

Formal concept analysis over attributes with levels of granularity

Radim Bělohlávek, Vladimír Sklenář

Dept. Computer Science, Palacký University, Tomkova 40, CZ-779 00, Olomouc, Czech Republic

Email: {radim.belohlavek, vladimir.sklenar}@upol.cz

Abstract—Formal concept analysis (FCA) is a method of exploratory analysis of object-attribute data tables. The two main outputs are a hierarchical structure of clusters (so-called formal concepts) and a non-redundant basis of so-called attribute implications. An important topic in FCA is to cope with a possibly large number of resulting clusters. We propose a method to control the number of clusters by means of specification of a granularity level of attributes. A user selects an appropriate level of granularity of each attribute. If the corresponding set of clusters is too large, the user can select a lower level of granularity for appropriate attributes. The resulting set of clusters is then smaller and can be seen as a rougher version of the original set of clusters. If the corresponding set of clusters is too small, the user can select a finer level of granularity for appropriate attributes. The resulting set of clusters is then larger and can be seen as a refinement of the original set of clusters. The paper presents a preliminary study on this topic. We describe the motivations, the method, basic theoretical insight, and experiments demonstrating the method. Formal concept analysis (FCA) is a method of exploratory analysis of object-attribute data tables. The two main outputs are a hierarchical structure of clusters (so-called formal concepts) and a non-redundant basis of so-called attribute implications. An important topic in FCA is to cope with a possibly large number of resulting clusters. We propose a method to control the number of clusters by means of specification of a granularity level of attributes. A user selects an appropriate level of granularity of each attribute. If the corresponding set of clusters is too large, the user can select a lower level of granularity for appropriate attributes. The resulting set of clusters is then smaller and can be seen as a rougher version of the original set of clusters. If the corresponding set of clusters is too small, the user can select a finer level of granularity for appropriate attributes. The resulting set of clusters is then larger and can be seen as a refinement of the original set of clusters. The paper presents a preliminary study on this topic. We describe the motivations, the method, basic theoretical insight, and experiments demonstrating the method.

I. INTRODUCTION AND PROBLEM SETTING

Formal concept analysis (FCA) [7], [8] is an exploratory method of analysis of tabular data describing objects and their attributes. One of the outputs of FCA is a hierarchical structure of clusters from the input data table. The clusters are called formal concepts and these are pairs $\langle A, B \rangle$ consisting of a collection A of objects and a collection B of attributes. Formal concepts can be partially ordered by a natural subconcept-superconcept relation. The resulting partially ordered set, called a concept lattice, forms a complete lattice and can be visualized by a labeled Hasse diagram. Formal concepts $\langle A, B \rangle$ result from the idea (going back to traditional Port-Royal logic) of a concept as consisting of its extent A and its intent B . Alternatively, formal concepts can be thought of as maximal rectangles contained in the object-attribute data table. In the basic setting, the attributes

are binary presence/absence attributes and the data table is a 0/1-matrix. More general attributes are handled by so-called conceptual scaling (see [8]), i.e. a suitable transformation of a general data table into a 0/1-data table which respects the meaning of attributes. Formal concepts are clusters of data drawn together by having common attributes. FCA has been applied in various fields, among others in software engineering, reengineering problems (redesign of hierarchical structures), text classification (analyzing e-mail collections, classification of library items), browsing retrieval and database views, psychology (study of development of concepts by children), civil engineering (system for checking dependencies in regulations), classification and systematizing of heuristic methods, physiology (color perception), preprocessing of data; see [7], [8] and the references therein.

A problem related to the direct user interpretation of a concept lattice is very often caused the fact that the number of extracted formal concepts is not satisfactory. On the one hand, it may happen that the number of formal concepts is too large. Too large a number of formal concepts provides an overly fine granulation of the input objects and is difficult to interpret for the expert. On the other hand, it may happen that the number of formal concepts is too small. Too small a number of formal concepts does not provide a sufficiently fine granulation of the input objects for the expert.

In this paper, we present a method to control the number of extracted formal concepts by means of selecting levels of granularity of the attributes given in the input data. The paper is organized as follows. Section II presents preliminaries on formal concept analysis. In Section III we present our approach. Section IV contains illustrative examples. Section V presents conclusions and outlines future research.

II. PRELIMINARIES

Formal concept analysis deals with input data in the form of a table with rows corresponding to objects and columns corresponding to attributes which describes a relationship between the objects and attributes. The data table is formally represented by a so-called formal context which is a triplet $\langle X, Y, I \rangle$ where I is a binary relation between X and Y , $\langle x, y \rangle \in I$ meaning that the object x has the attribute y . For each $A \subseteq X$ denote by A^\uparrow a subset of Y defined by

$$A^\uparrow = \{y \mid \text{for each } x \in A : \langle x, y \rangle \in I\}.$$

Similarly, for $B \subseteq Y$ denote by B^\downarrow a subset of X defined by

$$B^\downarrow = \{x \mid \text{for each } y \in B : \langle x, y \rangle \in I\}.$$

That is, A^\uparrow is the set of all attributes from Y shared by all objects from A , and B^\downarrow is the set of all objects from X sharing all attributes from B . A formal concept in $\langle X, Y, I \rangle$ is a pair $\langle A, B \rangle$ of $A \subseteq X$ and $B \subseteq Y$ satisfying $A^\uparrow = B$ and $B^\downarrow = A$. That is, a formal concept consists of a set A of objects which fall under the concept and a set B of attributes which fall under the concept such that A is the set of all objects sharing all attributes from B and, conversely, B is the collection of all attributes from Y shared by all objects from A . This definition formalizes the traditional approach to concepts which is due to Port-Royal logic. The sets A and B are called the extent and the intent of the concept $\langle A, B \rangle$, respectively. The set

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

of all formal concepts in $\langle X, Y, I \rangle$ can be naturally equipped with a partial order \leq defined by $\langle A_1, B_1 \rangle \leq \langle A_2, B_2 \rangle$ iff $A_1 \subseteq A_2$ (or, equivalently, $B_2 \subseteq B_1$). That is, $\langle A_1, B_1 \rangle \leq \langle A_2, B_2 \rangle$ means that each object from A_1 belongs to A_2 (or, equivalently, each attribute from B_2 belongs to B_1). Therefore, \leq models the natural subconcept-superconcept hierarchy under which dog is a subconcept of mammal, etc. The structure of $\mathcal{B}(X, Y, I)$ is described by the so-called main theorem of concept lattices [8]:

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

$$\bigwedge_{j \in J} \langle A_j, B_j \rangle = \langle \bigcap_{j \in J} A_j, (\bigcup_{j \in J} B_j)^{\downarrow\uparrow} \rangle,$$

$$\bigvee_{j \in J} \langle A_j, B_j \rangle = \langle (\bigcup_{j \in J} A_j)^{\uparrow\downarrow}, \bigcap_{j \in J} B_j \rangle.$$

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

- (i) $\gamma(X)$ is \bigvee -dense in \mathbf{V} , $\mu(Y)$ is \bigwedge -dense in \mathbf{V} ;
- (ii) $\gamma(x) \leq \mu(y)$ iff $\langle x, y \rangle \in I$.

III. GRANULARITY OF ATTRIBUTES

In the basic setting of FCA, no further information except for the input data table $\mathcal{T} = \langle X, Y, I \rangle$ is taken into account. In this section, we present a possibility to have, instead of $\langle X, Y, I \rangle$, a more structured input data which allows us to control the granularity of the object attributes. The main gain is that this way we get a means to control the number of formal concepts extracted from the input data.

Granulation is an important phenomenon performed successfully by humans in everyday life. Basically, granulation means considering a collection of pieces of the outer world as a whole—a granule. For example, looking at a person, we may distinguish her head, left arm, right arm, etc. Then, the head, left arm, right arm, etc., are granules for us. The granules might serve as basic units with which our reasoning is concerned. Depending on a particular situation, we might want to use finer or rougher granules, i.e. to increase or decrease a level of granularity. A finer granularity usually leads to a more precise reasoning at the cost of higher computational demands.

	L	R	G
a	×		×
b	×		×
c	×		×
d			×
e			×
f	×	×	
g		×	

	L	R	IG	dG
a	×		×	
b	×		×	
c	×			×
d			×	
e				×
f	×	×		
g		×		

TABLE I

DATA TABLES DESCRIBING OBJECTS (a, ..., g) AND THEIR ATTRIBUTES (L ... LARGE, R ... RED, G ... GREEN, IG ... LIGHT GREEN, dG ... DARK GREEN).

	extent	intent
1	{a, ..., g}	{}
2	{a, b, c, f}	{L}
3	{f, e}	{R}
4	{a, b, c, d, e}	{G}
5	{f}	{L, R}
6	{a, b, c}	{L, G}
7	{}	{L, R, G}

	extent	intent
1	{a, ..., g}	{}
2	{a, b, c, f}	{L}
3	{f, e}	{R}
4	{a, b, d}	{IG}
5	{c, e}	{dG}
6	{f}	{L, R}
7	{a, b}	{L, IG}
8	{c}	{L, dG}
9	{}	{L, R, IG, dG}

TABLE II

LEFT: FORMAL CONCEPTS OF THE LEFT TABLE FROM TAB. I. RIGHT: FORMAL CONCEPTS OF THE RIGHT TABLE FROM TAB. I.

A rougher granularity leads to a less precise reasoning with the benefit of being less computationally demanding. For instance, we increase the level of granularity, if we distinguish nose, ears, eyes, etc., instead of distinguishing only a head. The importance of the phenomenon of granulation and its role in data manipulation and reasoning has been repeatedly emphasized by Zadeh [12].

Granulation and granularity levels naturally appear in tabular data describing objects and their attributes. For instance, consider the data depicted in Tab. I. The tables describe objects a, ..., g. The left table describes the objects by means of their attributes L (large), R (red), and G (green). The right table, however, uses attributes IG (light green) and dG (dark green) instead of a single attribute green. Attributes IG and dG provide a higher level of granularity for the description of the objects than a single attribute G. Namely, attributes IG and dG may be seen as a refinement of G.

The left table of Tab. II shows all the formal concepts extracted from the left table of Tab. I (denote this collection of formal concepts by \mathcal{B}_1), the right table of Tab. II shows all the formal concepts extracted from the right table of Tab. I (denote this collection by \mathcal{B}_2). As one can see from Tab. II, the increase of granularity level represented replacing attribute G by attributes IG and dG leads to an increase in the number of extracted formal concepts. Nevertheless, we can see that there is a natural relationship between \mathcal{B}_1 and \mathcal{B}_2 . Namely, \mathcal{B}_2 can be seen as “refinement” of \mathcal{B}_1 in the sense that it contains finer concepts than \mathcal{B}_1 . For instance, instead of a “rougher” formal concept no. 6 from \mathcal{B}_1 with its extent {a, b, c} and its intent {L, G}, \mathcal{B}_2 contains two finer formal concepts, namely,

no. 7 with its extent $\{a,b\}$ and its intent $\{L,IG\}$ and no. 8 with its extent $\{c\}$ and its intent $\{L,gG\}$. Although the example is an illustrative one (and simplifies some effects), it illustrates well what happens if one increases the level of granularity of attributes.

One can refine a given attribute according to a specified hierarchy. For instance, an attribute “large” (representing distances, e.g., from 30km to 1000km) can be refined, resulting into attributes “a bit large” (30–100km), “medium large” (101–300km), “very large” (301–1000km). Furthermore, “very large” can be refined, resulting into attributes VL1 (301–600km) and VL2 (601–1000km). The hierarchy in question can be formally described as follows.

Definition 2: Let X be a set of objects. A gl-tree (granularity-level tree) for attribute y is a rooted tree with the following properties:

- each node of the tree is labeled by a symbol (denoted usually by $y, y_i, z \dots$ and called an attribute); the root is denoted by y ;
- to each label z of a node there is associated a set $\{z\}^\downarrow \subseteq X$; objects from $\{z\}^\downarrow$ are considered as objects to which attribute z applies;
- if a node labeled by z has as its successors nodes labeled by z_1, \dots, z_n , then $\{\{z_1\}^\downarrow, \dots, \{z_n\}^\downarrow\}$ is a partition of $\{z\}^\downarrow$.

Remark 1: (1) The fact that $\{\{z_1\}^\downarrow, \dots, \{z_n\}^\downarrow\}$ is a partition of $\{z\}^\downarrow$ means that, first, each $\{z_i\}^\downarrow$ is non-empty; second, for $i \neq j$ we have $\{z_i\}^\downarrow \cap \{z_j\}^\downarrow = \emptyset$ (attributes z_i and z_j are disjoint); third, $\{z_1\}^\downarrow \cup \dots \cup \{z_n\}^\downarrow = \{z\}^\downarrow$ (attributes z_1, \dots, z_n cover $\{z\}^\downarrow$).

(2) The definition of a structure describing several levels of granularity may be more general. We work with the above definition for the sake of simplicity.

Example 1: Consider Tab. I. Then one may consider a simple gl-tree for attribute G with a root labeled by G, two successors of the root, labeled by IG and dG, and the corresponding sets of objects given by $\{G\}^\downarrow = \{a, b, c, d, e\}$, $\{IG\}^\downarrow = \{a, b, d\}$, $\{dG\}^\downarrow = \{c, e\}$.

A selection of an appropriate level of granularity can be described by the following notion of a cut in a gl-tree.

Definition 3: A cut in a gl-tree for y is a set $C = \{y_1, \dots, y_n\}$ of labels of nodes of the gl-tree such that for each leaf node u , there exists exactly one node v on the path from the root to u such that the label of v belongs to C .

Remark 2: (1) In other words, C is a cut if and only if $\{\{y_1\}^\downarrow, \dots, \{y_n\}^\downarrow\}$ is a partition of $\{y\}^\downarrow$. In formally, a cut is a refinement of attribute y which can be obtained by moving down the paths of the tree, starting in the root.

(2) For example, $\{G\}$ and $\{IG, dG\}$ are the only cuts of the gl-tree from Example 1.

The relation of a refinement induces a partial order on the set of all cuts of a given gl-tree by putting for cuts $C_1 = \{y_1, \dots, y_n\}$ and $C_2 = \{z_1, \dots, z_m\}$,

$$C_1 \leq C_2$$

iff $\{\{y_1\}^\downarrow, \dots, \{y_n\}^\downarrow\}$ is a subpartition of $\{\{z_1\}^\downarrow, \dots, \{z_m\}^\downarrow\}$. For instance, $\{IG, dG\} \leq \{G\}$, cf. above.

Let now $\langle X, Y, I \rangle$ be an input data table. Suppose that we have for each attribute $y \in Y$ a gl-tree T_y for y . Let for each $y \in Y$, C_y be a cut in T_y and denote by

$$C = \{C_y \mid y \in Y\}$$

the collection of all these cuts. Each such a collection C induces a data table $\langle X, Y_C, I_C \rangle$ such that

$$Y_C = \bigcup_{y \in Y} C_y$$

and we put for each $z \in Y_C$

$$\langle x, z \rangle \in I_C \text{ iff } x \in \{z\}^\downarrow.$$

That is, $\langle X, Y_C, I_C \rangle$ results from $\langle X, Y, I \rangle$ by replacing each attribute $y \in Y$ by the corresponding collection C_y of attributes (refinements of y).

Example 2: For the example from Tab. I, putting $C_1 = \{\{L\}, \{R\}, \{G\}\}$ and $C_2 = \{\{L\}, \{R\}, \{IG\}, \{dG\}\}$, the left table of Tab. I is just $\langle X, Y_{C_1}, I_{C_1} \rangle$ and the right table of Tab. I is just $\langle X, Y_{C_2}, I_{C_2} \rangle$.

Remark 3: It is easy to see that putting $C = \{\{y\} \mid y \in Y\}$ we have $\langle X, Y, I \rangle = \langle X, Y_C, I_C \rangle$, according to intuition.

Denote the concept lattice corresponding to $\langle X, Y_C, I_C \rangle$ by $\mathcal{B}(X, Y_C, I_C)$ or simply by \mathcal{B}_C .

Each collection C of cuts represents a selection of levels of granularity of the attributes under consideration. Now, the main question we are interested in the following: given two selections C_1 and C_2 of levels of granularity, what is the relationship between the corresponding concept lattices $\mathcal{B}(X, Y_{C_1}, I_{C_1})$ and $\mathcal{B}(X, Y_{C_2}, I_{C_2})$? Due to the limited scope, we restrict ourselves to the condition when C_1 is a refinement of C_2 , denoted by

$$C_1 \leq C_2,$$

meaning that for each $y \in Y$ we have $C_{1y} \leq C_{2y}$ for the corresponding cuts $C_{1y} \in C_1$ and $C_{2y} \in C_2$.

Theorem 4: If $C_1 \leq C_2$ then for each formal concept $\langle A, B \rangle \in \mathcal{B}(X, Y_{C_2}, I_{C_2})$ there are formal concepts $\langle A_k, B_k \rangle \in \mathcal{B}(X, Y_{C_1}, I_{C_1})$, $k \in K$, such that $A = \bigcup_{k \in K} A_k$.

Proof: Omitted due to lack of space. ■

The previous theorem says that if we refine our attributes then the extent (cluster of objects) of each formal concept from the concept lattice of the “rougher attributes” is a union of extents (clusters of objects) of the concept lattice of the “finer attributes”. We omit further theoretical description of the relationships due to lack of space.

IV. ILLUSTRATIVE EXAMPLES

We now present illustrative examples. We use Hasse diagrams and label the nodes corresponding to formal concepts by boxes containing concept descriptions. For example, $(\{1, 3, 7\}, \{a, b\})$ is a concept with extent $\{1, 3, 7\}$ and intent $\{a, b\}$. Consider a data table described in Tab. III. The

	cause	day	time
accident 1	speed	thursday	9-10
accident 2	alcohol	friday	23-24
accident 3	priority	saturday	9-10
accident 4	priority	monday	9-10
accident 5	brakes	saturday	10-11
accident 6	steering	thursday	12-13
accident 7	steering	sunday	10-11
accident 8	brakes	monday	10-11
accident 9	speed	monday	1-2

TABLE III

FORMAL CONTEXT GIVEN BY ACCIDENTS AND THEIR PROPERTIES.

table represents data about nine car accidents (accident 1, ..., accident 9) and their three attributes (cause, day, time). Attribute cause describes the cause of the accident, attribute day describes the day the accident happened, and attribute time describes the time interval when the accident happened.

The accidents represent the objects, i.e. X has nine elements. We denote the accidents by their numbers only, i.e. $X = \{1, \dots, 9\}$. In order to obtain data tables with binary attributes, we may consider the following binary attributes and the corresponding gl-trees:

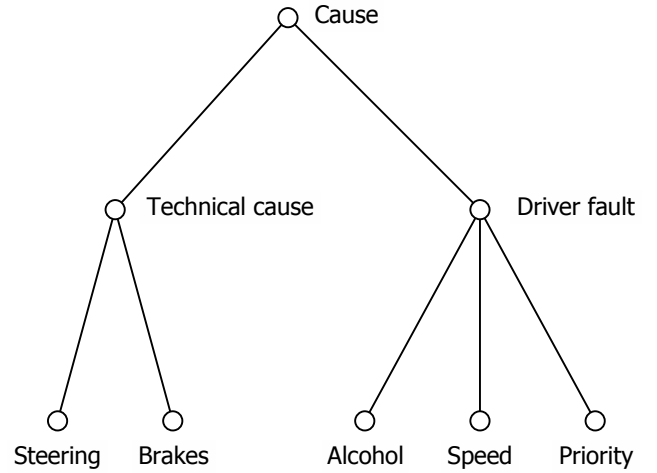
- Attributes related to cause: The most general attribute (root of the tree) will be called “cause”, it is the label of the root of the tree and we have $\{\text{cause}\}^\downarrow = X$. The root has two successors labeled by “technical cause” and “driver fault” and we have $\{\text{technicalcause}\}^\downarrow = \{5, 6, 7, 8\}$ and $\{\text{driverfault}\}^\downarrow = \{1, 2, 3, 4, 9\}$. Refinements of “technical cause” are the attributes “steering” and “brakes” with $\{\text{steering}\}^\downarrow = \{6, 7\}$ and $\{\text{brakes}\}^\downarrow = \{5, 8\}$. Refinements of “driver fault” are the attributes “alcohol”, “speed”, and “priority” with $\{\text{alcohol}\}^\downarrow = \{2\}$, $\{\text{speed}\}^\downarrow = \{1, 9\}$, and $\{\text{priority}\}^\downarrow = \{3, 4\}$. The corresponding gl-tree is depicted in Fig. 1.
- Attributes related to day: The most general attribute (root of the tree) will be called “day” and it has attributes “working day” and “weekend”, and “monday”, dots, “sunday” as refinements (we omit details).
- Attributes related to time: The most general attribute (root of the tree) will be called “time” and it has attributes “daytime” and “night”, and “0-1”, ..., “23-0” as refinements (we omit details).

Consider first a selection of granularity levels given by a collection $\mathcal{C} = \{C_c, C_d, C_t\}$ with $C_c = \{\text{brakes, steering, alcohol, speed, priority}\}$, $C_d = \{\text{monday, \dots, sunday}\}$, $C_t = \{0-1, \dots, 23-0\}$. The corresponding concept lattice is depicted in Fig. 2.

Suppose now the user find the formal concepts too fine and their number too large. The user can select other granularity levels, e.g. those represented by a collection $\mathcal{C} = \{C_c, C_d, C_t\}$ with $C_c = \{\text{technical cause, driver fault}\}$, $C_d = \{\text{working day, \dots, weekend}\}$, $C_t = \{\text{daytime, night}\}$. The corresponding concept lattice is depicted in Fig. 3.

Both of the concept lattices provide a classification of the

Fig. 1. A gl-tree of attribute cause.



accidents. One can see that the concept lattice corresponding to a smaller level of granularity has a less number of formal concepts and that these concepts can be seen as providing a rougher granulation of the set of objects and thus provides a rougher classification.

V. CONCLUSIONS AND FUTURE RESEARCH

The paper presents preliminary results on incorporating the idea of granulations and levels of granularity into formal concept analysis. The main practical effect of the presented approach is a possibility to control, in a parameterized way, the number of extracted formal concepts from input data and to control the granulation by means of the formal concepts.

The future research will be focused on the following topics.

- Relationships to conceptual scaling. In the framework of FCA, many-valued (non-binary) attributes are handled by means of so-called conceptual scaling [8]. There is an obvious connection between the levels of attribute granularity considered in this paper and different scalings of many-valued attributes which needs to be explored.
- In addition to formal concepts, the other output of FCA is represented by so-called attribute implications [8]. The effect of changing granularity levels of attributes on the extracted attribute implications and interesting subsets of attribute implications (like non-redundant bases) will be investigated.
- The basic setting of FCA was generalized to fuzzy attributes in several papers, see e.g. [1], [2], [6], [10]. Investigation of the topics presented in this paper is a natural way to continue the state of art.
- We presented only basic theoretical insight; the next step is to look at further relationships between concept lattices

Fig. 2. Concept lattice of a data table from Tab. III with $\mathcal{C} = \{C_c, C_d, C_t\}$ and $C_c = \{\text{brakes, steering, alcohol, speed, priority}\}$, $C_d = \{\text{monday, \dots, sunday}\}$, $C_t = \{0-1, \dots, 23-0\}$.

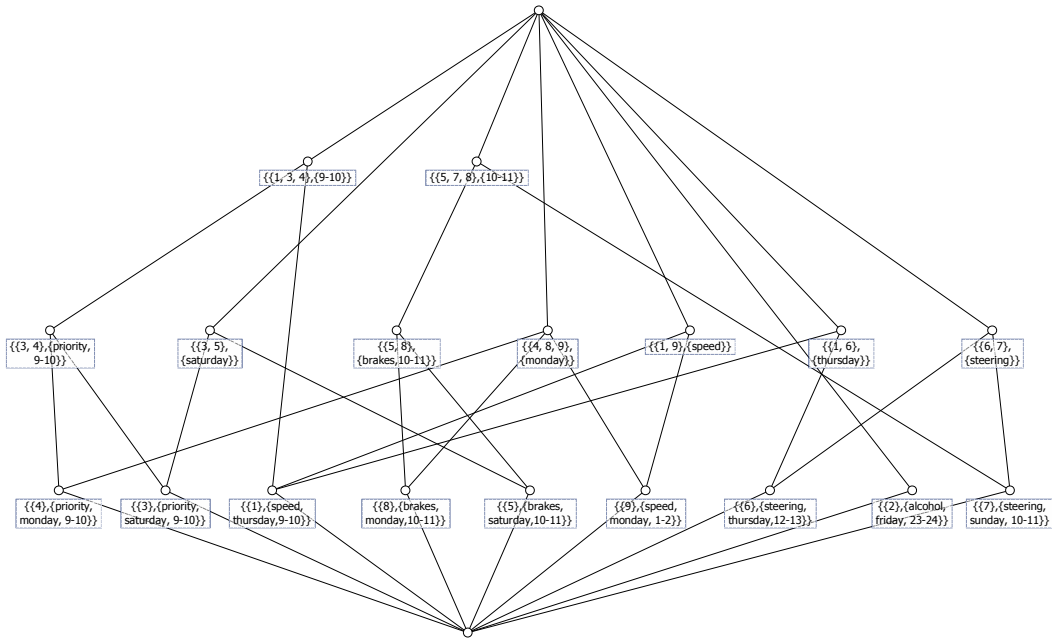
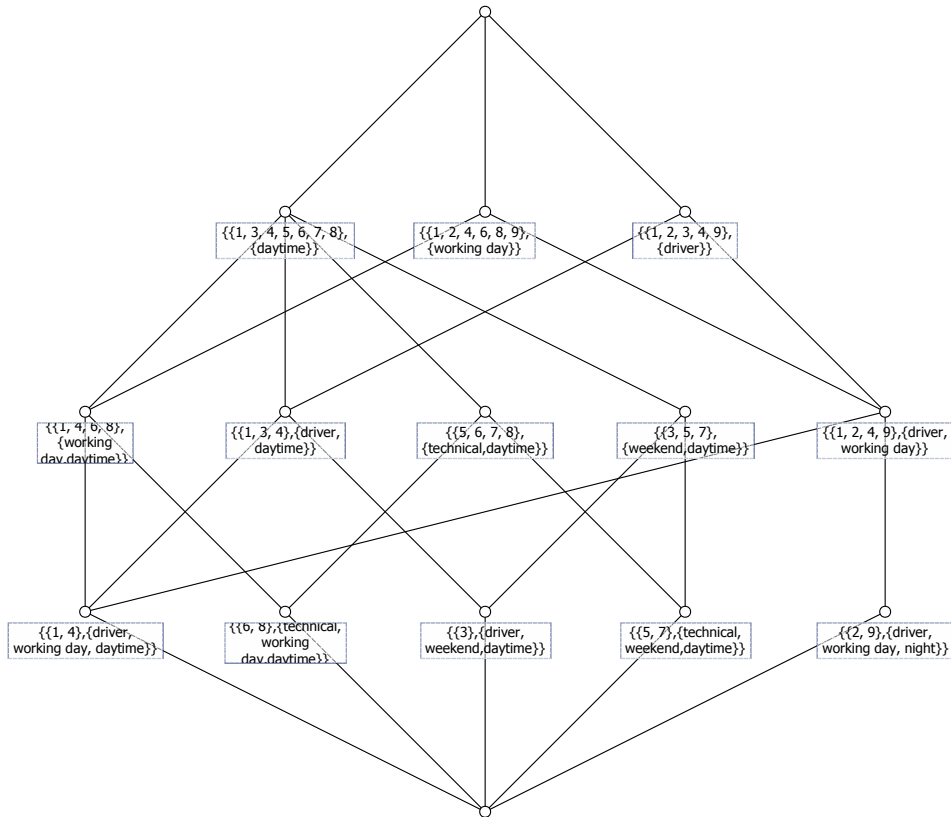


Fig. 3. Concept lattice of a data table from Tab. III with $\mathcal{C} = \{C_c, C_d, C_t\}$ and $C_c = \{\text{technical cause, driver fault}\}$, $C_d = \{\text{working day, \dots, weekend}\}$, $C_t = \{\text{daytime, night}\}$.



of data with a changed level of granularity.

- Algorithms. Design of algorithms enabling to compute a new concept lattice corresponding to a data with an increased level of granularity with the help of the original concept lattice.

ACKNOWLEDGMENT

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

REFERENCES

- [1] Bělohlávek R.: *Fuzzy Relational Systems: Foundations and Principles*. Kluwer Academic/Plenum Publishers, New York, 2002.
- [2] Bělohlávek R.: Concept lattices and order in fuzzy logic. *Ann. Pure Appl. Logic* **128**(2004), 277–298.
- [3] Bělohlávek R., Sklenář V., Zacpal J.: Formal concept analysis with hierarchically ordered attributes. *Int. J. General Systems* **33**(4)(2004), 283–294.
- [4] Bělohlávek R., Sklenář V., Zacpal J.: Concept lattices constrained by equivalence relations. *Proc. CLA 2004*, Ostrava, Czech Republic, pp. 58–66.
- [5] Bělohlávek R., Sklenář V., Zacpal J.: Formal concept analysis constrained by attribute-dependency formulas. In: B. Ganter and R. Godin (Eds.): *ICFCA 2005*, Lecture Notes in Computer Science 3403, pp. 176–191, Springer-Verlag, Berlin/Heidelberg, 2005.
- [6] Burusco A., Fuentes-González R.: The study of the L-fuzzy concept lattice. *Mathware & Soft Computing*, **3**(1994), 209–218.
- [7] Carpineto C., Romano G.: *Concept Data Analysis. Theory and Applications*. J. Wiley, 2004.
- [8] Ganter B., Wille R.: *Formal Concept Analysis. Mathematical Foundations*. Springer-Verlag, Berlin, 1999.
- [9] Maier D.: *The Theory of Relational Databases*. Computer Science Press, Rockville, 1983.
- [10] Pollandt S.: *Fuzzy Begriffe*. Springer-Verlag, Berlin/Heidelberg, 1997.
- [11] Wille R.: Restructuring lattice theory: an approach based on hierarchies of concepts. In: Rival I.: *Ordered Sets*. Reidel, Dordrecht, Boston, 1982, 445–470.
- [12] Zadeh L. A.: Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic. *Fuzzy Sets and Systems* **90**(2)(1997), 111–127.