$ by

$$A^\uparrow(y) = \bigwedge_{x \in X} (A(x) \rightarrow I(x, y)) \quad (1), \qquad B^\downarrow(x) = \bigwedge_{y \in Y} (B(y) \rightarrow I(x, y)) \quad (2).$$

Then $A^{\uparrow}(y)$ is the truth degree of the fact "y is shared by all objects from A" and $B^{\downarrow}(x)$ is the truth degree of the fact "x has all attributes from B". Put $\mathcal{B}(X,Y,I) = \{\langle A,B\rangle \mid A^{\uparrow} = B,\ B^{\downarrow} = A\}$. Elements of $\mathcal{B}(X,Y,I)$ are called formal concepts of $\langle X,Y,I\rangle$ (interesting clusters in data); $\mathcal{B}(X,Y,I)$ is called the concept lattice given by $\langle X,Y,I\rangle$.

Putting $\langle A_1,B_1\rangle \leq \langle A_1,B_1\rangle$ iff $A_1 \subseteq A_2$ (iff $B_1 \supseteq B_2$) for $\langle A_1,B_1\rangle, \langle A_2,B_2\rangle \in \mathcal{B}(X,Y,I)$, $\leq$ models the subconcept-superconcept hierarchy in $\mathcal{B}(X,Y,I)$ (being more general means to apply to a larger collection of objects and to cover a smaller collection of attributes). The structure of $\mathcal{B}(X,Y,I)$ is characterized in [4]. For further information on fuzzy concept lattices, see e.g. [3, 7].

2 Fast factorization by similarity

2.1 Factorization by similarity

In this section, we recall the method presented in [2]. Given $\langle X,Y,I\rangle$, introduce a binary fuzzy relation $\approx$ on $\mathcal{B}(X,Y,I)$ by $(\langle A_1,B_1\rangle \approx \langle A_2,B_2\rangle) = \bigwedge_{x\in X} A_1(x) \leftrightarrow A_2(x)$ for $\langle A_i,B_i\rangle \in \mathcal{B}(X,Y,I)$, $i = 1,2$, where $a \leftrightarrow b = (a \to b) \wedge (b \to a)$. $(\langle A_1,B_1\rangle \approx \langle A_2,B_2\rangle)$ is called the degree of similarity of $\langle A_1,B_1\rangle$ and $\langle A_2,B_2\rangle$ (just the truth degree of "for each object x: x is covered by A_1 iff x is covered by A_2"). One can show that $(\langle A_1,B_1\rangle \approx \langle A_2,B_2\rangle) = \bigwedge_{y\in Y} B_1(y) \leftrightarrow B_2(y)$.

Given a truth degree $a \in L$ (a threshold specified by a user), consider the thresholded relation ${}^a\!\approx$ on $\mathcal{B}(X,Y,I)$ defined by $(\langle A_1,B_1\rangle, \langle A_2,B_2\rangle) \in {}^a\!\approx$ iff $(\langle A_1,B_1\rangle \approx \langle A_2,B_2\rangle) \geq a$. That is, ${}^a\!\approx$ denotes "being similar in degree at least a". ${}^a\!\approx$ is reflexive and symmetric, but need not be transitive. Call a subset B of $\mathcal{B}(X,Y,I)$ a ${}^a\!\approx$-block if it is a maximal subset of $\mathcal{B}(X,Y,I)$ such that each two concepts from B are similar in degree at least a. Denote by $\mathcal{B}(X,Y,I)/{}^a\!\approx$ the collection of all ${}^a\!\approx$-blocks. Put

$$\langle A,B\rangle_a := \bigwedge\{\langle A',B'\rangle \mid (\langle A,B\rangle, \langle A',B'\rangle) \in {}^a\!\approx\}$$
$$\langle A,B\rangle^a := \bigvee\{\langle A',B'\rangle \mid (\langle A,B\rangle, \langle A',B'\rangle) \in {}^a\!\approx\}.$$

Lemma 1. *${}^a\!\approx$-blocks are exactly intervals of $\mathcal{B}(X,Y,I)$ of the form $[\langle A,B\rangle_a, (\langle A,B\rangle_a)^a]$, i.e.*

$$\mathcal{B}(X,Y,I)/{}^a\!\approx \ = \ \{[\langle A,B\rangle_a, (\langle A,B\rangle_a)^a] \mid \langle A,B\rangle \in \mathcal{B}(X,Y,I)\}.$$

Now, define a partial order $\preceq$ on blocks of $\mathcal{B}(X,Y,I)/{}^a\!\approx$ by $[c_1,c_2] \preceq [d_1,d_2]$ iff $c_1 \leq d_1$ (iff $c_2 \leq d_2$) where $[c_1,c_2],[d_1,d_2] \in \mathcal{B}(X,Y,I)/{}^a\!\approx$, i.e. c_1,c_2,d_1,d_2 are suitable formal concepts from $\mathcal{B}(X,Y,I)$ and $c_i \leq d_i$ denotes that in $\mathcal{B}(X,Y,I)$, c_i is under (a subconcept of) d_i. Then we have

Theorem 1. *$\mathcal{B}(X,Y,I)/{}^a\!\approx$ equipped with $\preceq$ is a partially ordered set which is a complete lattice, the so-called factor lattice of $\mathcal{B}(X,Y,I)$ by similarity $\approx$ and a threshold a.*

Elements of $\mathcal{B}(X,Y,I)/^a\!\approx$ can be seen as similarity-based granules of formal concepts/clusters from $\mathcal{B}(X,Y,I)$.$\mathcal{B}(X,Y,I)/^a\!\approx$ thus provides a granular view on (the possibly large) $\mathcal{B}(X,Y,I)$. For details we refer to [2].

We now present an illustrative example. Consider $\mathbf{L}$ with $L = \{0, \frac{1}{2}, 1\}$ and Łukasiewicz fuzzy logical connectives. Consider the data in Tab. 1. X contains nine objects (Mercury, ..., Pluto), Y contains four attributes ("size small", ..., "near to sun"). The corresponding concept lattice is depicted in Fig. 1.

Table 1. A simple fuzzy context given by planets and their properties

	small	large	far	near
Mercury (Me)	1	0	0	1
Venus (V)	1	0	0	1
Earth (E)	1	0	0	1
Mars (Ma)	1	0	$\frac{1}{2}$	1
Jupiter (J)	0	1	1	$\frac{1}{2}$
Saturn (S)	0	1	1	$\frac{1}{2}$
Uranus (U)	$\frac{1}{2}$	$\frac{1}{2}$	1	0
Neptune (N)	$\frac{1}{2}$	$\frac{1}{2}$	1	0
Pluto (P)	1	0	1	0

Consider now the $a = \frac{1}{2}$. There are twelve $^{1/2}\!\approx$-blocks and they are depicted

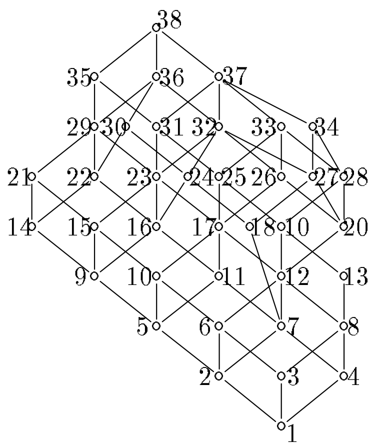


Fig. 1. Concept lattice $\mathcal{B}(X,Y,I)$ of data in Tab. 1

in Fig. 2 (blocks are higlighted by solid lines). The corresponding factor lattice $\mathcal{B}(X,Y,I)/^{\frac{1}{2}}\!\approx$ is depicted in Fig. 3.

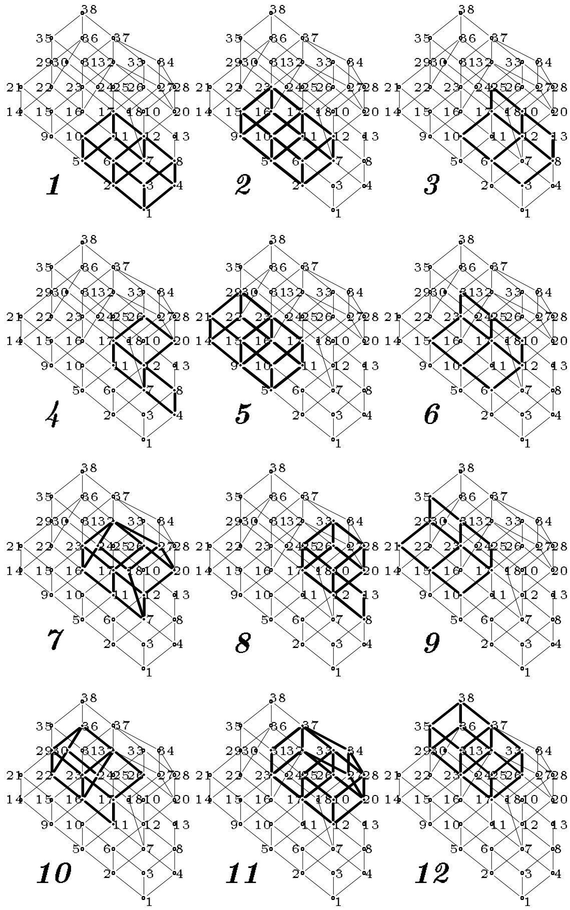


Fig. 2. $\frac{1}{2}\!\approx$-blocks on the concept lattice of Fig. 1

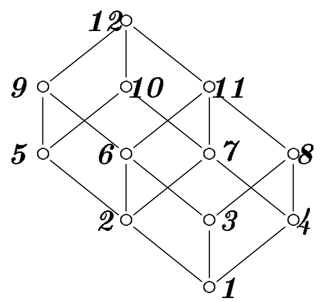


Fig. 3. Factor lattice $\mathcal{B}\,(X, Y, I)/^{\frac{1}{2}}\!\approx$

2.2 Computing the factor lattice $\mathcal{B}(X, Y, I)/{}^a\!\approx$ directly from input data

We are going to propose a way to compute $\mathcal{B}(X, Y, I)/{}^a\!\approx$ directly from input data. It will turn out that our algorithm has a polynomial time delay (see [9]). We present the solution step-by-step but, due to the limited scope, without proofs. For a fuzzy set C in U and $a \in L$, the fuzzy sets $a \to C$ and $a \otimes C$ in U are defined by $(a \to C)(u) = a \to C(u)$ and $(a \otimes C)(u) = a \otimes C(u)$ for each $u \in U$. For fuzzy sets C, D in U, put $(C \approx D) = \bigwedge_{u \in U} C(u) \leftrightarrow D(u)$. Furthermore, we call a fuzzy set A in X an extent if there is a fuzzy set B in Y such that $\langle A, B\rangle \in \mathcal{B}(X, Y, I)$ (similarly, B is an intent if there is A with $\langle A, B\rangle \in \mathcal{B}(X, Y, I)$).

Lemma 2. *If A is an extent then so is $a \to A$; similarly, if B is an intent then so is $a \to B$.*

Proof. See [6].

The next lemma shows that for a formal concept $\langle A, B\rangle$, $\langle A, B\rangle_a$ and $\langle A, B\rangle^a$ (defined as infimum and supremum of all formal concepts similar to $\langle A, B\rangle$ in degree at least a) can be computed from $\langle A, B\rangle$ directly.

Lemma 3. *For $\langle A, B\rangle \in \mathcal{B}(X, Y, I)$, we have (a) $\langle A, B\rangle_a = \langle (a \otimes A)^{\uparrow\downarrow}, a \to B\rangle$ and (b) $\langle A, B\rangle^a = \langle (a \to A), (a \otimes B)^{\downarrow\uparrow}\rangle$.*

Proof. See [6].

Thus we have $(\langle A, B\rangle_a)^a = \langle a \to (a \otimes A)^{\uparrow\downarrow}, (a \otimes (a \to B))^{\downarrow\uparrow}\rangle$.

Lemma 4. *For $\langle A, B\rangle \in \mathcal{B}(X, Y, I)$ we have $\langle A, B\rangle_a = (((\langle A, B\rangle_a)^a)_a$.*

Proof. See [6].

By Lemma 4, if a ${}^a\!\approx$-block $[c_1, c_2]$ is generated by $\langle A, B\rangle \in \mathcal{B}(X, Y, I)$, i.e. $c_1 = \langle A, B\rangle_a$, $c_2 = (\langle A, B\rangle_a)^a$, then it is also generated by c_2, i.e. $c_1 = (c_2)_a$ and $c_2 = ((c_2)_a)^a$. Therefore, ${}^a\!\approx$-blocks $[c_1, c_2]$ are uniquely given by their suprema c_2. Moreover, since each formal concept $c_2 = \langle A, B\rangle$ is uniquely given by A (namely, $B = A^{\uparrow}$), ${}^a\!\approx$-blocks are uniquely given by extents of their suprema. Therefore, denote the set of all extents of suprema of ${}^a\!\approx$-blocks by $\mathrm{ESB}(a)$, i.e.

$$\mathrm{ESB}(a) = \{A \in L^X \mid \langle A, B\rangle \in \mathcal{B}(X, Y, I),\ [\langle A, B\rangle_a, \langle A, B\rangle] \in \mathcal{B}(X, Y, I)/{}^a\!\approx\}.$$

We are going to present the main result. Let $C : A \to C(A)$ be a mapping (assigning a fuzzy set $C(A)$ in X to a fuzzy set A in X). A fixed point of C is any fuzzy set A in X such that $A = C(A)$. Let $\mathrm{fix}(C)$ denote the set of all fixed points of C, i.e. $\mathrm{fix}(C) = \{A \in L^X \mid A = C(A)\}$.

Recall (see e.g. [5]) that C is called a fuzzy closure operator in X if $A \subseteq C(A)$, $S(A_1, A_2) \leq S(C(A_1), C(A_2))$, $C(A) = C(C(A))$, for any $A, A_1, A_2 \in L^X$.

Theorem 2. *Given input data $\langle X, Y, I\rangle$ and a threshold $a \in L$, a mapping C_a sending a fuzzy set A in X to a fuzzy set $a \to (a \otimes A)^{\uparrow\downarrow}$ in X is a fuzzy closure operator in X for which* $\mathrm{fix}(C_a) = \mathrm{ESB}(a)$.

Proof. See [6].

Therefore, A is a fixed point of C_a if and only if A is the extent of some formal concept c_2 which is the supremum of some ${}^a{\approx}$-block $[c_1, c_2] \in \mathcal{B}(X, Y, I)/{}^a{\approx}$.

Remark 1. Suppose we can compute $\mathrm{fix}(C_a)$ (we will se later how to do it). By Theorem 2 and the above considerations, going through $\mathrm{fix}(C_a)$ and computing for each $A \in \mathrm{fix}(C_a)$ the corresponding $[\langle A, A^{\uparrow}\rangle_a, \langle A, A^{\uparrow}\rangle] = [\langle (a \otimes A)^{\uparrow\downarrow}, a \to A^{\uparrow}\rangle, \langle A, A^{\uparrow}\rangle]$ generates all ${}^a{\approx}$-blocks of $\mathcal{B}(X, Y, I)/{}^a{\approx}$.

Remark 2. Strictly speaking, proceeding the just-described way, we do not generate the ${}^a{\approx}$-blocks $[c_1, c_2] \in \mathcal{B}(X, Y, I)/{}^a{\approx}$, i.e. we do not generate ${}^a{\approx}$-blocks $[c_1, c_2]$ as collections of formal concepts $[c_1, c_2] = \{\langle A, B\rangle \mid c_1 \leq \langle A, B\rangle \leq c_2\}$. For us, generating a ${}^a{\approx}$-block $[c_1, c_2]$ means generating the boundary formal concepts $c_1, c_2 \in \mathcal{B}(X, Y, I)$. This is, however, in acordance with the purpose of the factorization of $\mathcal{B}(X, Y, I)$: We are looking for a granular view which is more concise than $\mathcal{B}(X, Y, I)$ itself.

Let us turn to the problem of generating $\mathrm{fix}(C_a)$. To this end, we can use the algorithm for generating all formal concepts of a given fuzzy context described in [5]. Indeed, the algorithm described in [5] generates extents of all formal concepts from $\mathcal{B}(X, Y, I)$. Now, the extents of formal concepts are exactly the fixed points of a fuzzy closure operator C defined by $C(A) = A^{\uparrow\downarrow}$. Furthermore, as one can check, as the algorithm uses only properties of fuzzy closure operators, it is in fact an algorithm for generating the set of fixed points of a fuzzy closure operator. Adapting the algorithm for our situation and taking in account Remark 1, we get the following algorithm for computing ${}^a{\approx}$-blocks $[c_1, c_2]$, i.e. elements of $\mathcal{B}(X, Y, I)/{}^a{\approx}$:

Suppose $X = \{1, 2, \ldots, n\}$; $L = \{0 = a_1 < a_2 < \cdots < a_k = 1\}$ (the assumption that L is linearly ordered is in fact not essential). For $i, r \in \{1, \ldots, n\}$, $j, s \in \{1, \ldots, k\}$ we put $(i, j) \leq (r, s)$ iff $i < r$ or $i = r, a_j \geq a_s$. In the following, we will freely refer to a_i just by i, thus not distinguish between $X \times L$ and $\{1, \ldots, n\} \times \{1, \ldots, k\}$, i.e. we denote $(i, a_j) \in X \times L$ also simply by (i, j). For $A \in L^X$, $(i, j) \in X \times L$, put

$$A \oplus (i, j) := C_a((A \cap \{1, 2, \ldots, i-1\}) \cup \{{}^{a_j}/i\}).$$

Here, $A \cap \{1, 2, \ldots, i-1\}$ is the intersection of a fuzzy set A and the ordinary set $\{1, 2, \ldots, i-1\}$, i.e. $(A \cap \{1, 2, \ldots, i-1\})(x) = A(x)$ for $x < i$ and $(A \cap \{1, 2, \ldots, i-1\})(x) = 0$ otherwise. Furthermore, for $A, C \in L^X$, put

$$A <_{(i,j)} C \text{ iff } A \cap \{1, \ldots, i-1\} = C \cap \{1, \ldots, i-1\} \text{ and } A(i) < C(i) = a_j.$$

Finally, $A < C$ iff $A <_{(i,j)} C$ for some (i, j). The algorithm is based on the following theorem (see [5]).

Theorem 3. *The least fixed point A^+ which is greater (w.r.t. $<$) than a given $A \in L^X$ is given by $A^+ = A \oplus (i,j)$ where (i,j) is the greatest one with $A <_{(i,j)} A \oplus (i,j)$.*

The algorithm for generating ${}^a\!\approx$-blocks follows.

INPUT: $\langle X, Y, I\rangle$ (data table with fuzzy attributes), $a \in L$ (similarity threshold)
OUTPUT: $\mathcal{B}(X,Y,I)/{}^a\!\approx$ (${}^a\!\approx$-blocks $[c_1, c_2]$)

$A := \emptyset$
`while` $A \neq X$ `do`
  $A := A^+$
  `store`$([\langle (a \otimes A)^{\uparrow\downarrow}, a \rightarrow A^{\uparrow}\rangle, \langle A, A^{\uparrow}\rangle])$

As argued in [5], generating fix(C_a) has polynomial time delay complexity (i.e., given a fixed point, the next one is generated in time polynomial in terms of size of the input $\langle X, Y, I\rangle$ [9]). Since generating a ${}^a\!\approx$-block $[\langle (a \otimes A)^{\uparrow\downarrow}, a \rightarrow A^{\uparrow}\rangle, \langle A, A^{\uparrow}\rangle]$ from A takes a polynomial time, our algorithm is of polynomial time delay complexity as well.

3 Examples and experiments

Due to the limited scope, we demonstrate our algorithm on a data table (fuzzy context) from Tab. 2 for which we consider various parameters a (threshold) and some characteristics for comparison. The data table contains countries (objects from X) and some of their economic characteristics (attributes from Y). The original values of the characteristics are scaled to interval $[0,1]$ so that the characteristics can be considered as fuzzy attributes. Tab. 3 summarizes the effect of our algorithm and some related characteristics when using Łukasiewicz fuzzy logical connectives. The whole concept lattice $\mathcal{B}(X,Y,I)$ contains 774 formal concepts, computing $\mathcal{B}(X,Y,I)$ using the polynomial time delay algorithm from [5] takes 2292ms. The columns correspond to different threshold values $a = 0.2, 0.4, 0.6, 0.8$. Entries "size $|\mathcal{B}(X,Y,I)/{}^a\!\approx|$" contain the number of ${}^a\!\approx$-blocks; "naive algorithm (ms)" contain the time in ms for computing $\mathcal{B}(X,Y,I)/{}^a\!\approx$ by first generating $\mathcal{B}(X,Y,I)$ and subsequently generating the ${}^a\!\approx$-blocks by producing $[\langle A,B\rangle_a, (\langle A,B\rangle_a)^a]$; "our algorithm (ms)" contain the time in ms for computing $\mathcal{B}(X,Y,I)/{}^a\!\approx$ by our algorithm; "reduction $|\mathcal{B}(X,Y,I)/{}^a\!\approx|/|\mathcal{B}(X,Y,I)|$" contain the reduction factors of the size of the concept lattice; "time reduction" contain "our algorithm (ms)" divided by "naive algorithm (ms)" (1/"time reduction" is thus the speedup). Fig. 4 contains graphs depicting reduction $|\mathcal{B}(X,Y,I)/{}^a\!\approx|/|\mathcal{B}(X,Y,I)|$ and time reduction from Tab. 3.

The example demonstrates that smaller thresholds lead to larger reduction (in time and size of the concept lattice). Furthermore, we can see that the time needed for computing the factor lattice $\mathcal{B}(X,Y,I)/{}^a\!\approx$ is smaller than time for computing the original concept lattice $\mathcal{B}(X,Y,I)$.

Table 2. Data table (fuzzy context)

	1	2	3	4	5	6	7
1 Czech	0.4	0.4	0.6	0.2	0.2	0.4	0.2
2 Hungary	0.4	1.0	0.4	0.0	0.0	0.4	0.2
3 Poland	0.2	1.0	1.0	0.0	0.0	0.0	0.0
4 Slovakia	0.2	0.6	1.0	0.0	0.2	0.2	0.2
5 Austria	1.0	0.0	0.2	0.2	0.2	1.0	1.0
6 France	1.0	0.0	0.6	0.4	0.4	0.6	0.6
7 Italy	1.0	0.2	0.6	0.0	0.2	0.6	0.4
8 Geramny	1.0	0.0	0.6	0.2	0.2	1.0	0.6
9 UK	1.0	0.2	0.4	0.0	0.2	0.6	0.6
10 Japan	1.0	0.0	0.4	0.2	0.2	0.4	0.2
11 Canada	1.0	0.2	0.4	1.0	1.0	1.0	1.0
12 USA	1.0	0.2	0.4	1.0	1.0	0.2	0.4

attributes: 1 - Gross Domestic Product per capita (USD), 2 - Consumer Price Index (1995=100) , 3 - Unemployment Rate (percent - ILO), 4 - Production of electricity per capita (kWh), 5 - Energy consumption per capita (GJ), 6 - Export per capita (USD), 7 - Import per capita (USD)

Table 3. Łukasiewicz fuzzy logical connectives, $\mathcal{B}(X, Y, I)$ of data from Tab. 2: $|\mathcal{B}(X, Y, I)| = 774$, time for computing $\mathcal{B}(X, Y, I) = 2292$ ms; table entries for thresholds $a = 0.2, 0.4, 0.6, 0.8$

	0.2	0.4	0.6	0.8
size $\lvert\mathcal{B}(X, Y, I)/^a\!\approx\rvert$	8	57	193	423
naive algorithm (ms)	8995	9463	8573	9646
our algorithm (ms)	23	214	383	1517
reduction $\lvert\mathcal{B}(X, Y, I)/^a\!\approx\rvert/\lvert\mathcal{B}(X, Y, I)\rvert$	0.010	0.073	0.249	0.546
time reduction	0.002	0.022	0.044	0.157

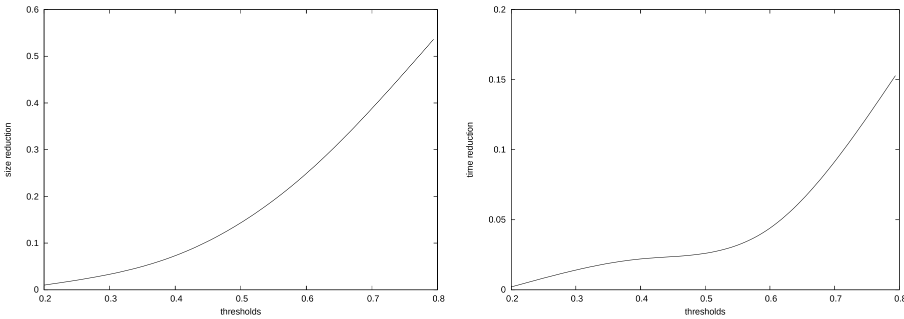


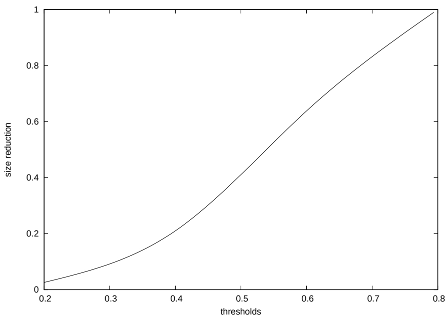
Fig. 4. Reduction $|\mathcal{B}(X, Y, I)/^a\!\approx|/|\mathcal{B}(X, Y, I)|$ and time reduction from Tab. 3

Tab. 4 and Fig. 5 show the same characteriztics when using the minimum-based fuzzy logical operations.

Finally, we demonstrate the effects on an example of data table from Tab. 5 with a finer distribution of thresholds, $a = 0.1, 0.2, \ldots, 0.9$. Using Łukasiewicz fuzzy logical operations, the characteristics are the same as for the above example and are depicted in Fig. 6.

Table 4. Minimum-based fuzzy logical connectives, $\mathcal{B}(X, Y, I)$ of data from Tab. 2: $|\mathcal{B}(X, Y, I)| = 304$, time for computing $\mathcal{B}(X, Y, I) = 341$ ms; table entries for thresholds $a = 0.2, 0.4, 0.6, 0.8$

	0.2	0.4	0.6	0.8
size $\lvert\mathcal{B}(X, Y, I)/^a\!\approx\rvert$	8	64	194	304
naive algorithm (ms)	1830	1634	3787	4440
our algorithm (ms)	23	106	431	1568
reduction $\lvert\mathcal{B}(X, Y, I)/^a\!\approx\rvert / \lvert\mathcal{B}(X, Y, I)\rvert$	0.026	0.210	0.638	1.000
time reduction	0.012	0.064	0.113	0.353



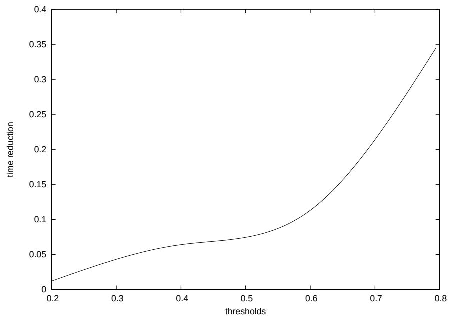


Fig. 5. Reduction $|\mathcal{B}(X, Y, I)/^a\!\approx| / |\mathcal{B}(X, Y, I)|$ and time reduction from Tab. 4

4 Open problem

Is there a suitable context-factorization construction by similarity such that for the factorized context $\langle X, Y, I\rangle/^a\!\approx$, the concept lattice $\mathcal{B}(\langle X, Y, I\rangle/^a\!\approx)$ over $\langle X, Y, I\rangle/^a\!\approx$ is isomorphic to $\mathcal{B}(X, Y, I)/^a\!\approx$?

Table 5. Data table (fuzzy context)

	1	2	3	4	5
	1.0	0.8	0.2	0.3	0.5
	0.8	1.0	0.2	0.6	0.9
	0.2	0.3	0.2	0.3	0.4
	0.4	0.7	0.1	0.2	0.3
	1.0	0.9	0.3	0.2	0.4

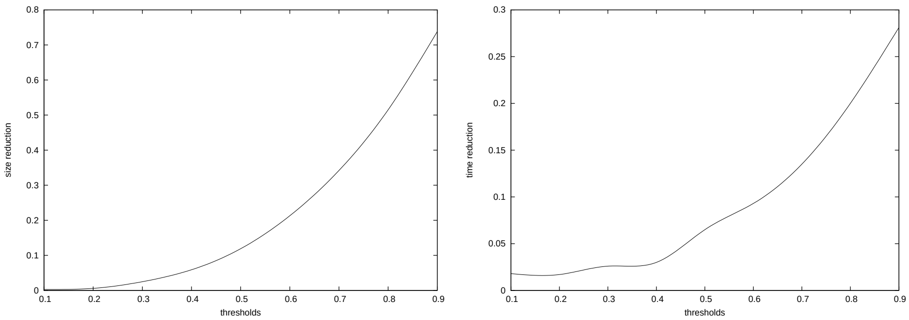


Fig. 6. Reduction $|\mathcal{B}(X,Y,I)/^a\approx|/|\mathcal{B}(X,Y,I)|$ and time reduction from Tab. 5

References

1. R Bělohlávek. Fuzzy concepts and conceptual structures: induced similarities. In *Proc. Joint Conf. Inf. Sci.'98*, Vol. I, pages 179–182, Durham, NC, 1998.
2. R Bělohlávek. Similarity relations in concept lattices. *J. Logic Comput.* 10(6):823–845, 2000.
3. R. Bělohlávek. *Fuzzy Relational Systems: Foundations and Principles.* Kluwer Academic/Plenum Publishers, New York, 2002.
4. R Bělohlávek. Concept lattices and order in fuzzy logic. *Annals of Pure and Applied Logic* (to appear, 22 pp.).
5. Bělohlávek R.: Getting maximal rectangular submatrices from [0,1]-valued object-attribute tables: algorithms for fuzzy concept lattices (submitted). Preliminary version appeared in *Proc. Fourth Int. Conf. on Recent Advances in Soft Computing.* Nottingham, United Kingdom, 12-13 December, 2002, pp. 200–205.
6. R Bělohlávek, J. Dvořák, J. Outrata. Fast factorization of concept-clusters by similarity (in preparation).
7. A. Burusco, R. Fuentes-Gonzáles. The study of the L-fuzzy concept lattice. *Mathware & Soft Computing*, 3:209–218, 1994.
8. B. Ganter, R. Wille. *Formal Concept Analysis. Mathematical Foundations.* Springer, Berlin, 1999.
9. D. S. Johnson, M. Yannakakis, C. H. Papadimitrou. On generating all maximal independent sets. *Inf. Processing Letters* 15:129–133, 1988.
10. G. .J. Klir, B. Yuan. *Fuzzy Sets and Fuzzy Logic. Theory and Applications.* Prentice Hall, Upper Saddle River, NJ, 1995.
11. S. Pollandt. *Fuzzy Begriffe.* Springer, Berlin, 1997.