The Lattice of Concept Graphs of a Relationally Scaled Context

Susanne Prediger and Rudolf Wille

Technische Universität Darmstadt, Fachbereich Mathematik
Schloßgartenstr. 7, D–64289 Darmstadt,
{prediger,wille}@mathematik.tu-darmstadt.de

To be published in: W. Tepfenhart et al.: Knowledge Science and Engineering with Conceptual Structures, Lecture Notes in Artificial Intelligence, Springer, Berlin 1999

Abstract. The aim of this paper is to contribute to Data Analysis by clarifying how concept graphs may be derived from data tables. First it is shown how, by the method of relational scaling, a many-valued data context can be transformed into a power context family. Then it is proved that the concept graphs of a power context family form a lattice which can be described as a subdirect product of specific intervals of the concept lattices of the power context family (each extended by a new top-element). How this may become practical is demonstrated using a data table about the domestic flights in Austria. Finally, the lattice of syntactic concept graphs over an alphabet of object, concept, and relation names is determined and related to the lattices of concept graphs of the power context families which are semantic models of the given contextual syntax.

Contents

1 Introduction

Conceptual Graphs have been introduced by J. F. Sowa as a system of logic "to express meaning in a form that is logically precise, humanly readable, and computationally tractable" [So92]. In [Wi97] and [Pr98b] Sowa's Theory of Conceptual Graphs has been used in combination with *Formal Concept Analysis* [GW99] to design a mathematical logic of judgment as part of a contextual logic. This *"Contextual Logic"* is understood as a mathematization of the traditional philosophical logic which is based on "the three essential main functions of thinking - concepts, judgments, and conclusions" [Ka88]. Contextual Logic, which is philosophically supported by Peirce's pragmatic epistemology, is grounded on families of related formal contexts whose formal concepts allow a mathematical representation of the concepts and relations of conceptual graphs. Such representation of a conceptual graph is called a *"Concept Graph"* of the context family from which it is derived. To indicate the specific relationship between the considered contexts, such a family of contexts is named a *"Power Context Family"*.

The aim of this paper is to contribute to Data Analysis by clarifying how concept graphs may be derived from data tables. Since data tables mostly relate objects, attributes, and atttribute values, our approach starts with many-valued contexts which have been introduced to formalize such data tables (see [Wi82], [GW99]). In Section 2 we explain first how to turn a many-valued context into a power context family. This transformation, called *"Relational Scaling"*, is guided by the specific purpose of the data analysis to perform. We demonstrate the method of relational scaling by data about the domestic flights in Austria. In Section 3 we show how to determine all concept graphs of a power context family. For this task it is useful that the concept graphs of a power context family form a lattice which is isomorphic to a subdirect product of specific intervals of the concept lattices of the power context family (each extended by a new top-element). How this may become practical is demonstrated by the example of the domestic flights in Austria. In Section 4, more generally, the lattice of syntactic concept graphs, defined on a contextual alphabet, is determined. Its connection to the lattices of concept graphs of power context families is also clarified. Further research is discussed in the final section.

The following explanations presuppose some knowledge about Formal Concept Analysis for which we refer to the monograph [GW99]. For a better understanding of the connection between Conceptual Graphs and Formal Concept Analysis, the papers [Wi97] and [Pr98a] might be helpful.

2 From a Many-valued Context to a Power Context Family

Data tables representing relationships between objects, attributes, and attribute values are mathematized by many-valued contexts. A *many-valued context* is defined as a set structure (G, M, W, I) where G is a set of (*formal*) *objects*, M is a set of (*formal*) *attributes*, W is a set of (*formal*) *attribute values*, and I is a ternary relation between G, M, and W (i.e. $I \subseteq G \times M \times W$) for which $(g, m, v) \in I$ and $(g, m, w) \in I$ always imply that $v = w$; $(g, m, w) \in I$ is read: the object g has the attribute value w for the attribute m. The data table in Figure 1 (see [OAG98]) may be understood as a many-valued context (G, M, W, I) for which G is the set of all listed flights, M is the set consisting of the attributes "Airline", "Departure Airport", "Departure Time", "Arrival Airport", "Arrival Time", "Days", and "Aircraft", while W is a set containing all attribute values described by the entries in the columns of the table.

		Departure		Arrival			
Flight	Airline	Airport	Time	Airport	Time	Days	Aircraft
070	VO	Vienna	07.50	Innsbruck	08.40	1-6	F70
071	VO	Innsbruck	06.25	Vienna	07.20	1-5	F70
072a	VO	Vienna	10.20	Innsbruck	11.35	6	DH8
072b	VO	Vienna	10.50	Innsbruck	12.05	1-5, 7	DH8
073a	VO	Innsbruck	08.35	Vienna	09.45	67	DH8
073b	VO	Innsbruck	09.05	Vienna	09.55	1-5	F70
074	VO	Vienna	13.55	Innsbruck	15.10	2-5	DH8
075	VO	Innsbruck	11.40	Vienna	12.50	1-5	DH8
076a	VO	Vienna	17.45	Innsbruck	18.40	1-6	F70
076b	VO	Vienna	18.40	Innsbruck	19.55	7	DH8
077	VO	Innsbruck	15.35	Vienna	16.45	2-5	DH8
078a	VO	Vienna	20.35	Innsbruck	21.25	1-4	F70
078b	VO	Vienna	21.30	Innsbruck	22.45	7	DH8
078c	VO	Vienna	21.40	Innsbruck	22.35	5	CRJ
330	VO	Linz	06.20	Salzburg	06.50	1-6	CRJ
331	VO	Salzburg	11.20	Linz	11.45	1-5	CRJ
332	VO	Linz	16.05	Salzburg	16.35	1-5	CRJ
333	VO	Salzburg	21.50	Linz	22.15	1-5, 7	CRJ
409	VO	Graz	12.10	Linz	12.45	1-5	CRJ
410	VO	Linz	16.10	Graz	16.50	1-5	CRJ
412	VO	Linz	10.35	Graz	11.10	1-5	CRJ
413	VO	Graz	06.15	Salzburg	06.50	1-5	CRJ
415	VO	Graz	17.30	Salzburg	18.10	1-5	CRJ
416	VO	Salzburg	21.50	Graz	22.25	1-5, 7	CRJ
417	VO	Graz	17.15	Linz	17.45	7	CRJ
501	VO	Klagenfurt	06.00	Salzburg	06.45	1-5	DH8
502	VO	Salzburg	21.55	Klagenfurt	22.40	1-5, 7	DH8
531*	VO-OS	Linz	06.00	Vienna	06.45	1-6	DH8
532*	VO-OS	Vienna	10.40	Linz	11.20	1-5, 7	DH8
533*	VO-OS	Linz	08.35	Vienna	09.25	1-7	DH8
534*	VO-OS	Vienna	22.15	Linz	23.00	1-5, 7	DH8
536a*	VO-OS	Vienna	17.10	Linz	17.55	5	DH8
536b*	VO-OS	Vienna	17.15	Linz	17.55	1-4, 7	DH8
537*	VO-OS	Linz	12.00	Vienna	12.50	1-5, 7	DH8
538*	VO-OS	Vienna	20.30	Linz	21.15	1-7	DH8
539*	VO-OS	Linz	18.15	Vienna	19.00	1-5, 7	DH8
540*	VO-OS	Vienna	10.45	Graz	11.30	1-7	DH8
541*	VO-OS	Graz	06.05	Vienna	06.45	1-6	DH8
542*	VO-OS	Vienna	13.50	Graz	14.35	1-5	DH8
543*	VO-OS	Graz	08.50	Vienna	09.35	1-7	DH8
544*	VO-OS	Vienna	17.20	Graz	18.00	1-7	DH8
545*	VO-OS	Graz	11.55	Vienna	12.35	1-5	DH8
546*	VO-OS	Vienna	19.40	Graz	20.20	1-7	DH8
547*	VO-OS	Graz	15.30	Vienna	16.15	1-5, 7	DH8
548*	VO-OS	Vienna	22.30	Graz	23.10	1-5, 7	DH8
549*	VO-OS	Graz	15.30	Vienna	16.15	1-5, 7	DH8
550*	VO-OS	Vienna	07.25	Klagenfurt	08.15	1-5	DH8
551*	VO-OS	Klagenfurt	06.00	Vienna	06.50	1-6	DH8
552*	VO-OS	Vienna	10.40	Klagenfurt	11.30	1-7	DH8
553*	VO-OS	Klagenfurt	08.40	Vienna	09.30	1-7	DH8
554*	VO-OS	Vienna	13.55	Klagenfurt	14.50	1-5	DH8
555*	VO-OS	Klagenfurt	11.55	Vienna	12.45	1-7	DH8
556*	VO-OS	Vienna	17.10	Klagenfurt	18.00	1-7	DH8
557*	VO-OS	Klagenfurt	15.15	Vienna	16.10	1-5	DH8
558*	VO-OS	Vienna	19.50	Klagenfurt	20.45	1-7	DH8
559*	VO-OS	Klagenfurt	18.20	Vienna	19.10	1-7	DH8
560*	VO-OS	Vienna	22.30	Klagenfurt	23.20	457	DH8
561*	VO-OS	Klagenfurt	21.00	Vienna	22.00	457	DH8
590*	VO-OS	Vienna	10.25	Salzburg	11.20	1-7	DH8
591*	VO-OS	Salzburg	17.15	Vienna	18.10	7	DH8
593*	VO-OS	Salzburg	08.15	Vienna	09.15	1-7	DH8
594*	VO-OS	Vienna	17.35	Salzburg	18.35	1-7	DH8
595*	VO-OS	Salzburg	11.45	Vienna	12.40	1-7	DH8
596a*	VO-OS	Vienna	20.25	Salzburg	21.20	6	DH8
596b*	VO-OS	Vienna	20.35	Salzburg	21.30	1-5, 7	DH8
597*	VO-OS	Salzburg	19.05	Vienna	20.00	1-7	DH8
1557	VO	Klagenfurt	16.00	Vienna	16.50	7	DH8
1583	VO	Innsbruck	15.10	Graz	15.55	7	CRJ
1596	VO	Vienna	14.05	Salzburg	15.05	5	DH8
2980	VO	Innsbruck	06.10	Salzburg	06.40	1-7	DH8
2981	VO	Salzburg	12.30	Innsbruck	13.00	1-7	DH8
2983	VO	Salzburg	16.40	Linz	17.05	1-5	CRJ
2984	VO	Innsbruck	14.35	Salzburg	15.10	1-7	DH8
2985	VO	Salzburg	21.40	Innsbruck	22.05	1-7	DH8
2986	VO	Innsbruck	10.20	Salzburg	10.55	7	DH8

Fig. 1. A Many-valued Context about *Domestic flights in Austria*

	Airport	>=20 min	>=25 min	>=30 min	>= 3 km	>= 4 km	>= 5 km	>=12 km	>=16 km
Graz	×	×	×	×	×	×	×	×	
Innsbruck	×	×	×	×	×	×			
Klagenfurt	×	×			×				
Linz	×	×			×	×	×	×	×
Salzburg	×	×			×	×			
Vienna	×	×	×		×	×	×	×	×

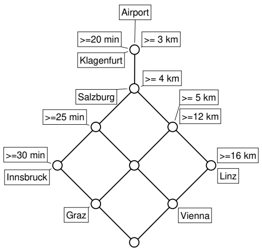

Fig. 2. Conceptual Scale *Airports* and its Concept Lattice

Conceptual Scaling [GW89] has been established as a useful method for the conceptual analysis of many-valued data contexts. For such data analysis it is desirable to make available, besides structures of formal concepts, also structures of formal judgments. Therefore we extend the method of conceptual scaling to that of *"Relational Scaling"* for deriving structures of concept graphs from many-valued contexts in addition to concept lattices.

Before describing this in general, we explain the idea of relational scaling by our example of the domestic flights in Austria. The aim is to turn the many-valued context represented in Figure 1 into a family of formal contexts whose formal concepts also yield binary relations. Such transformation should be guided by some purpose which we assume to be the support of flight information. In the formal context represented in Figure 2, for the six airports given in Figure 1, the minimum connecting time between domestic flights and the distance from the airport to the city are indicated (in this scale, additional information from [OAG98] is coded); the concept lattice of this formal context is shown next to it. The concept lattice in Figure 4 yields, for the 75 domestic flights given

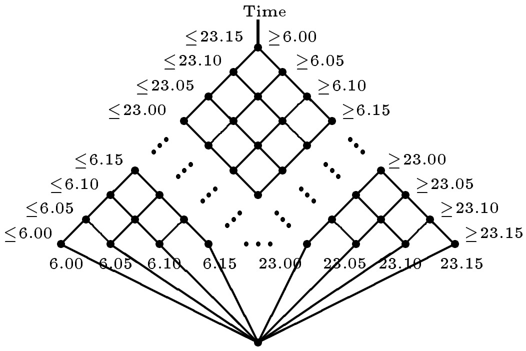

Fig. 3. Concept Lattice of the Conceptual Scale *Time*

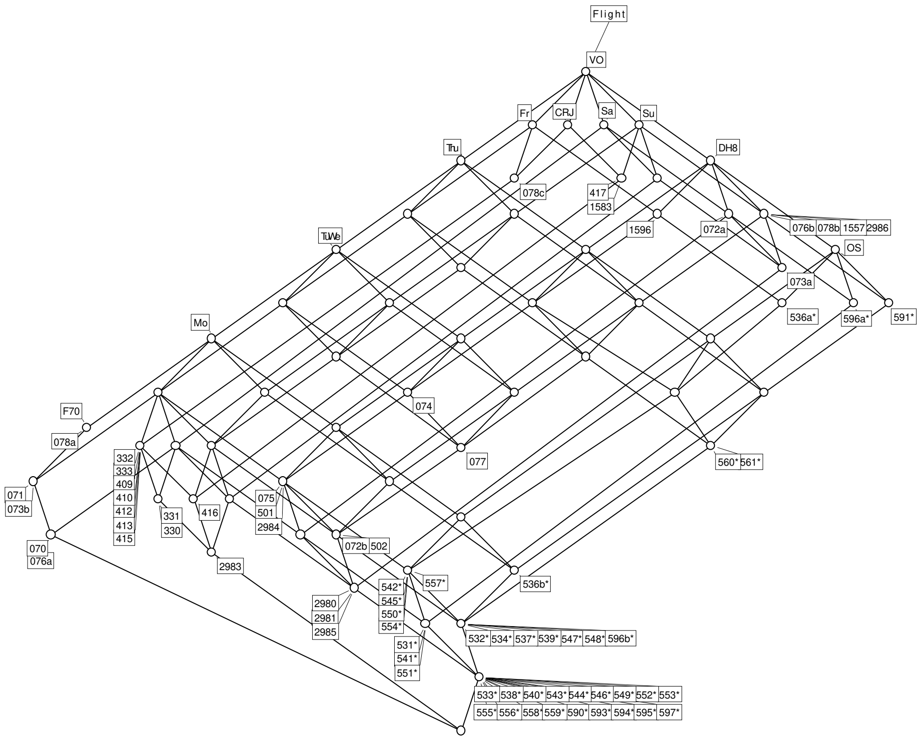

Fig. 4. Concept Lattice of the Conceptual Scale *Flights*

in Figure 1, the information about the airline, the aircraft, and the days of operating. The lattice of time intervals is represented in Figure 3. Connections between formal attributes in Figure 1 are coded by the binary relations "Flight - Departure Airport", "Flight - Departure Time", "Flight - Arrival Airport", "Flight - Arrival Time", and represented with respect to the attributes "From", "To", "a.m.", "p.m.", "Graz", "Innsbruck", "Klagenfurt", "Linz", "Salzburg", and "Vienna" by the nested line diagram in Figure 5.

If the underlying contexts of the concept lattices in Figure 2, 3, and 4 are associated as a union $\mathbb{K}_0$ and if $\mathbb{K}_2$ is the underlying context of the concept lattice in Figure 5, then the contexts $\mathbb{K}_0$ and $\mathbb{K}_2$ form a power context family. In general, a *power context family* is a sequence $\vec{\mathbb{K}} := (\mathbb{K}_k)_{k=0,\dots,n}$ of formal contexts $\mathbb{K}_k := (G_k, M_k, I_k)$ $(k = 0, \dots, n)$ with $G_k \subseteq (G_0)^k$ for $k \geq 1$. The formal concepts of $\mathbb{K}_k$ with $k = 1, \dots, n$ represent by their extents k-ary relations on the object set G_0; they are therefore called *"relation concepts"*.

To derive a power context family from a many-valued context (G, M, W, I), several contexts are formed (guided by some purpose) in combining elements of G and W to object sets of formal contexts where the objects of each one of these contexts have always to be k-tuples for a fixed natural number k; these contexts are called *relational scales*. Object sets of formal contexts may also be formed by

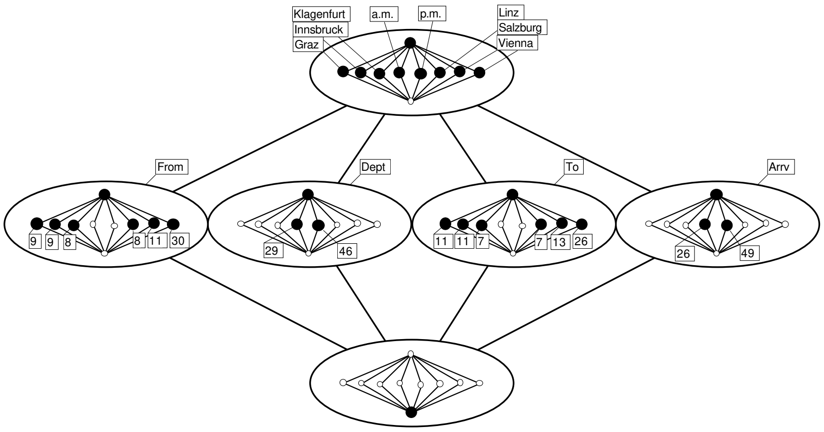

Fig. 5. Concept Lattice of the Relational Scale *Flight Schedule*

single elements of G and W; those contexts are called *conceptual scales.* Then the many-valued context together with the chosen conceptual and relational scales is said to be a *relationally scaled context.* Now, a power context family can be derived from a relationally scaled context by associating the conceptual scales to a formal context $\mathbb{K}_0$ and the relational scales to formal contexts $\mathbb{K}_1, \ldots, \mathbb{K}_n$ where the resulting power context family $\vec{\mathbb{K}} := (\mathbb{K}_k)_{k=0,\ldots,n}$ should represent the same information as the relationally scaled context from which it is obtained.

In order to analyze the structure of the data given in the resulting power context family, concept lattices and concept graphs are derived from it. In [VW95], it is shown how the software tool TOSCANA helps to determine and visualize concept lattices from conceptually scaled contexts. For deriving concept graphs from power context families, the first steps are done in [Wi97] and [Pr98b]. In the next section it is explained how *all* concept graphs and specially their natural order can be derived from a power context family.

3 The Lattice of Concept Graphs of a Power Context Family

Concept graphs of a power context family are finite directed multi-hypergraphs whose vertices and edges are specifically labelled by concepts and objects taken from the given power context family. A finite *directed multi-hypergraph* is defined as a set structure (V, E, ν) consisting of two finite sets V and E and a mapping $\nu : E \to \bigcup_{k=1}^{n} V^k$ $(n \geq 2)$; the elements of V and E are called *vertices* and *edges*, respectively, and, if $\nu(e) = (v_1, \ldots, v_k)$, we say that $v_1, \ldots, v_k$ are the *adjacent vertices* of the *k-ary edge* e. We write $|v| = 0$ for $v \in V$ and $|e| = k$ for $e \in E$ with $\nu(e) = (v_1, \ldots, v_k)$.

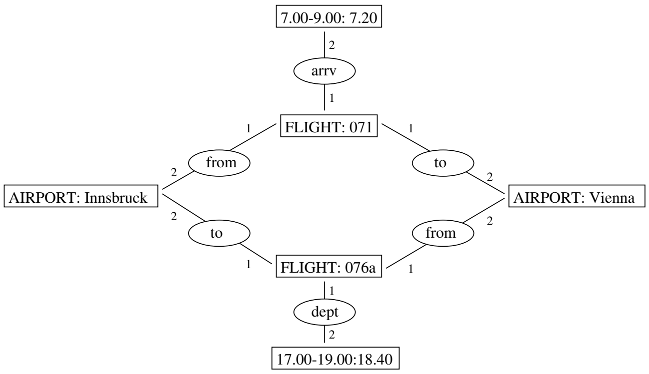

Fig. 6. Concept Graph *Commuter Flight Connection*

A *(simple) concept graph of a power context family* $\vec{\mathbb{K}} := (\mathbb{K}_k)_{k=0,\dots,n}$ with $\mathbb{K}_k := (G_k, M_k, I_k)$ for $k = 0, \dots, n$ is a set structure $\mathfrak{G} := (V, E, \nu, \kappa, \rho)$ where (V, E, ν) is a finite directed multi-hypergraph, κ assigns to each vertex v a formal concept $\kappa(v)$ of $\mathbb{K}_0$ and to each k-ary edge e a formal concept $\kappa(e)$ of $\mathbb{K}_k$, and ρ yields finite sets $\rho(v)$ of references from the extents of the $\kappa(v)$ so that the extents of the $\kappa(e)$ consist of the k-tuples formed by the references of the adjacent vertices of e, respectively; more precisely,

- $\kappa\colon V \cup E \to \bigcup_{k=0,\dots,n} \underline{\mathfrak{B}}(\mathbb{K}_k)$ is a mapping such that $\kappa(u) \in \underline{\mathfrak{B}}(\mathbb{K}_k)$ for all $u \in V \cup E$ with $|u| = k$, and
- $\rho\colon V \cup E \to \bigcup_{k=0,\dots,n} \mathfrak{P}_{fin}(G_k)\backslash\{\emptyset\}$ is a mapping such that $\rho(u) \subseteq Ext(\kappa(u))$ for all $u \in V \cup E$ and, if $|u| = k > 0$ and $\nu(u) = (v_1, \dots, v_k)$, $\rho(u) = \rho(v_1) \times \cdots \times \rho(v_k)$.

An example for a concept graph of the derived power context family presented above, is given in Figure 6 where the flight connection of a commuter living in Innsbruck and working in Vienna is formalized. The concepts of this concept graph are taken from the conceptual scales described above and the relations are relation concepts of the relational scale in Figure 5 (the concepts having the same name as the attributes in the scales are the attribute concepts of the corresponding attribute). The commuter has to be in Vienna at about 8 o'clock (we took the concept representing the interval 7.00 - 9.00) and can go back as soon as possible after 17 o'clock. Therefore, he takes the flights no. 071 and 076a. For the organization of the flight, it might be interesting to know more specialized concept graphs, having all information given in the concept graph above and additional information (for example about the distances of the airports to the city). For finding a suitable concept graph, a characterization of the generalization is needed.

Therefore, we consider the natural quasi-order $\lesssim$ of *generalization* on the set $\Gamma(\vec{\mathbb{K}})$ of all concept graphs of a power context family $\vec{\mathbb{K}}$. For two concept graphs $\mathfrak{G}_1 := (V_1, E_1, \nu_1, \kappa_1, \rho_1)$ and $\mathfrak{G}_2 := (V_2, E_2, \nu_2, \kappa_2, \rho_2)$, we say $\mathfrak{G}_1$ is *more general* than $\mathfrak{G}_2$ (in symbols: $\mathfrak{G}_1 \lesssim \mathfrak{G}_2$) if for all $u \in V_2 \cup E_2$ there exist $u_1, \dots, u_j \in V_1 \cup E_1$ with $|u| = |u_1| = \cdots = |u_j|$ and $\kappa(u_1), \dots, \kappa(u_j) \leq \kappa(u)$ and $\rho(u) \subseteq \rho(u_1) \cup \cdots \cup \rho(u_j)$. The concept graphs $\mathfrak{G}_1$ and $\mathfrak{G}_2$ are said to be *equivalent* in $\vec{\mathbb{K}}$ (in symbols: $\mathfrak{G}_1 \sim \mathfrak{G}_2$) if $\mathfrak{G}_1 \lesssim \mathfrak{G}_2$ and $\mathfrak{G}_2 \lesssim \mathfrak{G}_1$. The class of all concept graphs of $\vec{\mathbb{K}}$ which are equivalent to a given concept graph $\mathfrak{G}$ is denoted by $\underset{\sim}{\mathfrak{G}}$.

The set of all equivalence classes of concept graphs in $\vec{\mathbb{K}}$ together with the order induced by the quasi-order $\lesssim$ is an ordered set denoted by $\underset{\sim}{\Gamma}(\vec{\mathbb{K}})$. For the purpose-oriented search of suitable concept graphs, humanly readable representations of $\underset{\sim}{\Gamma}(\vec{\mathbb{K}})$ are desirable. For this it is useful that $\underset{\sim}{\Gamma}(\vec{\mathbb{K}})$ is always a lattice which is isomorphic to a subdirect product of specific sublattices of the concept lattices $\underline{\mathfrak{B}}(\mathbb{K}_k)$ $(k = 0, \dots, n)$, each extended by a new top element, as Proposition 1 states.

Proposition 1. *Let $\vec{\mathbb{K}} := (\mathbb{K}_k)_{k=0,\dots,n}$ be a power context family with $\mathbb{K}_k := (G_k, M_k, I_k)$ for $k = 0, \dots, n$; furthermore, for each $g \in G_k$, let*

$$L_k^g := \{\mathfrak{c} \in \underline{\mathfrak{B}}(\mathbb{K}_k) \mid g \in Ext(\mathfrak{c})\} \cup \{\top_k^g\}$$

be the interval of all superconcepts of (g'', g') in $\underline{\mathfrak{B}}(\mathbb{K}_k)$, together with a new top-element $\top_k^g$. Then $\underset{\sim}{\Gamma}(\vec{\mathbb{K}})$ is isomorphic to the subdirect product of the lattices L_k^g with $k \in \{0, \dots, n\}$ and $g \in G_k$ consisting of all elements $\vec{\mathfrak{a}} := (\mathfrak{a}_k^g)_{k=0,\dots,n}^{g \in G_k}$ of the directed product with only finitely many non-top components satisfying the following condition:

$(\star)$ *If $\mathfrak{a}_k^g \neq \top_k^g$ and $g = (g_1, \dots, g_k)$ then $\mathfrak{a}_0^{g_i} \neq \top_0^{g_i}$ for $i = 1, \dots, k$.*

Proof. By definition, a concept graph $\mathfrak{G} := (V, E, \nu, \kappa, \rho)$ is equivalent to the disjoint union of all its elementary subgraphs consisting of at most one edge. If an elementary subgraph $\mathfrak{H}$ has an edge e and an adjacent vertex v with $\kappa(v) \neq \top_0 := (G_0, G_0')$, then $\mathfrak{H}$ is equivalent to the disjoint union of the subgraph that is only consisting of the single vertex v and of $\mathfrak{H}$, modified by setting $\kappa(v) := \top_0$. This argument shows that a concept graph $\mathfrak{G}$ is always equivalent to the disjoint union of all its elementary subgraphs consisting of only one vertex and of all concept graphs derived from the elementary subgraphs with exactly one edge by replacing the images of the adjacent vertices under κ by $\top_0$. Further, we use that a concept graph consisting of only one vertex v with object set $\rho(v)$ is equivalent to the disjoint union of $|\rho(v)|$-many of its copies having just one object out of $\rho(v)$ as reference; analogously, a concept graph consisting of only one edge (having adjacent $\top_0$-vertices) with object set $\rho(e)$ is equivalent to the disjoint union of $|\rho(e)|$-many of its copies having just one object out of $\rho(e)$ as reference. In this way we obtain that the concept graph $\mathfrak{G}$ is equivalent to the disjoint union of the derived *atomic concept graphs* which are either single vertices with only one reference or single edges whose adjacent vertices have assigned only one reference and concepts $\top_0$.

Now, for $g \in G_k$, let $\mathfrak{c}_k^g(\mathfrak{G})$ be the element $\top_k^g$ if there is no $u \in V \cup E$ with $g \in \rho(u)$, and let it otherwise be the infimum of all $\kappa(u)$ with $g \in \rho(u)$ (and

$|u| = k$). In the second case we construct the concept graph consisting only of $\hat{u}$ with $\kappa(\hat{u}) = \mathfrak{c}_k^g(\mathfrak{G})$ and $\rho(\hat{u}) = \{g\}$ and, if $g = (g_1, \ldots, g_k)$, also of $\hat{v}_1, \ldots, \hat{v}_k$ with $\nu(\hat{u}) = (\hat{v}_1, \ldots, \hat{v}_k)$, $\kappa(\hat{v}_i) = \top_0$ and $\rho(\hat{v}_i) = g_i$ for $i = 1, \ldots, k$. The constructed concept graph is equivalent to the disjoint union of all the derived atomic concept graphs having g as reference.

To sum up, the concept graph $\mathfrak{G} := (V, E, \nu, \kappa, \rho)$ is equivalent to the disjoint union of the atomic concept graphs $\mathfrak{G}(\mathfrak{c}_0^g(\mathfrak{G})) := (\{v\}, \emptyset, \nu_0^g, \kappa_0^g, \rho_0^g)$ ($g \in G_0$ and $v \in V$) with $\kappa_0^g(v) = \mathfrak{c}_0^g(\mathfrak{G})$ and $\rho_0^g(v) = \{g\}$ and of the atomic concept graphs $\mathfrak{G}(\mathfrak{c}_k^g(\mathfrak{G})) := (\{v_1, \ldots, v_k\}, \{e\}, \nu_k^g, \kappa_k^g, \rho_k^g)$ (with $k \in \{1, \ldots, n\}$, $g \in G_k$, $e \in E$ and $\nu(e) = (v_1, \ldots, v_k)$) with $\kappa_k^g(e) = \mathfrak{c}_k^g(\mathfrak{G})$ and $\rho_k^g(e) = \{g\}$. It follows that

$$\begin{aligned} \eta^{\vec{\mathbb{K}}}: \quad \underset{\sim}{\Gamma}(\vec{\mathbb{K}}) &\to \prod (L_k^g \mid k \in \{0, \ldots, n\} \text{ and } g \in G_k) \quad \text{with} \\ \eta^{\vec{\mathbb{K}}}(\underset{\sim}{\mathfrak{G}}) &:= (\mathfrak{c}_k^g(\mathfrak{G}) \mid k \in \{0, \ldots, n\} \text{ and } g \in G_k) \end{aligned}$$

is a mapping, the image of which consists of all elements of the direct product with only finitely many non-top components satisfying condition ($\star$). It can be easily seen that this image is a subdirect product. For concept graphs $\mathfrak{G}_1$ and $\mathfrak{G}_2$, we have the equivalences

$$\begin{aligned} \mathfrak{G}_1 \lesssim \mathfrak{G}_2 \quad &\Leftrightarrow \quad \forall k \in \{0, \ldots, n\}\ \forall g \in G_k\colon\ \mathfrak{c}_k^g(\mathfrak{G}_1) \leq \mathfrak{c}_k^g(\mathfrak{G}_2) \\ &\Leftrightarrow \quad \eta^{\vec{\mathbb{K}}}(\underset{\sim}{\mathfrak{G}}_1) \leq \eta^{\vec{\mathbb{K}}}(\underset{\sim}{\mathfrak{G}}_2). \end{aligned}$$

Therefore, $\eta^{\vec{\mathbb{K}}}$ is an injective homomorphism from the set $\underset{\sim}{\Gamma}(\vec{\mathbb{K}})$ into the product $\prod(L_k^g \mid k \in \{0, \ldots, n\}$ and $g \in G_k)$. Thus, the assertion of the proposition is proved. □

Proposition 1 yields, for the concept graphs of a power context family, a system of representatives for their equivalence classes which makes the ordering of generalization transparent. These representatives are described by the elements of all the direct products

$$\prod_{(k,g) \in U} L_k^g \setminus \{\top_k^g\}$$

for which U is a finite subset of $\bigcup_{k=0,\ldots,n} \{k\} \times G_k$ satisfying the implication

$$(g, k) \in U \text{ and } g = (g_1, \ldots, g_k) \Rightarrow (0, g_1), \ldots, (0, g_k) \in U.$$

An element $\vec{\mathfrak{a}} := (\mathfrak{a}_k^g)_{(k,g) \in U}$ of the product represents the concept graph which is the disjoint union of the atomic concept graphs $\mathfrak{G}(\mathfrak{a}_k^g)$ with $(k, g) \in U$. In the full direct product $\prod(L_k^g \mid k \in \{0, \ldots, n\}$ and $g \in G_k)$ this concept graph is represented by the element which coincides with $\vec{\mathfrak{a}}$ on U and has outside U the corresponding top-elements as components (the top-element $\top_k^g$ as a component indicates that g is not a reference in the represented graph).

Proposition 1 clarifies how to deduce concept graphs from the concept lattices of a power context family. This shall be demonstrated by our flight example. In Figure 7 the lattice of concept graphs is shown so far that the representation of the concept graph in Figure 6 becomes visible, i.e. for each vertex and edge of the concept graph in Figure 6, we have considered its reference g and represented

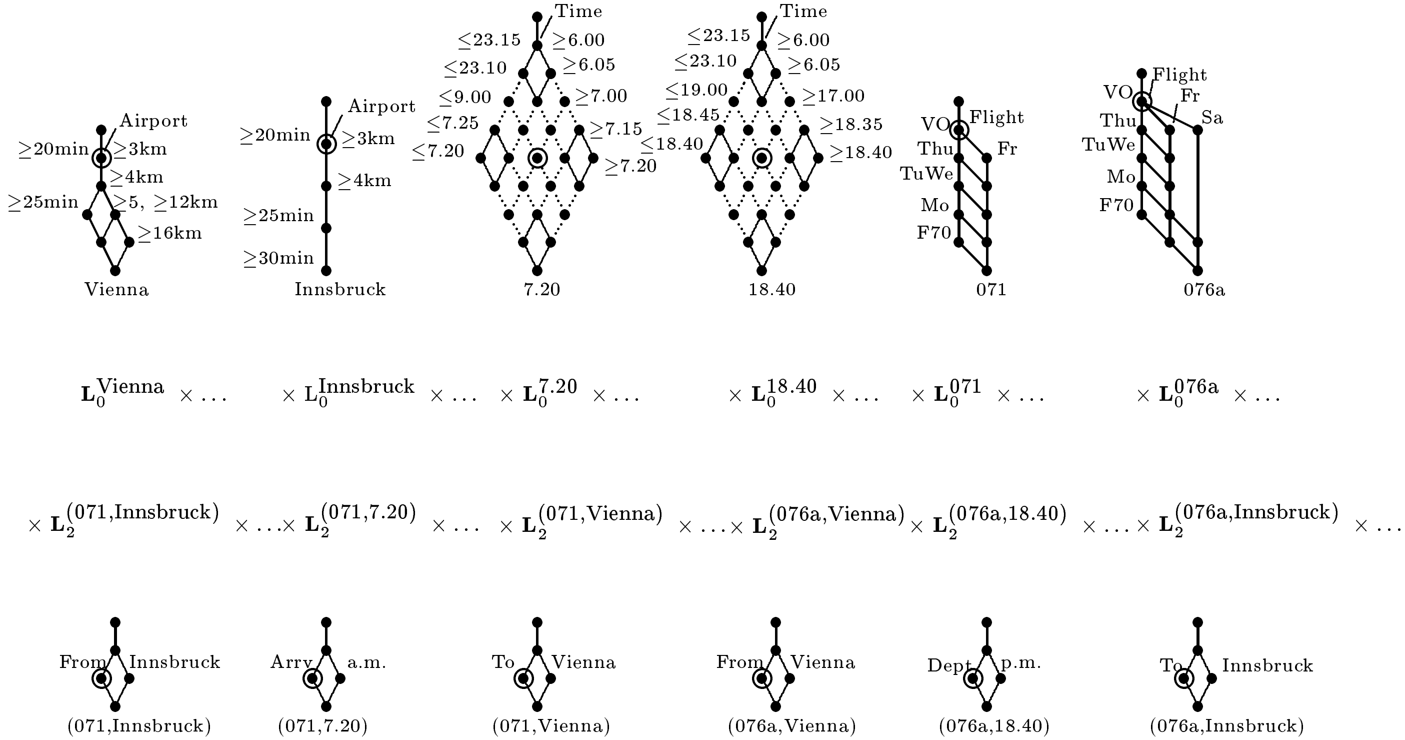

Fig. 7. Lattice of Concept Graphs of the Relationally Scaled Context *Flights in Austria*

the lattice L_k^g (where $k=0$ if g is reference of a vertex and $k=2$ if g is a pair of objects, i. e. a reference of an edge). Theses lattices are obtained by adding a new top element to the interval of all superconcepts of the object concept (g'', g') of the reference g in $\underline{\mathfrak{B}}(\mathbb{K}_k)$. For example, for the vertice of the concept graph in Figure 6 being labbeled with Flight and referenced with $\mathsf{071}$, the lattice L_0^{071}, without the top element, is a sublattice of the lattice in Figure 4.

For deducing the concept graph from the product of all lattices L_k^g, we take, for each g, a component $\mathfrak{a}_k^g$ that is represented as an element of L_k^g. For example, the component $\mathsf{Flight}_0^{\mathsf{071}}$ is represented by the attribute concept of Flight in the lattice L_0^{071}. For our concept graph in Figure 6, we choose the following components (and the top elements for all other g):

$$\mathsf{AIRPORT}_0^{\mathsf{Vienna}}, \mathsf{AIRPORT}_0^{\mathsf{Innsbruck}}, \mathsf{7.00\text{-}9.00}_0^{\mathsf{7.20}}, \mathsf{17.00\text{-}19.00}_0^{\mathsf{18.40}},$$
$$\mathsf{FLIGHT}_0^{\mathsf{071}}, \mathsf{FLIGHT}_0^{\mathsf{076a}}, \mathsf{from}_2^{(\mathsf{071,Innsbruck})}, \mathsf{arrv}_2^{(\mathsf{071,7.20})},$$
$$\mathsf{to}_2^{(\mathsf{071,Vienna})}, \mathsf{from}_2^{(\mathsf{076a,Vienna})}, \mathsf{dept}_2^{(\mathsf{076a,18.40})}, \mathsf{to}_2^{(\mathsf{076a,Innsbruck})}.$$

Their representations in the corresponding lattices are marked by a second circle. This may justify hopes that line diagrams could support the finding of meaningful concept graphs for interpreting data. If, for example, the commuter wants a concept graph that is more specified than the one in Figure 6, he can take each graph that is represented by elements being under the marked components. Since the lattices L_k^g can be easily represented as concept lattices of suitable contexts so that their product becomes the concept lattice of the sum of those contexts, it is possible to use the TOSCANA software [VW95] for navigating visually through the lattice of concept graphs of a given power context family (corresponding to a relationally scaled data context).

4 The Lattice of Concept Graphs of a Contextual Syntax

So far, we have studied the lattice of concept graphs of a power context family. The results change slightly when we start with a contextual syntax and examine all concept graphs over the given alphabet, independent of a concrete model. For this, we consider concept graphs as syntactical constructs with semantics in power context families. This has been presented in [Pr98a]; here, we only repeat briefly the main definitions.

Considering a contextual syntax, we start with a *conceptual alphabet* which is a triple $(\mathcal{G}, \mathcal{C}, \mathcal{R})$ where $\mathcal{G}$ is a finite set of *object names*, $(\mathcal{C}, \leq_{\mathcal{C}})$ is a finite ordered set of *concept names*, and $(\mathcal{R}, \leq_{\mathcal{R}})$ is a set, partitioned into finite ordered sets $(\mathcal{R}_k, \leq_{\mathcal{R}_k})$ of *relation names* (with $k = 1, \ldots, n$). These orders are determined by the taxonomies of the domains in view; they formalize background knowledge.

A *(syntactic) concept graph over the alphabet* $(\mathcal{G}, \mathcal{C}, \mathcal{R})$ is a structure $\mathfrak{G} := (V, E, \nu, \kappa, \rho)$ where κ and ρ map into $\mathcal{G}, \mathcal{C}$ and $\mathcal{R}$. Thus, it is a directed multi-hypergraph with vertices and edges labelled by object, concept and relation names. More precisely,

- $\kappa\colon V \cup E \to \mathcal{C} \cup \mathcal{R}$ is a mapping such that $\kappa(V) \subseteq \mathcal{C}$ and $\kappa(E) \subseteq \mathcal{R}$, and all $e \in E$ with $\nu(e) = (v_1, \ldots, v_k)$ satisfy $\kappa(e) \in \mathcal{R}_k$, and
- $\rho : V \cup E \to \bigcup_{k=0,\ldots,n} \mathfrak{P}(\mathcal{G}^k)\backslash\{\emptyset\}$ is a mapping with $\rho(V) \subseteq \mathfrak{P}(\mathcal{G})$ and $\rho(e) = \rho(v_1) \times \cdots \times \rho(v_k)$ for all $e \in E$ with $\nu(e) = (v_1, \ldots, v_k)$.

For this syntactical construct, a *semantics* is given in a power context family $\vec{\mathbb{K}} := (\mathbb{K}_k)_{k=0,\ldots,n}$ with $\mathbb{K}_k := (G_k, M_k, I_k)$ for each k. The object names are interpreted by objects of G_0, the concept names by concepts of $\mathbb{K}_0$ and the relation names of $\mathcal{R}_k$ by relation concepts of $\mathbb{K}_k$. This interpretation is described by an order-preserving mapping $\iota : \mathcal{G} \cup \mathcal{C} \cup \mathcal{R} \to G_0 \cup \mathfrak{B}(\mathbb{K}_0) \cup \bigcup_{k=1,\ldots,n} \underline{\mathfrak{B}}(\mathbb{K}_k)$. The context-interpretation $(\vec{\mathbb{K}}, \iota)$ is called a model if ι is consistent with all information given by the concept graph, i.e. all $u \in V \cup E$ satisfy $\iota(\rho(u)) \subseteq \mathrm{Ext}\ (\iota(\kappa(u)))$.

An interesting model for our purpose is the so-called standard model of a given concept graph. We recall the definition: The *standard model* of the concept graph $\mathfrak{G} := (V, E, \nu, \kappa, \rho)$ over the alphabet $(\mathcal{G}, \mathcal{C}, \mathcal{R})$ is defined by $\vec{\mathbb{K}}^{\mathfrak{G}} := (G_k^{\mathfrak{G}}, M_k^{\mathfrak{G}}, I_k^{\mathfrak{G}})_{k=0,\ldots,n}$ with $G_0^{\mathfrak{G}} := \mathcal{G}, M_0^{\mathfrak{G}} := \mathcal{C}$, $G_k^{\mathfrak{G}} := \mathcal{G}^k$, and $M_k^{\mathfrak{G}} := \mathcal{R}_k$ for all $k = 1, \ldots, n$. The incidence relation $I_0^{\mathfrak{G}}$ is defined in such a way that all $g \in \mathcal{G}$ and $c \in \mathcal{C}$ satisfy $gI_0^{\mathfrak{G}}c$ if there exists a $v \in V$ with $\kappa(v) \leq_{\mathcal{C}} c$ and $g \in \rho(v)$. Analogously, all $(g_1, \ldots, g_k) \in \mathcal{G}^k$ and all $R \in \mathcal{R}_k$ with $k = 1, \ldots, n$ satisfy $(g_1, \ldots, g_k)I_k^{\mathfrak{G}}R$ if there exists an $e \in E$ with $\kappa(e) \leq_{\mathcal{R}} R$ and $(g_1, \ldots, g_k) \in \rho(e)$. In this power context family, the object names of $\mathfrak{G}$ are interpreted by themselves, the concept and relation names by the corresponding attribute concepts.

We can use the standard model as an interesting tool for the problem of entailment of concept graphs. By definition, $\mathfrak{G}_1$ entails $\mathfrak{G}_2$ if $\mathfrak{G}_2$ is valid in every model for $\mathfrak{G}_1$. The notion of entailment corresponds to a sound and complete set of derivation rules (see [Pr98a]) and the definition of generalization given above. Thus, entailment of two concept graphs can be characterized by subsumption of the incidence relations of their standard models. This is stated in the following proposition that is proved in [Pr98b].

Proposition 2. *Let $\mathfrak{G}_1$ and $\mathfrak{G}_2$ be two concept graphs over the same alphabet and let $(G_k^{\mathfrak{G}_i}, M_k^{\mathfrak{G}_i}, I_k^{\mathfrak{G}_i})_{k=0,\ldots,n}$ for $i = 1, 2$ be their standard models. Then, we have*

$$\mathfrak{G}_1 \models \mathfrak{G}_2 \quad \Longleftrightarrow \quad I_k^{\mathfrak{G}_1} \supseteq I_k^{\mathfrak{G}_2} \textit{ for all } k = 0, \ldots, n.$$

The proposition implies that equivalent concept graphs have equal standard models. With it, we can characterize the lattice $(\underline{\Gamma}(\mathcal{A}), \models)$ of (equivalence classes of) concept graphs of a given alphabet $\mathcal{A}$ by means of the incidence relation in the corresponding standard models. The greatest lower bound of the two equivalence classes $\underline{\mathfrak{G}}_1$ and $\underline{\mathfrak{G}}_2$ is the equivalence class of the concept graph $\mathfrak{G}_1 \dot\cup \mathfrak{G}_2$ which is the disjoint union of both parts. The incidence relations of its standard model are $I_k^{\mathfrak{G}_1 \dot\cup \mathfrak{G}_2} = I_k^{\mathfrak{G}_1} \cup I_k^{\mathfrak{G}_2}$ for all $k = 0, \ldots, n$. The lowest upper bound (i.e. the least common generalization) of two concept graphs (resp. their equivalence classes) is given by the intersection of the incidence relations. By this means, we can easily get that the least common generalization is obtained by the disjoint union of all common generalizations.

Furthermore, we can derive from Proposition 2 that the lattice of equivalence classes of concept graphs of a given alphabet is complete and distributive because intersection and union are. For finite distributive lattices, we know from Birkhoff's Theorem (cf. eg. [GW99]) that they are isomorphic to the lattice of the orderfilters of the ordered set of all $\wedge$-irreducible elements of the lattice. Since the $\wedge$-irreducible elements of $\underset{\sim}{\Gamma}(\mathcal{A})$ are exactly the equivalence classes of the atomic concept graphs (those that consist of only one isolated vertex with a single reference or one edge with its adjacent vertices labelled with $\top$ and only one reference), we obtain the following result:

Proposition 3. *The lattice $(\underset{\sim}{\Gamma}(\mathcal{A}), \models)$ of all equivalence classes of concept graphs of an alphabet $\mathcal{A}$ is the free $\wedge$-semilattice over the ordered set of the atomic concept graphs of the contextual syntax.*

This proposition helps us to clarify how the lattice $\underset{\sim}{\Gamma}(\mathcal{A})$ of concept graphs of a given alphabet $\mathcal{A}$ is related to the lattice $\underset{\sim}{\Gamma}(\vec{\mathbb{K}})$ of concept graphs of a power context family $\vec{\mathbb{K}}$ for a context-interpretation $(\vec{\mathbb{K}}, \iota)$ of $\mathcal{A}$. With Proposition 1 and $\underset{\sim}{\Gamma}(\mathcal{A})$ being the free $\wedge$-semilattice over the ordered set of atomic concept graphs, we obtain that there exists a ι-faithful $\wedge$-homomorphism from $\underset{\sim}{\Gamma}(\mathcal{A})$ to $\underset{\sim}{\Gamma}(\vec{\mathbb{K}})$. Thus, whereas the order of $\underset{\sim}{\Gamma}(\mathcal{A})$ is only determined by the orders $\leq_{\mathcal{C}}$ and $\leq_{\mathcal{R}}$ of the alphabet, the $\underset{\sim}{\Gamma}(\vec{\mathbb{K}})$ is restricted by additional dependencies given in $\vec{\mathbb{K}}$.

To sum up, the lattice of concept graphs of a power context family as well as the lattice of concept graphs of a contextual syntax can completely be described by structural considerations. This gives us additional methods to determine logical dependencies of concept graphs besides the inference tools presented in [Pr98a].

5 Further Research

During the work on this paper, "Relational Scaling" of many-valued contexts has been created as an extension of Conceptual Scaling. It seems worth to investigate the range of applications of this new method in Data Analysis. Relational Scaling especially allows to activate the Theory of Conceptual Graphs for analyzing and interpreting data. This stimulates many research questions. An important one asks for methods of finding meaningful concept graphs contributing to the fulfillment of specific purposes of data analysis. In particular, graphical methods are desirable for which the TOSCANA software should be further elaborated.

Another research problem is how to use conceptual graphs for purpose-oriented retrieval on relationally scaled databases. For this, the theory of concept graphs with quantifiers should be further developed.

From such a development, the general research program of establishing a comprehensive contextual logic would benifit too. Contextual Logic in its support of conceptual knowledge representation and processing will gain from further research about concept graphs of relationally scaled contexts. In particular, the development of conceptual knowledge and information systems should integrate such research on relational scaling and concept graphs.

References

[GW89] B. Ganter, R. Wille: Conceptual scaling. In: F. S. Roberts (ed.): Applications of Combinatorics and Graph Theory to the Biological and Social Sciences. Springer, Berlin-Heidelberg-New York 1989, 139–167.

[GW99] B. Ganter, R. Wille: Formal Concept Analysis: Mathematical Foundations. Springer, Berlin-Heidelberg-New York 1999.

[Ka88] I. Kant: Logic. Dover, New York 1988.

[OAG98] OAG Pocket Flight Guide - Europe/Africa/Middle East. Reed Elsevier, July 1998.

[Pr98a] S. Prediger: Simple concept graphs: a logic approach. In: M.-L. Mugnier. M. Chein (eds.): Conceptual Structures: Theory, Tools and Applications. Springer, Berlin-Heidelberg-New York 1998, 225–239.

[Pr98b] S. Prediger: Kontextuelle Urteilslogik mit Begriffsgraphen. Ein Beitrag zur Restrukturierung der mathematischen Logik. Dissertation, Shaker Verlag, Aachen 1998.

[So84] J. F. Sowa: Conceptual structures: information processing in mind and machine. Addison-Wesley, Reading 1984.

[So92] J. F. Sowa: Conceptual graphs summary. In: T. E. Nagle, J. A. Nagle, L. L. Gerholz, P. W. Eklund (eds.): Conceptual Structures: Current Research and Practice. Ellis Horwood 1992, 3–51.

[VW95] F. Vogt, R. Wille: TOSCANA - a graphical tool for analyzing and exploring data. In: R. Tamassia, I. G. Tollis (eds.): Graph Drawing. LNCS 894. Springer, Berlin-Heidelberg-New York 1995, 226–233.

[Wi82] R. Wille: Restructuring lattice theory: an approach based on hierarchies of concepts. In: I. Rival (ed.): Ordered sets. Reidel, Dordrecht, Boston 1982, 445–470.

[Wi97] R. Wille: Conceptual Graphs and Formal Concept Analysis. In: D. Lukose, H. Delugach, M. Keeler, L. Searle, J. Sowa (eds.): Conceptual Structures: Fulfilling Peirce's Dream. Springer, Berlin-Heidelberg-New York 1997, 290–303.