Direct Factorization by Similarity of Fuzzy Concept Lattices by Factorization of Input Data*

Radim Belohlavek[1,2], Jan Outrata[2], and Vilem Vychodil[1,2]

[1] Dept. Systems Science and Industrial Engineering
T. J. Watson School of Engineering and Applied Science
Binghamton University–SUNY, PO Box 6000, Binghamton, NY 13902–6000, USA
{rbelohla,vychodil}@binghamton.edu
[2] Dept. Computer Science, Palacky University, Olomouc
Tomkova 40, CZ-779 00 Olomouc, Czech Republic
jan.outrata@upol.cz

Abstract. The paper presents additional results on factorization by similarity of fuzzy concept lattices. A fuzzy concept lattice is a hierarchically ordered collection of clusters extracted from tabular data. The basic idea of factorization by similarity is to have, instead of a possibly large original fuzzy concept lattice, its factor lattice. The factor lattice contains less clusters than the original concept lattice but, at the same time, represents a reasonable approximation of the original concept lattice and provides us with a granular view on the original concept lattice. The factor lattice results by factorization of the original fuzzy concept lattice by a similarity relation. The similarity relation is specified by a user by means of a single parameter, called a similarity threshold. Smaller similarity thresholds lead to smaller factor lattices, i.e. to more comprehensible but less accurate approximations of the original concept lattice. Therefore, factorization by similarity provides a trade-off between comprehensibility and precision. We first recall the notion of factorization. Second, we present a way to compute the factor lattice of a fuzzy concept lattice directly from input data, i.e. without the need to compute the possibly large original concept lattice.

1 Introduction and Motivation

Formal concept analysis (FCA) is a method of exploratory data analysis which aims at extracting a hierarchical structure of clusters from tabular data describing objects and their attributes. The history of FCA goes back to Wille's paper [19], foundations, algorithms, and a survey of applications can be found in [11,12].

The clusters $\langle A, B\rangle$, called formal concepts, consist of a collection A (concept extent) of objects and a collection B (concept intent) of attributes which

* Supported by grant No. 1ET101370417 of GA AV ČR, by grant No. 201/05/0079 of the Czech Science Foundation, and by institutional support, research plan MSM 6198959214.

S. Ben Yahia et al. (Eds.): CLA 2006, LNAI 4923, pp. 68–79, 2008.

are maximal with respect to the property that each object from A has every attribute from B. The extent-intent definition of formal concepts goes back to traditional Port-Royal logic. Alternatively, formal concepts can be thought of as maximal rectangles contained in object-attribute data table. Formal concepts can be partially ordered by a natural subconcept-superconcept relation (narrower clusters are under larger ones). The resulting partially ordered set of concepts forms a complete lattice, called a concept lattice, and can be visualized by a labelled Hasse diagram. In the basic setting, the attributes are binary, i.e. each table entry contains either 0 or 1. FCA was extended to data tables with fuzzy attributes, i.e. tables with entries containing degrees to which a particular attribute applies to a particular object, see e.g. [4,5,18].

A direct user comprehension and interpretation of the partially ordered set of clusters may be difficult due to a possibly large number of clusters extracted from a data table. A way to go is to consider, instead of the whole concept lattice, its suitable factor lattice which can be considered a granular version of the original concept lattice: its elements are classes of clusters and the factor lattice is smaller. A method of factorization by a so-called compatible reflexive and symmetric relation (a tolerance) on the set of clusters was described in [12]. Interpreting the tolerance relation as similarity on clusters/concepts, the elements of the factor lattice are classes of pairwise similar clusters. The specification of the tolerance relation is, however, left to the user. In [2], a method of parameterized factorization of concept lattices computed from data with fuzzy attributes was presented: the tolerance relation is induced by a threshold (parameter of factorization) specified by a user. Using a suitable measure of similarity degree of clusters/concepts (see later), the method does the following. Given a threshold a (e.g. a number from $[0, 1]$), the elements of the factor lattice are similarity blocks determined by a, i.e. maximal collections of formal concepts which are pairwise similar to degree at least a. The smaller a, the smaller the factor lattice, i.e. the larger the reduction. For a user, the factor lattice provides a granular view on the original concept lattice, where the granules are the similarity blocks.

In order to compute the factor lattice directly by definition, we have to compute the whole concept lattice (this can be done by an algorithm with a polynomial time delay, see [3]) and then compute all the similarity blocks, i.e. elements of the factor lattice (again, this can be accomplished by an algorithm with polynomial time delay).

In this paper, we present a way to compute the factor lattice directly from data. The resulting algorithm is significantly faster than computing first the whole concept lattice and then computing the similarity blocks. In addition to that, the smaller the similarity threshold, the faster the computation of the factor lattice. This feature corresponds to a rule "the more tolerance to imprecision, the faster the result" which is characteristic for human categorization. The method presented can be seen as an alternative to a method of fast factorization of concept lattices by similarity presented in [6].

The paper is organized as follows. Section 2 presents preliminaries on fuzzy sets and formal concept analysis of data with fuzzy attributes. In Section 3, we present the main results. Examples and experiments demonstrating the speed-up are contained in Section 4. Section 5 presents a summary and an outline of a future research.

2 Preliminaries

2.1 Fuzzy Sets and Fuzzy Logic

In this section, we recall necessary notions from fuzzy sets and fuzzy logic. We refer to [4,14,16] for further details. The concept of a fuzzy set generalizes that of an ordinary set in that an element may belong to a fuzzy set in an intermediate truth degree not necessarily being 0 or 1. As a structure of truth degrees, equipped with operations for logical connectives, we use complete residuated lattices, i.e. structures $\mathbf{L} = \langle L, \wedge, \vee, \otimes, \rightarrow, 0, 1\rangle$, where $\langle L, \wedge, \vee, 0, 1\rangle$ is a complete lattice with 0 and 1 being the least and greatest element of L, respectively; $\langle L, \otimes, 1\rangle$ is a commutative monoid (i.e. $\otimes$ is commutative, associative, and $a \otimes 1 = 1 \otimes a = a$ for each $a \in L$); and $\otimes$ and $\rightarrow$ satisfy so-called adjointness property, i.e. $a \otimes b \leq c$ iff $a \leq b \rightarrow c$ for each $a, b, c \in L$. Elements a of L are called truth degrees, $\otimes$ and $\rightarrow$ are (truth functions of) "fuzzy conjunction" and "fuzzy implication".

The most commonly used set L of truth degrees is the real interval $[0, 1]$; with $a \wedge b = \min(a, b)$, $a \vee b = \max(a, b)$. The three most important pairs of "fuzzy conjunction" and "fuzzy implication" are: Łukasiewicz, with $a \otimes b = \max(a + b - 1, 0)$, $a \rightarrow b = \min(1 - a + b, 1)$; minimum, with $a \otimes b = \min(a, b)$, $a \rightarrow b = 1$ if $a \leq b$ and $= b$ else; and product, with $a \otimes b = a \cdot b$, $a \rightarrow b = 1$ if $a \leq b$ and $= b/a$ else. Often, we need a finite chain $\{a_0 = 0, a_1, \ldots, a_n = 1\}$ ($a_0 < \cdots < a_n$); with corresponding Łukasiewicz ($a_k \otimes a_l = a_{\max(k+l-n,0)}$, $a_k \rightarrow a_l = a_{\min(n-k+l,n)}$) or minimum ($a_k \otimes a_l = a_{\min(k,l)}$, $a_k \rightarrow a_l = a_n$ for $a_k \leq a_l$ and $a_k \rightarrow a_l = a_l$ otherwise) connectives. Note that complete residuated lattices are basic structures of truth degrees used in fuzzy logic, see [13,14]. Residuated lattices cover many particular structures, i.e. sets of truth degrees and fuzzy logical connectives, used in applications of fuzzy logic.

A fuzzy set A in a universe set U is a mapping $A : U \rightarrow L$ with $A(u)$ being interpreted as a degree to which u belongs to A. To make $\mathbf{L}$ explicit, fuzzy sets are also called $\mathbf{L}$-sets. By $\mathbf{L}^U$ or L^U we denote the set of all fuzzy sets in universe U, i.e. $L^U = \{A \mid A \text{ is a mapping of } U \text{ to } L\}$. If $U = \{u_1, \ldots, u_n\}$ then A is denoted by $A = \{{}^{a_1}/u_1, \ldots, {}^{a_n}/u_n\}$ meaning that $A(u_i)$ equals a_i. For brevity, we omit elements of U whose membership degree is zero. A binary fuzzy relation I between sets X and Y is a fuzzy set in universe $U = X \times Y$, i.e. a mapping $I : X \times Y \rightarrow L$ assigning to each $x \in X$ and $y \in Y$ a degree $I(x, y)$ to which x is related to y.

For $A \in L^U$ and $a \in L$, a set ${}^aA = \{u \in U \mid A(u) \geq a\}$ is called an a-cut of A (the ordinary set of elements from U which belong to A to degree at least a); a fuzzy set $a \rightarrow A$ in U defined by $(a \rightarrow A)(u) = a \rightarrow A(u)$ is called an a-shift of A; a fuzzy set $a \otimes A$ in U defined by $(a \otimes A)(u) = a \otimes A(u)$ is called an a-multiple of A.

Given $A, B \in \mathbf{L}^U$, we define a subsethood degree $S(A,B) = \bigwedge_{u \in U}\big(A(u) \rightarrow B(u)\big)$, which generalizes the classical subsethood relation $\subseteq$. $S(A,B)$ represents a degree to which A is a subset of B. In particular, we write $A \subseteq B$ iff $S(A,B) = 1$ (A is fully contained in B). As a consequence, $A \subseteq B$ iff $A(u) \leq B(u)$ for each $u \in U$.

2.2 Fuzzy Concept Lattices

A data table with fuzzy attributes can be identified with a triplet $\langle X, Y, I\rangle$ where X is a non-empty set of objects (table rows), Y is a non-empty set of attributes (table columns), and I is a (binary) fuzzy relation between X and Y, i.e. $I : X \times Y \rightarrow L$. In formal concept analysis, the triplet $\langle X, Y, I\rangle$ is called a formal fuzzy context. For $x \in X$ and $y \in Y$, a degree $I(x, y) \in L$ is interpreted as a degree to which object x has attribute y (table entry corresponding to row x and column y). For $L = \{0, 1\}$, formal fuzzy contexts can be identified in an obvious way with ordinary formal contexts.

For fuzzy sets $A \in L^X$ and $B \in L^Y$ we define fuzzy sets $A^{\Uparrow} \in L^Y$ and $B^{\Downarrow} \in L^X$ (denoted also $A^{\Uparrow_I}$ and $B^{\Downarrow_I}$ to make I explicit) by

$$A^{\Uparrow}(y) = \textstyle\bigwedge_{x \in X}(A(x) \rightarrow I(x, y)), \quad (1)$$

$$B^{\Downarrow}(x) = \textstyle\bigwedge_{y \in Y}(B(y) \rightarrow I(x, y)). \quad (2)$$

Using basic rules of predicate fuzzy logic one can see that $A^{\Uparrow}$ is a fuzzy set of all attributes common to all objects from A, and $B^{\Downarrow}$ is a fuzzy set of all objects sharing all attributes from B. The set

$$\mathcal{B}(X, Y, I) = \{\langle A, B\rangle \mid A^{\Uparrow} = B,\ B^{\Downarrow} = A\}$$

of all fixpoints of $\langle {}^{\Uparrow}, {}^{\Downarrow}\rangle$ is called a fuzzy concept lattice associated to $\langle X, Y, I\rangle$; elements $\langle A, B\rangle \in \mathcal{B}(X, Y, I)$ are called formal concepts of $\langle X, Y, I\rangle$; A and B are called the extent and intent of $\langle A, B\rangle$, respectively. Under a partial order $\leq$ defined on $\mathcal{B}(X, Y, I)$ by

$$\langle A_1, B_1\rangle \leq \langle A_2, B_2\rangle \ \text{ iff } \ A_1 \subseteq A_2,$$

$\mathcal{B}(X, Y, I)$ happens to be a complete lattice. The following theorem, so-called *main theorem of fuzzy concept lattices*, describes the structure of $\mathcal{B}(X, Y, I)$, see [4] for details.

Theorem 1. *$\mathcal{B}(X, Y, I)$ is under $\leq$ a complete lattice where the infima and suprema are given by*

$$\textstyle\bigwedge_{j \in J}\langle A_j, B_j\rangle = \Big\langle \bigcap_{j \in J} A_j, (\bigcup_{j \in J} B_j)^{\Downarrow\Uparrow}\Big\rangle, \quad (3)$$

$$\textstyle\bigvee_{j \in J}\langle A_j, B_j\rangle = \Big\langle (\bigcup_{j \in J} A_j)^{\Uparrow\Downarrow}, \bigcap_{j \in J} B_j\Big\rangle. \quad (4)$$

Moreover, an arbitrary complete lattice $\mathbf{K} = \langle K, \leq\rangle$ is isomorphic to some $\mathcal{B}(X, Y, I)$ iff there are mappings $\gamma : X \times L \rightarrow K$, $\mu : Y \times L \rightarrow K$ such that

(i) $\gamma(X \times L)$ is $\bigwedge$-dense in K, $\mu(Y \times L)$ is $\bigvee$-dense in K and
(ii) $\gamma(x, a) \leq \mu(y, b)$ iff $a \otimes b \leq I(x, y)$.

3 Factorization of $\mathcal{B}(X, Y, I)$ by Similarity

3.1 The Notion of Factorization of Fuzzy Concept Lattice by Similarity

We need to recall the parameterized method of factorization introduced in [2]. Given $\langle X, Y, I\rangle$, introduce a binary fuzzy relation $\approx_{\mathrm{Ext}}$ on $\mathcal{B}(X, Y, I)$ by

$$(\langle A_1, B_1\rangle \approx_{\mathrm{Ext}} \langle A_2, B_2\rangle) \;=\; \bigwedge\nolimits_{x\in X}(A_1(x) \leftrightarrow A_2(x)) \tag{5}$$

for $\langle A_i, B_i\rangle \in \mathcal{B}(X, Y, I)$, $i = 1, 2$. Here, $\leftrightarrow$ is a so-called biresiduum (i.e., a truth function of equivalence connective) defined by

$$a \leftrightarrow b = (a \to b) \wedge (b \to a).$$

$(\langle A_1, B_1\rangle \approx_{\mathrm{Ext}} \langle A_2, B_2\rangle)$, called a degree of similarity of $\langle A_1, B_1\rangle$ and $\langle A_2, B_2\rangle$, is just the truth degree of "for each object x: x is covered by A_1 iff x is covered by A_2". One can also consider a fuzzy relation $\approx_{\mathrm{Int}}$ defined by

$$(\langle A_1, B_1\rangle \approx_{\mathrm{Int}} \langle A_2, B_2\rangle) \;=\; \bigwedge\nolimits_{y\in Y}(B_1(y) \leftrightarrow B_2(y)). \tag{6}$$

It can be shown [4] that measuring similarity of formal concepts via intents B_i coincides with measuring similarity via extents A_i, i.e. $\approx_{\mathrm{Ext}}$ coincides with $\approx_{\mathrm{Int}}$, corresponding naturally to the duality of extent/intent view. As a result, we write also just $\approx$ instead of $\approx_{\mathrm{Ext}}$ and $\approx_{\mathrm{Int}}$. Note also that $\approx$ is a fuzzy equivalence relation on $\mathcal{B}(X, Y, I)$.

Given a truth degree $a \in L$ (a similarity threshold specified by a user), consider the thresholded relation ${}^a\!\approx$ on $\mathcal{B}(X, Y, I)$ defined by

$$\langle\langle A_1, B_1\rangle, \langle A_2, B_2\rangle\rangle \in {}^a\!\approx \quad \text{iff} \quad (\langle A_1, B_1\rangle \approx \langle A_2, B_2\rangle) \geq a.$$

That is, ${}^a\!\approx$ is an ordinary relation "being similar to degree at least a" and we thereby call it simply similarity (relation). ${}^a\!\approx$ is a reflexive and symmetric binary relation (i.e., a tolerance relation) on $\mathcal{B}(X, Y, I)$. However, ${}^a\!\approx$ need not be transitive (it is transitive if, for instance, $a \otimes b = a \wedge b$ holds true in $\mathbf{L}$). ${}^a\!\approx$ is said to be compatible if it is preserved under arbitrary suprema and infima in $\mathcal{B}(X, Y, I)$, i.e. if $\langle c_j, c_j'\rangle \in {}^a\!\approx$ for $j \in J$ implies both $\langle \bigwedge_{j\in J} c_j, \bigwedge_{j\in J} c_j'\rangle \in {}^a\!\approx$ and $\langle \bigvee_{j\in J} c_j, \bigvee_{j\in J} c_j'\rangle \in {}^a\!\approx$ for any $c_j, c_j' \in \mathcal{B}(X, Y, I)$, $j \in J$. We call $\approx$ compatible if ${}^a\!\approx$ is compatible for each $a \in L$.

Call a subset B of $\mathcal{B}(X, Y, I)$ an ${}^a\!\approx$-block if it is a maximal subset of $\mathcal{B}(X, Y, I)$ such that any two formal concepts from B are similar to degree at least a, i.e., for any $c_1, c_2 \in B$ we have $\langle c_1, c_2\rangle \in {}^a\!\approx$. Note that the notion of an ${}^a\!\approx$-block generalizes that of an equivalence class: if ${}^a\!\approx$ is an equivalence relation then ${}^a\!\approx$-blocks are exactly the equivalence classes of ${}^a\!\approx$. Denote by $\mathcal{B}(X, Y, I)/{}^a\!\approx$ the collection of all ${}^a\!\approx$-blocks. It follows from the results on tolerances on complete lattices [12] that if ${}^a\!\approx$ is compatible, then ${}^a\!\approx$-blocks are special intervals in the concept lattice $\mathcal{B}(X, Y, I)$. For a formal concept

$\langle A, B\rangle \in \mathcal{B}(X, Y, I)$, denote by $\langle A, B\rangle_a$ and $\langle A, B\rangle^a$ the infimum and the supremum of the set of all formal concepts which are similar to $\langle A, B\rangle$ to degree at least a, that is,

$$\langle A, B\rangle_a = \bigwedge\{\langle A', B'\rangle \mid \langle\langle A, B\rangle, \langle A', B'\rangle\rangle \in {}^a{\approx}\}, \quad (7)$$

$$\langle A, B\rangle^a = \bigvee\{\langle A', B'\rangle \mid \langle\langle A, B\rangle, \langle A', B'\rangle\rangle \in {}^a{\approx}\}. \quad (8)$$

Operators $\ldots_a$ and $\ldots^a$ are important in description of ${}^a{\approx}$-blocks [12]:

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.,*

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

Note that an interval with lower bound $\langle A_1, B_1\rangle$ and upper bound $\langle A_2, B_2\rangle$ is the subset $[\langle A_1, B_1\rangle, \langle A_2, B_2\rangle] = \{\langle A, B\rangle \in \mathcal{B}(X, Y, I) \mid \langle A_1, B_1\rangle \leq \langle A, B\rangle \leq \langle A_2, B_2\rangle\}$.

Now, define a partial order $\preceq$ on blocks of $\mathcal{B}(X, Y, I)/{}^a{\approx}$ by

$$[c_1, c_2] \preceq [d_1, d_2] \quad \text{iff} \quad c_1 \leq d_1 \quad (\text{iff} \quad c_2 \leq d_2) \quad (9)$$

for any $[c_1, c_2], [d_1, d_2] \in \mathcal{B}(X, Y, I)/{}^a{\approx}$. Then we have [2]:

Theorem 2. *$\mathcal{B}(X, Y, I)/{}^a{\approx}$ 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 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 further details and properties of $\mathcal{B}(X, Y, I)/{}^a{\approx}$ we refer to [2].

3.2 Similarity-Based Factorization of Input Data $\langle X, Y, I\rangle$ and Direct Computing of the Factor Lattice $\mathcal{B}(X, Y, I)/{}^a{\approx}$

We now turn our attention to the problem of how to compute the factor lattice. One way is to follow the definition and to split the computation of $\mathcal{B}(X, Y, I)/{}^a{\approx}$ into two steps: (1) compute the possibly large fuzzy concept lattice $\mathcal{B}(X, Y, I)$ and (2) compute the ${}^a{\approx}$-blocks, i.e. the elements of $\mathcal{B}(X, Y, I)/{}^a{\approx}$. Although there are efficient algorithms for both (1) and (2), computing $\mathcal{B}(X, Y, I)/{}^a{\approx}$ this way is time demanding. In what follows, we present a way to obtain $\mathcal{B}(X, Y, I)/{}^a{\approx}$ directly, without the need to compute $\mathcal{B}(X, Y, I)$ first and then to compute the blocks of ${}^a{\approx}$. We need the following lemmas.

Lemma 2 ([6]). *For $\langle A, B\rangle \in \mathcal{B}(X, Y, I)$, we have*

(a) $\langle A, B\rangle_a = \left\langle (a \otimes A)^{\Uparrow\Downarrow}, a \rightarrow B\right\rangle$,
(b) $\langle A, B\rangle^a = \left\langle a \rightarrow A, (a \otimes B)^{\Downarrow\Uparrow}\right\rangle$.

Lemma 3. *If A is an extent then we have $a \rightarrow A = (a \rightarrow A)^{\Uparrow\Downarrow}$; similarly for an intent B.*

Proof. Follows from Lemma 2, cf. [4].

Remark 1. Thus we have $(\langle A, B\rangle_a)^a = \left\langle a \rightarrow (a \otimes A)^{\Uparrow\Downarrow}, (a \otimes (a \rightarrow B))^{\Downarrow\Uparrow}\right\rangle$.

Let us now introduce the construction of a similarity-based factorization assigning to $\langle X, Y, I\rangle$ a "factorized data" $\langle X, Y, I\rangle/a$. For a formal fuzzy context $\langle X, Y, I\rangle$ and a (user-specified) threshold $a \in L$, introduce a formal fuzzy context $\langle X, Y, I\rangle/a$ by

$$\langle X, Y, I\rangle/a \ := \ \langle X, Y, a \rightarrow I\rangle.$$

$\langle X, Y, I\rangle/a$ will be called the factorized context of $\langle X, Y, I\rangle$ by threshold a. That is, $\langle X, Y, I\rangle/a$ has the same objects and attributes as $\langle X, Y, I\rangle$, and the incidence relation of $\langle X, Y, I\rangle/a$ is $a \rightarrow I$. Since

$$(a \rightarrow I)(x, y) \ = \ a \rightarrow I(x, y),$$

computing $\langle X, Y, I\rangle/a$ from $\langle X, Y, I\rangle$ is easy. Note that objects and attributes are more similar in $\langle X, Y, I\rangle/a$ than in the original context $\langle X, Y, I\rangle$. Indeed, for any $x_1, x_2 \in X$ and $y_1, y_2 \in Y$ one can easily verify that

$$I(x_1, y_1) \leftrightarrow I(x_2, y_2) \leq (a \rightarrow I)(x_1, y_1) \leftrightarrow (a \rightarrow I)(x_2, y_2)$$

which intuitively says that in the factorized context, the table entries are more similar (closer) than in the original one.

A way to obtain the factor lattice $\mathcal{B}(X, Y, I)/^a\!\approx$ directly from input data $\langle X, Y, I\rangle$ is based on the next theorem.

Theorem 3. *For a formal fuzzy context $\langle X, Y, I\rangle$ and a threshold $a \in L$ we have*

$$\mathcal{B}(X, Y, I)/^a\!\approx \ \cong \ \mathcal{B}(X, Y, a \rightarrow I).$$

In words, $\mathcal{B}(X, Y, I)/^a\!\approx$ is isomorphic to $\mathcal{B}(X, Y, a \rightarrow I)$. Moreover, under the isomorphism, $[\langle A_1, B_1\rangle, \langle A_2, B_2\rangle] \in \mathcal{B}(X, Y, I)/^a\!\approx$ corresponds to $\langle A_2, B_1\rangle \in \mathcal{B}(X, Y, a \rightarrow I)$.

Proof. Let $^\Uparrow$ and $^\Downarrow$ denote the operators (1) and (2) induced by I and $^{\Uparrow_a}$ and $^{\Downarrow_a}$ denote the operators induced by $a \rightarrow I$, that is, for $A \in L^X$ and $B \in L^Y$ we have

$$A^{\Uparrow_a}(y) = \bigwedge_{x \in X} A(x) \rightarrow (a \rightarrow I)(x, y),$$
$$B^{\Downarrow_a}(y) = \bigwedge_{y \in Y} B(y) \rightarrow (a \rightarrow I)(x, y).$$

Take any $A \in L^X$. Then we have

$$A^{\Uparrow_a}(y) = \bigwedge_{x\in X} A(x) \to (a \to I(x,y)) =$$
$$= \bigwedge_{x\in X} a \to (A(x) \to I(x,y)) =$$
$$= a \to \bigwedge_{x\in X} (A(x) \to I(x,y)) = a \to A^{\Uparrow}(x),$$

and

$$A^{\Uparrow_a\Downarrow_a}(x) = \bigwedge_{y\in Y} A^{\Uparrow_a}(y) \to (a \to I(x,y)) =$$
$$= \bigwedge_{y\in Y} a \to (A^{\Uparrow_a}(y) \to I(x,y)) = a \to \bigwedge_{y\in Y} (A^{\Uparrow_a}(y) \to I(x,y)) =$$
$$= a \to \bigwedge_{y\in Y} ([\bigwedge_{x\in X} a \to (A(x) \to I(x,y))] \to I(x,y)) =$$
$$= a \to \bigwedge_{y\in Y} ([\bigwedge_{x\in X} (a \otimes A(x)) \to I(x,y)] \to I(x,y)) =$$
$$= a \to \bigwedge_{y\in Y} ((a \otimes A)^{\Uparrow}(x) \to I(x,y)) = a \to (a \otimes A)^{\Uparrow\Downarrow}(x),$$

i.e.

$$A^{\Uparrow_a} = a \to A^{\Uparrow} \quad \text{and} \quad A^{\Uparrow_a\Downarrow_a} = a \to (a \otimes A)^{\Uparrow\Downarrow}. \tag{10}$$

Now, let $[\langle A_1, B_1\rangle, \langle A_2, B_2\rangle] \in \mathcal{B}(X,Y,I)/{}^a{\approx}$. By Lemmas 1, 2 and 3, there is $\langle A, B\rangle \in \mathcal{B}(X,Y,I)$ such that $\langle A_1, B_1\rangle = \langle A, B\rangle_a = \langle (a \otimes A)^{\Uparrow\Downarrow}, a \to B\rangle$ and $\langle A_2, B_2\rangle = (\langle A, B\rangle_a)^a = \langle a \to (a \otimes A)^{\Uparrow\Downarrow}, (a \otimes (a \to B))^{\Downarrow\Uparrow}\rangle$. Since $\langle A, B\rangle = \langle A, A^{\Uparrow}\rangle$, (10) yields

$$A_2 = a \to (a \otimes A)^{\Uparrow\Downarrow} = A^{\Uparrow_a\Downarrow_a}$$

and

$$B_1 = a \to B = a \to A^{\Uparrow} = A^{\Uparrow_a}.$$

This shows $\langle A_2, B_1\rangle \in \mathcal{B}(X,Y,a \to I)$.

Conversely, if $\langle A_2, B_1\rangle \in \mathcal{B}(X,Y,a \to I)$ then using (10), $B_1 = A_2^{\Uparrow_a} = a \to A_2^{\Uparrow}$ and $A_2 = A_2^{\Uparrow_a\Downarrow_a} = a \to (a \otimes A_2)^{\Uparrow\Downarrow}$. By Lemma 1 and Lemma 2, $[\langle B_1^{\Downarrow}, B_1\rangle, \langle A_2, A_2^{\Uparrow}\rangle] \in \mathcal{B}(X,Y,I)/{}^a{\approx}$. The proof is complete.

Remark 2. (1) The blocks of $\mathcal{B}(X,Y,I)/{}^a{\approx}$ can be reconstructed from the formal concepts of $\mathcal{B}(X,Y,a \to I)$:
If $\langle A, B\rangle \in \mathcal{B}(X,Y,a \to I)$ then $[\langle B^{\Downarrow}, B\rangle, \langle A, A^{\Uparrow}\rangle] \in \mathcal{B}(X,Y,I)/{}^a{\approx}$.

(2) Computing $\mathcal{B}(X, Y, a \to I)$ means computing of the ordinary fuzzy concept lattice. This can be done by an algorithm of polynomial time delay complexity, see [3].

This shows a way to obtain $\mathcal{B}(X, Y, I)/^a\!\approx$ without computing first the whole $\mathcal{B}(X, Y, I)$ and then computing the factorization. Note that in [6], we showed an alternative way to speed up the computation of $\mathcal{B}(X, Y, I)/^a\!\approx$ by showing that suprema of blocks of $\mathcal{B}(X, Y, I)/^a\!\approx$ are fixed points of a certain fuzzy closure operator. Compared to that, the present approach shows that the blocks of $\mathcal{B}(X, Y, I)/^a\!\approx$ can be interpreted as formal concepts in a "factorized context" $\langle X, Y, I\rangle/a$.

4 Examples and Experiments

In this section we demonstrate the effect of reduction of size of a fuzzy concept lattice by factorization by similarity, and the speed-up achieved by our algorithm based on Theorem 3. By reduction of size of a fuzzy concept lattice given by a data table $\langle X, Y, I\rangle$ with fuzzy attributes and a user-specified threshold a, we mean the ratio

$$\frac{|\mathcal{B}(X, Y, I)/^a\!\approx|}{|\mathcal{B}(X, Y, I)|}$$

of the number $|\mathcal{B}(X, Y, I)/^a\!\approx|$ of elements of $\mathcal{B}(X, Y, I)/^a\!\approx$, i.e. the number of elements of the factor lattice, to the number $|\mathcal{B}(X, Y, I)|$ of elements of $\mathcal{B}(X, Y, I)$,

Table 1. Data table with fuzzy attributes

	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 Germany	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 – High Gross Domestic Product per capita (USD), 2 – High Consumer Price Index (1995=100) , 3 – High Unemployment Rate (percent - ILO), 4 – High production of electricity per capita (kWh), 5 – High energy consumption per capita (GJ), 6 – High export per capita (USD), 7 – High import per capita (USD)

Table 2. Łukasiewicz fuzzy logical connectives, $\mathcal{B}(X,Y,I)$ of data from Tab. 1: $|\mathcal{B}(X,Y,I)| = 774$, 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
size reduction	0.010	0.073	0.249	0.546
naive algorithm (ms)	8995	9463	8573	9646
our algorithm (ms)	23	214	383	1517
speed-up	391.09	44.22	22.38	6.36

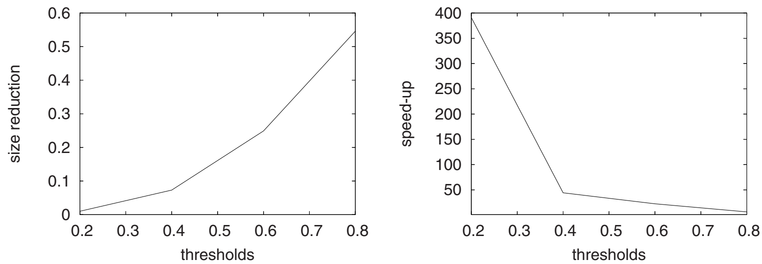

Fig. 1. Size reduction and speed-up from Tab. 2

Table 3. Minimum-based fuzzy logical connectives, $\mathcal{B}(X,Y,I)$ of data from Tab. 1: $|\mathcal{B}(X,Y,I)| = 304$, 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
size reduction	0.026	0.210	0.638	1.000
naive algorithm (ms)	1830	1634	3787	4440
our algorithm (ms)	23	106	431	1568
speed-up	79.57	15.42	8.79	2.83

i.e. the number of elements of the original lattice. By a speed-up we mean the ratio of the time for computing the factor lattice $\mathcal{B}(X,Y,I)/^{a}\!\approx$ by a naive algorithm to the time for computing $\mathcal{B}(X,Y,I)/^{a}\!\approx$ by our algorithm. By "our algorithm" we mean the algorithm computing $\mathcal{B}(X,Y,I)/^{a}\!\approx$ directly by reduction to the computation of $\mathcal{B}(\langle X,Y,I\rangle/a)$, described in subsection 3.2. By "naive algorithm" we mean computing $\mathcal{B}(X,Y,I)/^{a}\!\approx$ by first generating $\mathcal{B}(X,Y,I)$ (by a polynomial time-delay algorithm from [3]) and subsequently generating the $^{a}\!\approx$-blocks by producing $[\langle A,B\rangle_a, (\langle A,B\rangle_a)^a]$.

Consider the data table depicted in Tab. 1. The data table contains countries (objects from X) and some of their economic characteristics (attributes from Y). The values of the characteristics are scaled to interval $[0,1]$ so that the characteristics can be considered as fuzzy attributes.

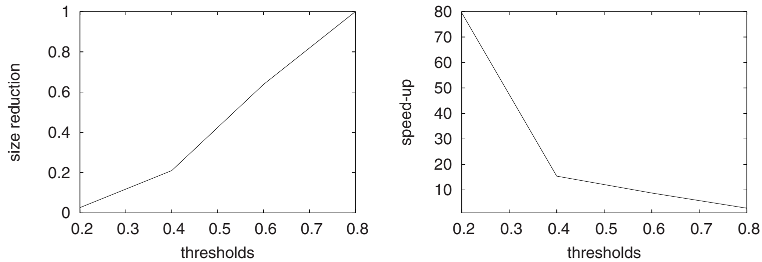

Fig. 2. Size reduction and speed-up from Tab. 3

Tab. 2 summarizes the results when using Łukasiewicz fuzzy logical operations and threshold values $a = 0.2, 0.4, 0.6, 0.8$. 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 [3] takes 2292ms.

The example demonstrates that smaller thresholds lead to both larger size reduction and speed-up. 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 $\mathcal{B}(X, Y, I)$.

Note also that since computing $\mathcal{B}(X, Y, I)$ takes 2292 ms, most of the time consumed by the naive algorithm is spent on factorization. For instance, for $a = 0.2$, 8995 ms is consumed in total of which 2292 ms is spent on computing $\mathcal{B}(X, Y, I)$ and $6703 = 8995 - 2292$ ms is spent on factorization, i.e. on computing $\mathcal{B}(X, Y, I)/^a\!\approx$ from $\mathcal{B}(X, Y, I)$.

Fig. 1 contains graphs depicting reduction $|\mathcal{B}(X, Y, I)/^a\!\approx|/|\mathcal{B}(X, Y, I)|$ and speed-up from Tab. 2.

Tab. 3 and Fig. 2 show the same characteristics when using the minimum-based fuzzy logical operations (instead of Łukasiewicz fuzzy logical operations).

5 Conclusions and Future Research

We presented an additional method of factorization of fuzzy concept lattices. A factor lattice represents an approximate version of the original fuzzy concept lattice. The size of the factor lattice is controlled by a user-specified threshold. The factor lattice can be computed directly from input data, without first computing the possibly large original fuzzy concept lattice.

Our future research will focus on factorization of further types of fuzzy concept lattices. In particular, [7] presents a method of fast factorization of fuzzy concept lattices with hedges, see [8], which can be seen as a generalization of the method from [6]. Fuzzy concept lattices with hedges serve as a common platform for some of the types of fuzzy concept lattices, see [9], and also [10,17]. An immediate problem is whether and to what extent the results presented in this paper can be accommodated for the setting of fuzzy concept lattices with hedges.

References

1. Belohlavek, R.: Fuzzy Galois connections. Math. Logic Quarterly 45(4), 497–504 (1999)
2. Belohlavek, R.: Similarity relations in concept lattices. J. Logic and Computation 10(6), 823–845 (2000)
3. Belohlavek, R.: Algorithms for fuzzy concept lattices. In: Proc. RASC 2002, Nottingham, UK, December 12–13, 2002, pp. 200–205 (2002)
4. Belohlavek, R.: Fuzzy Relational Systems: Foundations and Principles. Kluwer, Academic/Plenum Publishers, New York (2002)
5. Belohlavek, R.: Concept lattices and order in fuzzy logic. Ann. Pure Appl. Logic 128, 277–298 (2004)
6. Belohlavek, R., Dvořák, J., Outrata, J.: Fast factorization by similarity in formal concept analysis of data with fuzzy attributes. Journal of Computer and System Sciences 73(6), 1012–1022 (2007)
7. Belohlavek, R., Outrata, J., Vychodil, V.: On factorization by similarity of fuzzy concepts with hedges. Int. J. of Foundations of Computer Science (to appear)
8. Belohlavek, R., Vychodil, V.: Reducing the size of fuzzy concept lattices by hedges. In: FUZZ-IEEE 2005, The IEEE International Conference on Fuzzy Systems, Reno (Nevada, USA), May 22–25, 2005, pp. 663–668 (2005)
9. Belohlavek, R., Vychodil, V.: What is a fuzzy concept lattice? In: Proc. CLA 2005, 3rd Int. Conference on Concept Lattices and Their Applications, Olomouc, Czech Republic, September 7–9, 2005, pp. 34–45., http://ceur-ws.org/Vol-162/
10. Ben Yahia, S., Jaoua, A.: Discovering knowledge from fuzzy concept lattice. In: Kandel, A., Last, M., Bunke, H. (eds.) Data Mining and Computational Intelligence, pp. 167–190. Physica-Verlag, Heidelberg New York (2001)
11. Carpineto, C., Romano, G.: Concept Data Analysis. Theory and Applications. J. Wiley, Chichester (2004)
12. Ganter, B., Wille, R.: Formal Concept Analysis. Mathematical Foundations. Springer, Berlin (1999)
13. Goguen, J.A.: The logic of inexact concepts. Synthese 18(9), 325–373 (1968)
14. Hájek, P.: Metamathematics of Fuzzy Logic. Kluwer, Dordrecht (1998)
15. Hájek, P.: On very true. Fuzzy Sets and Systems 124, 329–333 (2001)
16. Klir, G.J., Yuan, B.: Fuzzy Sets and Fuzzy Logic. Theory and Applications. Prentice-Hall, Englewood Cliffs (1995)
17. Krajči, S.: Cluster based efficient generation of fuzzy concepts. Neural Network World 5, 521–530 (2003)
18. Pollandt, S.: Fuzzy Begriffe. Springer, Berlin/Heidelberg (1997)
19. Wille, R.: Restructuring lattice theory: an approach based on hierarchies of concepts. In: Rival, I. (ed.) Ordered Sets, Reidel, Dordrecht, Boston, pp. 445–470 (1982)