Concept Graphs without Negations: Standardmodels and Standardgraphs

Frithjof Dau

Technische Universität Darmstadt, Fachbereich Mathematik
Schloßgartenstr. 7, D-64289 Darmstadt, dau@mathematik.tu-darmstadt.de

Abstract In this article, we provide different possibilities for doing reasoning on simple concept(ual) graphs without negations or nestings. First of all, we have on the graphs the usual semantical entailment relation $\models$, and we consider the restriction $\vdash$ of the calculus for concept graph with cuts, which has been introduced in [Da02], to the system of concept graphs without cuts. Secondly, we introduce a semantical entailment relation $\models$ as well as syntactical transformation rules $\vdash$ between models. Finally, we provide definitions for standard graphs and standard models so that we translate graphs to models and vice versa. Together with the relations $\models$ and $\vdash$ on the graphs and on the models, we show that both calculi are adequate and that reasoning can be carried over from graphs to models and vice versa.

1 Introduction

This paper is mainly based on two treatises: The dissertations of the author Dau ([Da02]) and of Prediger ([Pr98a]). In [Pr98a], Prediger developed a mathematical theory for simple concept graphs. In Predigers graphs, negation cannot be expressed, so these graphs correspond to the existential-conjunctive fragment of conceptual graphs or first order predicate logic (FOPL). Reasoning on these graphs can be performed in two different ways: First of all, Prediger introduced a sound and complete calculus which consists of fairly simple transformation rules. Secondly, she assigned to each graph a corresponding *standard model* (and, vice versa, to each model a corresponding *standard graph*). For graphs without generic markers, Predigers standard models encode exactly the same information as the respective graphs. Thus, for these graphs, the reasoning on graphs can be carried over to the models.

In [Da02], the author extended the syntax of concept graphs by adding the *cuts* of Peirce's existential graphs. Cuts are syntactical devices which serve to negate subgraphs. As negation can be expressed in concept graphs with cuts, these graphs correspond to full FOPL (see [Da02]). For these graphs, a sound and complete calculus is provided, too. This calculus is not a further development of Predigers calculus, but it is based on Peirce's calculus for existential graphs. In particular, some of its transformation rules are fairly complex. In contrast to concept graphs without negations, it turns out that the information which is encoded by concept graphs with cuts cannot be encoded in standard models, so the notion of standard models has to be dropped.

In this article, we bring together the ideas of Prediger and Dau for concept graphs without negation (i. e. without cuts). First of all, on concept graphs without cuts, we introduce a restricted version of the calculus for concept graphs with cuts, where we have removed all rules where cuts are involved. Secondly, we extend Predigers notions of standard graphs and standard models (the differences to Predigers approach will be discussed in the next sections) such that even for graphs *with* generic markers, their standard models will encode exacty the same exactly the same information as the respective graphs. On the models, we introduce a semantical entailment relation $\models$ as well as transformation rules. The latter can be seen as a kind of calculus which yields a relation $\vdash$ between models. It will turn out that the relations $\models, \vdash$ for graphs and for models and the notions of standard models and standard graphs fit perfectly together.

2 Basic Definitions

In this section, we provide the basic definitions for this paper. The first four definitions are directly adopted or slightly modified definitions which can be found in [Pr98a] as well as in [Da02]. Definition 5 is a restricted version –which suits for concept graphs without negation– of the calculus in [Da02]. We start with the definitions of the underlying alphabet (which is called the *support*, *taxonomy* or *ontology* by other authors) and of simple concept graphs.

Definition 1 (Alphabet).
An alphabet *is a triple* $\mathcal{A} := (\mathcal{G}, \mathcal{C}, \mathcal{R})$ *of disjoint sets* $\mathcal{G}$, $\mathcal{C}$, $\mathcal{R}$ *such that*

- $\mathcal{G}$ *is a finite set whose elements are called* object names,
- $(\mathcal{C}, \leq_{\mathcal{C}})$ *is a finite ordered set with a greatest element* $\top$ *whose elements are called* concept names, *and*
- $(\mathcal{R}, \leq_{\mathcal{R}})$ *is a family of finite ordered sets* $(\mathcal{R}_k, \leq_{\mathcal{R}_k})$, $k = 1, \ldots, n$ *(for an* $n \in \mathbb{N}$*) whose elements are called* relation names. *Let* $\doteq \in \mathcal{R}_2$ *be a special name which is called* identity.

On $\mathcal{G} \dot\cup \{*\}$ *we define an order* $\leq_{\mathcal{G}}$ *such that* $*$ *is the greatest element* $\mathcal{G} \dot\cup \{*\}$, *but all elements of* $\mathcal{G}$ *are incomparable.*

Definition 2 (Simple Concept Graphs).
A simple concept graph over $\mathcal{A}$ *is a structure* $\mathfrak{G} := (V, E, \nu, \kappa, \rho)$ *where*

- V *and* E *are pairwise disjoint, finite sets whose elements are called* vertices *and* edges,
- $\nu : E \to \bigcup_{k \in \mathbb{N}} V^k$ *is a mapping (we write* $|e| = k$ *for* $\nu(e) \in V^k$*),*
- $\kappa : V \cup E \to \mathcal{C} \cup \mathcal{R}$ *is a mapping such that* $\kappa(V) \subseteq \mathcal{C}$, $\kappa(E) \subseteq \mathcal{R}$, *and all* $e \in E$ *with* $|e| = k$ *satisfy* $\kappa(e) \in \mathcal{R}_k$, *and*
- $\rho : V \to \mathcal{G} \dot\cup \{*\}$ *is a mapping.*

The set of these graphs is denoted by $CG^{\mathcal{A}}$. *If the alphabet is fixed, we sometimes write* CG. *For the set* E *of edges, let* $E^{id} := \{e \in E \mid \kappa(e) = \doteq\}$ *and* $E^{nonid} := \{e \in E \mid \kappa(e) \neq \doteq\}$. *The elements of* E^{id} *are called* identity-links. *Finally set* $V^* := \{v \in V \mid \rho(v) = *\}$ *and* $V^{\mathcal{G}} := \{v \in V \mid \rho(v) \in \mathcal{G}\}$. *The nodes in* V^* *are called* generic nodes.

These graphs correspond to the graphs in [Da02], where the cuts are removed. Although they are similar, there are some important differences between these graphs and the simple concept graphs as they are defined by Prediger in [Pr98a]:

First of all, in [Pr98a] Prediger assigned *sets* of objects instead of *single* objects to vertices (i.e. in [Pr98a] we have $\rho : V \to \mathfrak{P}(\mathcal{G}) \dot\cup \{*\}$ instead of $\rho : V \to \mathcal{G} \dot\cup \{*\}$). For concept graphs with cuts, it is not immediately clear what the meaning of a vertex is which is enclosed by a cut and which contains more than one object. For this reason, in [Da02] and thus in this article, ρ assigns single objects to vertices. The expressiveness of the graphs is not changed by this syntactical restriction.

Identity is in [Pr98a] expressed by an equivalence relation θ only on the set of generic vertices. In [Da02] and in this article, identity is expressed by identity links on the set of generic *and non-generic* vertices. Thus the concept graphs of this article have a slightly higher expressiveness than the concept graphs of [Pr98a]. This has to be taken into account in the definition of standard models and standard graphs, as well as in the calculus. To provide an example: Consider an alphabet with $\mathcal{C} := \{\top, A, B\}$ and $\mathcal{G} := \{a, b\}$, where A, B are incomparable concept names. In our approach, [A: a]–(=)–[B: b] is a well-defined graph[1] which expresses a proposition which cannot represented in Predigers approach. This graph entails [A: b]–(=)–[B: a]. Obviously, this entailment is based on the unique meaning of identity. For this reason, we will have rules in our calculus which capture the role of the identity relation and which allow to derive the second graph from the first one. A derivation like this cannot be performed with projections[2] ([CM92]) or with the calculus presented in [Mu00].

The next two definitions describe the 'contextual models' which we consider. These definitions appear in [Pr98a] as well as in [Da02] and are based on the theory of Formal Concept Analysis (see [GW99]).

Definition 3 (Power Context Family).

A power context family $\vec{\mathbb{K}} := (\mathbb{K}_0, \mathbb{K}_1, \mathbb{K}_2, \ldots)$ *is a family of formal contexts* $\mathbb{K}_k := (G_k, M_k, I_k)$ *that satisfies* $G_0 \neq \emptyset$ *and* $G_k \subseteq (G_0)^k$ *for each* $k \in \mathbb{N}$. *Then we write* $\vec{\mathbb{K}} := (G_k, M_k, I_k)_{k \in \mathbb{N}_0}$. *The elements of* G_0 *are the* objects *of* $\vec{\mathbb{K}}$. *A pair* (A, B) *with* $A \subseteq G_k$ *and* $B \subseteq M_k$ *is called a concept of* $\mathbb{K}_k$, *if and only*

[1] In contrast to other authors, like Mugnier [Mu00], we allow that different object names may refer to the same object, i.e. we do not adopt the *unique name assumption.* The unique name assumption is needed when a graph shall be transformed into its normal-form. This graph cannot be transformed into a normal-form and is not a well-defined graph in the approach of Mugnier.

[2] Moreover, as projections rely on normal-forms for graphs, they have further restrictions. See [Mu00].

if $A = \{g \in G_k \mid gI_k n \text{ for all } b \in B\}$ *and* $B = \{m \in M_k \mid aI_k m \text{ for all } a \in A\}$. $Ext(A,B) := A$ *is called* extension of the concept (A,B)*, and* $Int(A,B) := B$ *is called* intension of the concept (A,B)*. The set of all concepts of a formal context* $\mathbb{K}_k$ *is denoted by* $\mathfrak{B}(\mathbb{K}_k)$*. The elements of* $\bigcup_{k\in\mathbb{N}_0} \mathfrak{B}(\mathbb{K}_k)$ *are called* concepts*, and we set furthermore* $\mathfrak{R}_{\vec{\mathbb{K}}} := \bigcup_{k\in\mathbb{N}} \mathfrak{B}(\mathbb{K}_k)$*, and the elements of* $\mathfrak{R}_{\vec{\mathbb{K}}}$ *are called* relation-concepts*.*

To get a contextual structure over an alphabet, we have to interpret the object-, concept- and relation-names by objects, concepts and relation-concepts in a power context family. This is done in the following definition.

Definition 4 (Contextual Models).
For an alphabet $\mathcal{A} := (\mathcal{G}, \mathcal{C}, \mathcal{R})$ *and a power context family* $\vec{\mathbb{K}}$*, we call the disjoint union* $\lambda := \lambda_\mathcal{G} \dot\cup \lambda_\mathcal{C} \dot\cup \lambda_\mathcal{R}$ *of the mappings* $\lambda_\mathcal{G}: \mathcal{G} \to G_0$*,* $\lambda_\mathcal{C}: \mathcal{C} \to \underline{\mathfrak{B}}(\mathbb{K}_0)$ *and* $\lambda_\mathcal{R}: \mathcal{R} \to \mathfrak{R}_{\vec{\mathbb{K}}}$ *a* $\vec{\mathbb{K}}$-interpretation *of* $\mathcal{A}$ *if* $\lambda_\mathcal{C}$ *and* $\lambda_\mathcal{R}$ *are order-preserving, and* $\lambda_\mathcal{C}, \lambda_\mathcal{R}$ *satisfy* $\lambda_\mathcal{C}(\top) = \top$*,* $\lambda_\mathcal{R}(\mathcal{R}_k) \subseteq \underline{\mathfrak{B}}(\mathbb{K}_k)$ *for all* $k = 1, \ldots, n$*, and* $(g_1, g_2) \in Ext(\lambda_\mathcal{R}(\doteq)) \Leftrightarrow g_1 = g_2$ *for all* $g_1, g_2 \in G_0$*.*[3] *The pair* $(\vec{\mathbb{K}}, \lambda)$ *is called* contextual structure over $\mathcal{A}$ *or* contextual model over $\mathcal{A}$*. The set of these contextual structures is denoted by* $CS^\mathcal{A}$*. If the alphabet is fixed, we sometimes write* CS*.*

The calculus for concept graphs with cuts consists of the following rules (see[Da02]): erasure, insertion, iteration, deiteration, double cuts, generalization, specialization, isomorphism, exchanging references, merging two vertices, splitting a vertex, ⊤-erasure, ⊤-insertion, identity-erasure and identity-insertion.

Only the double-cut-rule allows to derive a concept graph with cuts from a concept graph without cuts. The rules insertion and specialization can only be applied if we have a graph with cuts. As we consider concept graphs without cuts, we remove these three rules from the calculus and interpret the remaining rules as rules for the system of concept graphs without cuts. So we have the following definition (for examples, an explanation and a precise mathematical definition for the rules, we refer to [Da02]):

Definition 5 (Calculus for Simple Concept Graphs).
The calculus for concept graphs over the alphabet $\mathcal{A} := (\mathcal{G}, \mathcal{C}, \mathcal{R})$ *consists of the following rules:*

Erasure, iteration, deiteration, generalization, isomorphism, exchanging references, merging two vertices, splitting a vertex, ⊤-erasure, ⊤-insertion, identity-erasure and identity-insertion.

If $\mathfrak{G}_a$*,* $\mathfrak{G}_b$ *are concept graphs, and if there is a sequence* $(\mathfrak{G}_1, \mathfrak{G}_2, \ldots, \mathfrak{G}_n)$ *with* $\mathfrak{G}_1 = \mathfrak{G}_a$*,* $\mathfrak{G}_b = \mathfrak{G}_n$ *such that each* $\mathfrak{G}_{i+1}$ *is derived from* $\mathfrak{G}_i$ *by applying one of the rules above, we say that* $\mathfrak{G}_b$ can be derived from $\mathfrak{G}_a$ *and write* $\mathfrak{G}_a \vdash_{pos} \mathfrak{G}_b$*.*

[3] As Prediger does not consider identity links, the last condition does not appear in her definition of contextual structures.

3 Contextual Structures: Standard Models and Semantical Entailment

In this section, we will assign to a graph its *standard model* which encodes exactly the same information as the graph. This is based on [Wi97] and has already done by Prediger in Definition 4.2.5. of [Pr98a]. Remember that identity is in [Pr98a] expressed by an equivalence relation θ only on the set of generic vertices, so Prediger used the following approach: The set of objects of the standard model consists of all object names $G \in \mathcal{G}$ and of all equivalence classes of θ. But we can express identity between arbitrary, i. e. generic or non-generic, vertices, thus we have to extend this idea. We start by defining an equivalence relation $\theta_{\mathcal{G}}$ on $V \dot\cup \mathcal{G}$, which is an appropriate generalization of Predigers θ.

Definition 6. *Let $\mathfrak{G} := (V, E, \nu, \kappa, \rho)$ be a concept graph over $\mathcal{A}$. We assume that V and $\mathcal{G}$ are disjoint. Let $\theta_{\mathfrak{G}}$ be the smallest equivalence relation on $V \dot\cup \mathcal{G}$ such that*

1. *if $\rho(v) = G \in \mathcal{G}$, then $v \theta_{\mathfrak{G}} G$, and*
2. *if $e \in E$ with $\nu(e) = (v_1, v_2)$ and $\kappa(e) \leq \doteq$, then $v_1 \theta_{\mathfrak{G}} v_2$.*

It is easy to see that two vertices which are equivalent must refer in each model to the same object, i. e. we have the following lemma:

Lemma 1. *Let $\mathfrak{G} := (V, E, \nu, \kappa, \rho)$ be a concept graph over $\mathcal{A}$. If $v_1, v_2 \in V$ with $v_1 \theta_{\mathfrak{G}} v_2$, then $ref(v_1) = ref(v_2)$ for each contextual structure $(\vec{\mathbb{K}}, \lambda)$ over $\mathcal{A}$ and each valuation $ref : V \to G_0$ with $(\vec{\mathbb{K}}, \lambda) \models \mathfrak{G}[ref]$.*

The opposite direction of the lemma holds as well, i. e. we could characterize $\theta_{\mathfrak{G}}$ by the condition in the lemma. This is not immediately clear, but it could easily be shown with the results of this paper.

Now we can assign to each concept graph an appropriate standard model which encodes exactly the same information as the graph.[4]

Definition 7 (Standard Model).
Let $\mathfrak{G} := (V, E, \nu, \kappa, \rho)$ be a concept graph over $\mathcal{A}$. We define the standard model *of $\mathfrak{G}$ as follows:*
For $\mathfrak{G} \neq \emptyset$ or $\mathcal{G} \neq \emptyset$,[5] we first define a power context family $\vec{\mathbb{K}}^{\mathfrak{G}}$ by

[4] This is possible because we consider only the existential-conjunctive fragment of concept graphs (in particular we do not consider negations or disjunctions of propositions), so we have to encode only the information a) whether objects have specific properties or whether objects are in a specific relation, b) whether objects *exist* with specific properties, and c) the conjunction of informations like these. These are the kinds of information which can be expressed in graphs (e.g. existential graphs or concept graphs) in an iconical way, i. e. we encode in standard models exactly the iconical features of concept graphs. For a deep discussion of this topic, we refer to [Sh99].

[5] As usual in logic, we only consider non-empty structures. For this reason, we have to treat the case $\mathfrak{G} = \emptyset = \mathcal{G}$ separately.

– $G_0^{\mathfrak{G}} := \{[k]\theta_{\mathfrak{G}} \mid k \in V\dot{\cup}\mathcal{G}\}$, *and* $G_i^{\mathfrak{G}} := (G_0^{\mathfrak{G}})^i$,
– $M_0^{\mathfrak{G}} := \mathcal{C}$, *and* $M_i^{\mathfrak{G}} := \mathcal{R}_i$ *for* $1 \leq i \leq n$.
– *For* $C \in \mathcal{C}$ *and* $g \in G_0$, *we set* $gI_0^{\mathfrak{G}}C \quad :\Longleftrightarrow$
  - $\exists v \in V \,.\, g = [v]\theta_{\mathfrak{G}} \wedge \kappa(v) \leq C$, *or*
  - $C = \top$.
– *For* $R_i \in \mathcal{R}_i$ *and* $g_1, \ldots, g_i \in G_0$, *we set* $(g_1, \ldots, g_i)I_i^{\mathfrak{G}}R_i \;:\Longleftrightarrow$
  - $\exists e = (v_1, \ldots, v_i) \in E \,.\, g_1 = [v_1]\theta_{\mathfrak{G}} \wedge \ldots \wedge g_i = [v_i]\theta_{\mathfrak{G}} \wedge \kappa(e) \leq R_i$, *or*
  - $i = 2$ *and* $\doteq \leq R_i \wedge g_1 = g2$.

The mappings $\lambda^{\mathfrak{G}}$ *are defined canonically:*

– $\lambda_{\mathcal{G}}^{\mathfrak{G}}(G) := [G]\theta_{\mathfrak{G}}$ *for all* $G \in \mathcal{G}$,
– $\lambda_{\mathcal{C}}^{\mathfrak{G}}(C) := \mu(C)$ *for all* $C \in \mathcal{C}$, *and*
– $\lambda_{\mathcal{R}}^{\mathfrak{G}}(R) := \mu(R)$ *for all* $R \in \mathcal{R}$.

If $\mathfrak{G} = \emptyset$ *and* $\mathcal{G} = \emptyset$, *let* g *be an arbitrary element. We define* $\vec{\mathbb{K}}^{\mathfrak{G}}$ *as follows:* $\mathbb{K}_0 := (\{g\}, \{\top\}, \{(g, \top)\})$, $\mathbb{K}_2 := (\{(g, g)\}, \{\doteq\}, \{((g, g), \doteq)\})$, *and for* $i \neq 0, 2$, *let* $\mathbb{K}_i := (\emptyset, \emptyset, \emptyset)$. *The mappings of* $\lambda^{\mathfrak{G}}$ *are defined canonically, i. e.* $\lambda_{\mathcal{G}}^{\mathfrak{G}} := \emptyset$, $\lambda_{\mathcal{C}}^{\mathfrak{G}}(\top) := \mu(\top)$, *and* $\lambda_{\mathcal{R}}^{\mathfrak{G}}(\doteq) := \mu(\doteq)$. *All remaining concept- or relation-names are mapped to the* $\perp$*-concept* $(\emptyset'', \emptyset')$ *of the respective formal context.*

The contextual structure $(\vec{\mathbb{K}}^{\mathfrak{G}}, \lambda^{\mathfrak{G}})$ *is called* standard model of $\mathfrak{G}$ *and is denoted by* $\mathcal{M}^{\mathfrak{G}}$.[6]

It is not immediately clear that the definition above yields indeed a contextual structure. Of course $\vec{\mathbb{K}}^{\mathfrak{G}}$ is a power context family. It remains to check that $\lambda := \lambda_{\mathcal{G}}^{\mathfrak{G}} \dot{\cup} \lambda_{\mathcal{C}}^{\mathfrak{G}} \dot{\cup} \lambda_{\mathcal{R}}^{\mathfrak{G}}$ fulfills the conditions of Definition 4. This is done now.

1. We have to show that $\lambda_{\mathcal{C}}^{\mathfrak{G}}$ and $\lambda_{\mathcal{R}}^{\mathfrak{G}}$ are order-preserving. We only consider the mapping $\lambda_{\mathcal{R}}^{\mathfrak{G}}$ (the case $\lambda_{\mathcal{C}}^{\mathfrak{G}}$ is done analogously). So let $R_1, R_2 \in \mathcal{R}_i$ and $(g_1, \ldots, g_i) \in \mathrm{Ext}(\lambda_{\mathcal{R}}^{\mathfrak{G}}(R_1))$. If there is an $e = (v_1, \ldots, v_i) \in E$ which satisfies $g_1 = [v_1]\theta_{\mathfrak{G}} \wedge \ldots \wedge g_i = [v_i]\theta_{\mathfrak{G}}$ and $\kappa(e) \leq R_1$, then we have $\kappa(e) \leq R_2$ as well, so, by Definition 4, we conclude $(g_1, \ldots, g_i) \in \mathrm{Ext}(\lambda_{\mathcal{R}}^{\mathfrak{G}}(R_2))$. If we have $i = 2$, $\doteq \leq R_1$ and $g_1 = g2$, then we have $\doteq \leq R_2$ as well, so, again by Definition 4, we conclude $(g_1, g_2) \in \mathrm{Ext}(\lambda_{\mathcal{R}}^{\mathfrak{G}}(R_2))$.
2. It is easy to see that $\lambda_{\mathcal{C}}^{\mathfrak{G}}(\top) = \top$ holds.
3. It remains to show: $(g_1, g_2) \in \mathrm{Ext}(\lambda_{\mathcal{R}}\mathfrak{G}(\doteq)) \Leftrightarrow g_1 = g_2$ for all $g_1, g_2 \in G_0$. The direction '$\Longleftarrow$' is easy to see: For $g \in G_0$, the second condition for R_i of Definition 4, applied to $i := 2$ and $R_i :=\doteq$ yields $(g, g)I_2^{\mathfrak{G}} \doteq$. In order to show direction '$\Longrightarrow$', we assume that we have $g_1, g_2 \in G_0$ and an edge $e = (v_1, v2) \in E$ with $g_1 = [v_1]\theta_{\mathfrak{G}}$, $g_2 = [v_2]\theta_{\mathfrak{G}}$ and $\kappa(e) \leq\doteq$. Definition 6 yields $v_1\theta_{\mathfrak{G}}v_2$, from which we conclude $g_1 = [v_1]\theta_{\mathfrak{G}} = [v_2]\theta_{\mathfrak{G}} = g_2$.

As all conditions of Definition 4 are fulfilled, we see that $(\vec{\mathbb{K}}^{\mathfrak{G}}, \lambda^{\mathfrak{G}})$ is in fact a conctextual structure.

It is easy to see that each graph holds in its standard model, i. e. we have:

[6] We write $\mathcal{M}^{\mathfrak{G}}$ instead of $\mathcal{M}_{\mathfrak{G}}$, because the different contexts of $\vec{\mathbb{K}}^{\mathfrak{G}}$ and the mappings of $\lambda^{\mathfrak{G}}$ already have indices at the bottom.

Lemma 2. *If* $\mathfrak{G} := (V, E, \nu, \kappa, \rho)$ *is a graph, then* $(\vec{\mathbb{K}}^{\mathfrak{G}}, \lambda^{\mathfrak{G}})$ *is a contextual structure over* $\mathcal{A}$. *For* $ref^{\mathfrak{G}} := \{(v, [v]\theta_{\mathfrak{G}}) \mid v \in V\}$, *we have* $(\vec{\mathbb{K}}^{\mathfrak{G}}, \lambda^{\mathfrak{G}}) \models \mathfrak{G}[ref^{\mathfrak{G}}]$.

In the following, we provide some examples for simple concept graphs (over the alphabet $(\{a, b, c\}, \{A, B, C, \top\}, \{\doteq\})$ with incomparable concept names A, B, C and their standard models.

$\mathfrak{G}_1 :=$ [$A : a$]–($\doteq$)–[$B : b$] [$C : *$] [$A : a$] [$C : *$]

$\mathcal{M}_{\mathfrak{G}_1} :=$

	A	B	C	$\top$
$\{a, b, v_1, v_2, v_4\}$	×	×		×
$\{v_3\}$			×	×
$\{v_5\}$			×	×

$\mathfrak{G}_2 :=$ [$A : a$]–($\doteq$)–[$B : b$] [$C : *$]

$\mathcal{M}_{\mathfrak{G}_2} :=$

	A	B	C	$\top$
$\{a, b, v_1, v_2\}$	×	×		×
$\{v_3\}$			×	×

$\mathfrak{G}_3 :=$ [$A : a$]–($\doteq$)–[$B : b$]

$\mathcal{M}_{\mathfrak{G}_3} :=$

	A	B	C	$\top$
$\{a, b, v_1, v_2\}$	×	×		×

$\mathfrak{G}_4 :=$ [$A : a$] [$B : b$] [$C : *$]

$\mathcal{M}_{\mathfrak{G}_2} :=$

	A	B	C	$\top$
$\{a, v_1\}$	×			×
$\{b, v_2\}$		×		×
$\{v_3\}$			×	×

$\mathfrak{G}_5 :=$ [$A : a$] [$B : b$]

$\mathcal{M}_{\mathfrak{G}_5} :=$

	A	B	C	$\top$
$\{a, v_1\}$	×			×
$\{b, v_2\}$		×		×

$\mathfrak{G}_6 :=$ [$A : a$]

$\mathcal{M}_{\mathfrak{G}_6} :=$

	A	B	C	$\top$
$\{a, v_1\}$	×			×
$\{b\}$				×

$\mathfrak{G}_7 :=$

$\mathcal{M}_{\mathfrak{G}_7} :=$

	A	B	C	$\top$
$\{a\}$				×
$\{b\}$				×

The well-known relation $\models$ on graphs can be understood as follows: $\mathfrak{G}_1 \models \mathfrak{G}_2$ holds iff $\mathfrak{G}_1$ contains the same or more information than $\mathfrak{G}_2$. This idea can be transferred to models as well. This yields the following definition:[7]

Definition 8 (Semantical Entailment between Contextual Structures).

Let $\mathcal{M}^a := (\vec{\mathbb{K}}^a, \lambda^a)$ *and* $\mathcal{M}^b := (\vec{\mathbb{K}}^b, \lambda^b)$ *be two* $\mathcal{A}$*-structures (with* $\vec{\mathbb{K}}^x = (\mathbb{K}_0^x, \ldots, \mathbb{K}_n^x)$ *and* $\mathbb{K}_i^x = (G_i^x, M_i^x, I_i^x)$, $i = 1, \ldots, n$ *and* $x = a, b$*). We set*

$$\mathcal{M}^a \models \mathcal{M}^b :\Longleftrightarrow \text{ it exists a mapping } f : G_0^b \to G_0^a \text{ with:}$$

1. *For all* $G \in \mathcal{G}$: $f(\lambda_{\mathcal{G}}^b(G)) = \lambda_{\mathcal{G}}^a(G)$ *(i. e.* f *respects* $\lambda_{\mathcal{G}}$*.)*
2. *For all* $g \in G_0^b$ *and* $C \in \mathcal{C}$: $g \in Ext(\lambda_{\mathcal{C}}^b(C)) \Longrightarrow f(g) \in Ext(\lambda_{\mathcal{C}}^a(C))$ *(i. e.* f *respects* $\lambda_{\mathcal{C}}$*).*
3. *For all* $\vec{g} \in G_i^b$ *and* $R \in \mathcal{R}_i$: $\vec{g} \in Ext(\lambda_{\mathcal{R}}^b(R)) \Longrightarrow f(\vec{g}) \in Ext(\lambda_{\mathcal{R}}^a(R))$ *where* $f(\vec{g}) = f(g_1, \ldots, g_i) := (f(g_1), \ldots, f(g_i))$ *(i. e.* f *respects* $\lambda_{\mathcal{R}}$*).*

We will sometimes write $\mathcal{M}^a \models_f \mathcal{M}^b$ *to denote the mapping* f*, too.*

[7] See also [Wi02], where Wille defines in a different way the *informational content* of a model.

We want to make the following remarks on standard models:

1. The direction of f is not surprising: It corresponds to the direction of projections between conceptual graphs (see [CM92]). [8] In fact, f can be roughly understood as projection between models (instead of between graphs). But note that projection between graphs is complete only under certain restrictions (e.g., on the normal form of one graph. But, as argued before Definition 3, not every graph can be transformed into a normal-form), but it will turn out that projection between models is complete without further restrictions. Thus, to evaluate whether a graph $\mathfrak{G}_a$ entails a graph $\mathfrak{G}_b$, a sound and complete approach is the following: First contruct the standard models $\mathcal{M}_{\mathfrak{G}_a}$ and $\mathcal{M}_{\mathfrak{G}_b}$, and then find out whether there is a 'projection' f from $\mathcal{M}_{\mathfrak{G}_b}$ to $\mathcal{M}_{\mathfrak{G}_a}$.
2. Please note that the models on the right are semantically equivalent, although they have a different number of objects. This is based on the fact that we cannot count in existential-conjunctive languages (without negation, we cannot express that we have *different* objects with the same properties).

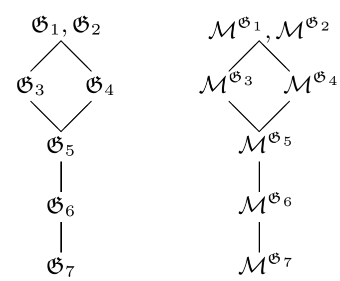
$\mathcal{M}_1 :=$

$\mathbb{K}_0^1$	P	$\top$
g	$\times$	$\times$

$\mathcal{M}_2 :=$

$\mathbb{K}_0^2$	P	$\top$
g	$\times$	$\times$
h	$\times$	$\times$

3. Concerning the relation $\models$ on the set of graphs resp. on the set of contextual structures, the graphs and their standard models above are ordered as follows:



4. In [Pr98a], standard models are compared as well, but Prediger compares only the restrictions of the incidence-relations to objects which are generated by non-generic nodes. E.g. in Predigers approach, the standard models of the graphs $\boxed{A : *}$ and $\boxed{B : *}$ are comparable, although they encode incomparable information. Thus Predigers approach is strictly weaker than semantical entailment between models.

The next theorem shows the main link between a graph and its standard model. As Prediger has no concept of semantical entailment between models, there is no corresponding theorem in [Pr98a].

Theorem 1 (Main Theorem for Graphs and their Standard Models).

[8] Another comparison can be drawn to algebra: A Standard Model of a graph can be compared with a free structure over a set of equations in an algebra, and every algebra which fulfills the equations can be mapped homomorphic into the free algebra.

Let $\mathfrak{G} := (V, E, \nu, \kappa, \rho)$ be a graph, $\mathcal{M}^{\mathfrak{G}} := (\vec{\mathbb{K}}^{\mathfrak{G}}, \lambda^{\mathfrak{G}})$ be its standard model and $\mathcal{M} := (\vec{\mathbb{K}}, \lambda)$ be an arbitrary contextual structure. Then we have

$$\mathcal{M} \models \mathcal{M}^{\mathfrak{G}} \quad \iff \quad \mathcal{M} \models \mathfrak{G} \ .$$

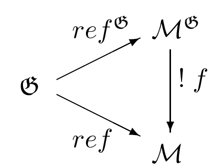
Proof:

$\Rightarrow$: We have $\mathfrak{G} \xrightarrow{ref^{\mathfrak{G}}} \mathcal{M}^{\mathfrak{G}} \xrightarrow{f} \mathcal{M}$. Set $ref := f \circ ref^{\mathfrak{G}}$. We want to show that ref is a valuation with $\mathcal{M} \models \mathfrak{G}[ref]$. First we have to check that the mapping ref is indeed a valuation. So let $v \in V^{\mathcal{G}}$ with $\rho(v) = G \in \mathcal{G}$. We have

$$ref(v) = f(ref^{\mathfrak{G}}(v)) \overset{\text{Def for val.}}{=} f(\lambda_{\mathcal{G}}^{\mathfrak{G}}(\rho(v)) \overset{\text{Def 8}}{=} \lambda_{\mathcal{G}}(\rho(v)) \ ,$$

thus ref is a valuation.
To show $\mathcal{M} \models \mathfrak{G}[ref]$, we have to check that ref satisfies the vertex- and edge-conditions of [Da02], Definition 10.3.
To see that all vertex-conditions are fulfilled, let $v \in V$ be a vertex. Let $\kappa(v) = C \in \mathcal{C}$. From $\mathcal{M}^{\mathfrak{G}} \models \mathfrak{G}[ref^{\mathfrak{G}}]$ we conclude $ref^{\mathfrak{G}}(v) \in \text{Ext}(\lambda_{\mathcal{G}}^{\mathfrak{G}}(C))$. Now condition 2) for f yields $ref(v) = f(ref^{\mathfrak{G}}(v)) \in \text{Ext}(\lambda_{\mathcal{G}}(C))$, hence ref satisfies the vertex condition for v.
The edge-conditions are shown analogously.

$\Leftarrow$: We have a valuation $ref : V \to G_0$ (where G_0 is the set of objects in $\mathcal{M} = (\vec{\mathbb{K}}, \lambda))$ with $\mathcal{M} \models \mathfrak{G}(ref)$. Furthermore we have the canonical valuation $ref^{\mathfrak{G}} : V \to G_0^{\mathfrak{G}}$ (where $G_0^{\mathfrak{G}}$ is the set of objects in $\mathcal{M}^{\mathfrak{G}}$) with $\mathcal{M}^{\mathfrak{G}} \models \mathfrak{G}(ref^{\mathfrak{G}})$. Let $(ref^{\mathfrak{G}})^{-1}$ be an inverse mapping of $ref^{\mathfrak{G}}$.



We set

$$f := ref \circ (ref^{\mathfrak{G}})^{-1} \,\dot{\cup}\, \{(\lambda_{\mathcal{G}}^{\mathfrak{G}}(G), \lambda(G)) \mid \neg\exists v \in V.\, \rho(v) = G\} \ .$$

It is easy to see that f is a function from $G_0^{\mathfrak{G}}$ to G_0. We have to check 1)–3) of Definition 8.

1. If $G \in \mathcal{G}$ such that there is no $v \in V$ with $\rho(v) = G$, then 1) is fulfilled by definition of f. So we assume that there is a $v \in V$ with $\rho(v) = G$, hence $ref^{\mathfrak{G}}(v) = \lambda_{\mathcal{G}}^{\mathfrak{G}}(G)$. Let $v_G := (ref^{\mathfrak{G}})^{-1}(\lambda_{\mathcal{G}}^{\mathfrak{G}}(G))$. We have $v \theta_{\mathfrak{G}} v_G$, hence $ref(v_G) \overset{\text{L. 1}}{=} ref(v) \overset{\mathcal{M} \models \mathfrak{G}[ref]}{=} \lambda_{\mathcal{G}}(G)$ We conclude
$$f(\lambda_{\mathcal{G}}^{\mathfrak{G}}(G)) = ref((ref^{\mathfrak{G}})^{-1}(\lambda_{\mathcal{G}}^{\mathfrak{G}}(G))) = ref(v_G) = \lambda_{\mathcal{G}}(G) \ ,$$
thus 1) is fulfilled.
2. Let $C \in \mathcal{C}$ and $g \in G_0^{\mathfrak{G}}$ with $g \in \text{Ext}(\lambda_{\mathcal{C}}^{\mathfrak{G}}(C)) = \text{Ext}(\mu(C))$. If $C = \top$, then 2) is fulfilled by definition of f. So we assume $C < \top$. Then we have a $v \in V$ with $g = [v]\theta_{\mathfrak{G}}$ and $\kappa(v) \leq C$. We set $v_g := (ref^{\mathfrak{G}})^{-1}(g)$, thus we have $v \theta_{\mathfrak{G}} v_g$. Similar to 1), we have $ref(v_g) = ref(v)$. This yields
$$f(g) = ref(ref^{\mathfrak{G}})^{-1}(g)) = ref(v_g) = ref(v) \in \text{Ext}(\lambda_{\mathcal{C}}(\kappa(v))) \ .$$
From $\kappa(v) \leq C$, hence $\text{Ext}(\lambda_{\mathcal{C}}(\kappa(v))) \subseteq \text{Ext}(\lambda_{\mathcal{C}}(C))$, we conclude 2).
3. Is shown analogously. □

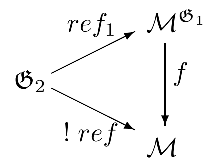
From the main theorem, we get the following corollary. The first equivalence of the corollary corresponds to Theorem 4.2.6. in [Pr98a]. But in [Pr98a], we find no result which can be compared to $\mathfrak{G}_1 \models \mathfrak{G}_2 \Leftrightarrow \mathcal{M}^{\mathfrak{G}_1} \models \mathcal{M}^{\mathfrak{G}_2}$. Again due to the lack of the concept of semantical entailment between models, Prediger has only proven an implication which is a weak variant of $\mathfrak{G}_1 \models \mathfrak{G}_2 \Rightarrow \mathcal{M}^{\mathfrak{G}_1} \models \mathcal{M}^{\mathfrak{G}_2}$

Corollary 1. *Let $\mathfrak{G}_1, \mathfrak{G}_2$ be two graphs. Then we have*

$$\mathfrak{G}_1 \models \mathfrak{G}_2 \quad \Longleftrightarrow \quad \mathcal{M}^{\mathfrak{G}_1} \models \mathfrak{G}_2 \quad \Longleftrightarrow \quad \mathcal{M}^{\mathfrak{G}_1} \models \mathcal{M}^{\mathfrak{G}_2} .$$

Proof:

The direction '⇒' of the first equivalence is trivial, and the second equivalence follows immediately from Theorem 1. So it remains to show that '⇐' of the first equivalence holds.



So let $\mathcal{M} = (\vec{\mathbb{K}}, \lambda)$ be a contextual structure with $\mathcal{M} \models \mathfrak{G}_1$. Theorem 1 yields a mapping $f : G_0^{\mathfrak{G}_1} \to G_0$ such that $\mathcal{M} \models_f \mathcal{M}^{\mathfrak{G}_1}$ holds. We furthermore have a valuation $ref_1 : V \to G_1$ with $\mathcal{M}^{\mathfrak{G}_1} \models \mathfrak{G}_2[ref_1]$. We set $ref := f \circ ref_1$ and want to show that ref is a valuation with $\mathcal{M} \models \mathfrak{G}_2[ref]$.

We have to check the vertex-conditions for ref, so let $v \in V_2$ be a vertex. Let $C := \kappa(v)$. As ref_1 fulfills the vertex-condition, we get $ref_1(v) \in \text{Ext}(\lambda^{\mathfrak{G}_1}(C))$. Now condition 2) for f yields $ref(v) = f(ref_1(v)) \in \text{Ext}(\lambda(C))$, which is the vertex-condition for v.

The edge-conditions are checked analogously. □

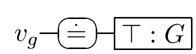
4 Standard Graphs

In the last section, we assigned to each graph a corresponding standard model which contains the same information as the graph. In this section, we do the same for the opposite direction: We assign to a model a standard graph which contains the same information. This is done with the following definition.

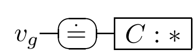
Definition 9 (Standard Graphs).

Let $\mathcal{M} := (\vec{\mathbb{K}}, \lambda)$ be a contextual structure. We define the standard graph of *$\mathcal{M}$ as follows:*

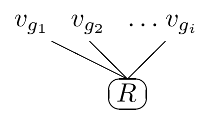
1. *For each $g \in G_0$, let $v_g :=$ $\boxed{\top : *}$ be a new vertex (i. e. we set $\kappa(v_g) := \top$ and $\rho(v_g) = *$).*
2. *For each $g \in \mathcal{G}$ with $\lambda_{\mathcal{G}}(G) = g \in G_0$, let $v_{g,G}$ be a new vertex and $e_{g,G}$ a new edge. We set $\kappa(v_{g,G}) := \top$, $\rho(v_{g,G}) := G$, $\nu(e) := (v_g, v_{g,G})$ and $\kappa(e) := \doteq$ (i. e. we add the vertex and edge on the right).*



3. *For each $C \in \mathcal{C} \backslash \{\top\}$ with $g \in Ext(\lambda_{\mathcal{C}}(C))$, let $v_{g,C}$ be a new vertex and $e_{g,C}$ a new edge. We set $\kappa(v_{g,C}) := C$, $\rho(v_{g,C}) := *$, $\nu(e) := (v_g, v_{g,C})$ and $\kappa(e) := \doteq$ (i. e. we add the vertex and edge on the right).*



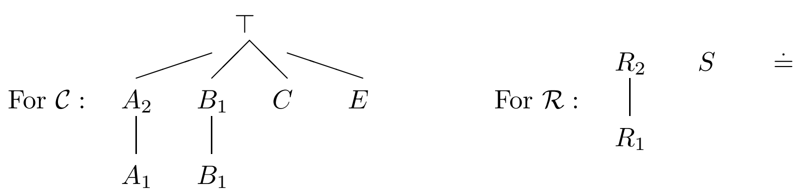
4. *For each* $R \in \mathcal{R}_i \backslash \{\dot{=}\}$ *with* $(g_1, \ldots, g_i) \in Ext(\lambda_{\mathcal{R}}(R))$, *let* $e_{g_1,\ldots,g_i,R}$ *be a new vertex. We set* $\kappa(e_{g_1,\ldots,g_i,R}) := R$ *and* $\nu(e) := (g_1, \ldots, g_i)$ *(i. e. we add the edge on the right).*



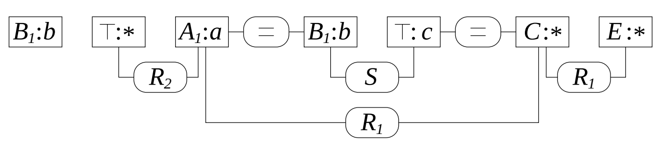
We denote this graph by $\mathfrak{G}_{(\vec{\mathbb{K}},\lambda)}$ *or* $\mathfrak{G}_{\mathcal{M}}$.[9]

In [Pr98a], Definition 4.2.15, we find a corresponding definition for Definition 9. But there is a crucial difference between Predigers approach and our approach: In [Pr98a], Prediger assigns a standard graph to a power context family instead to a contextual structure. Thus, she has first to define an alphabet which is derived from the power context family, then she defines the standard graph over this alphabet. Our approach is different: We fix an alphabet at the beginning and consider only graphs and structures over this fixed alphabet.

To get an impression of standard models and standard graphs, we provide a more extensive example. First we have to fix the alphabet. We set $\mathcal{A} := (\{a, b, c, d\}, \{A_1, A_2, B_1, B_2, C, E, \top\}, \{R_1, R_2, S, \dot{=}\})$, where R_1, R_2, S are dyadic relation names. The orderings on the concept- and relation-names shall be as follows:



We consider the following graph over $\mathcal{A}$:



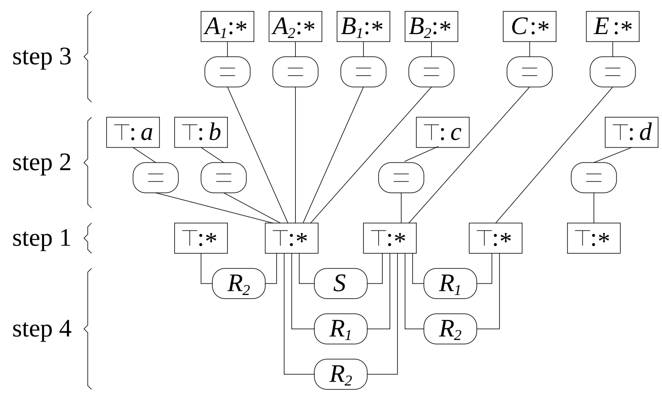
Below, we provide the standard model of this graph (the mappings λ_G, λ_C and λ_R are not explicit given, as they can easily be obtained from the power context family). We assume that the vertices of the graph are numbered from the left to the right, starting with 1, thus the i-th vertex is denoted by v_i.

9 $\mathfrak{G}_{\mathcal{M}}$ is given only up to isomorphism, but we have the implicit agreement that isomorphic graphs are identified.

$\mathbb{K}_0$	A_1	A_2	B_1	B_2	C	E	$\top$
$\{a, b, v_1, v_3, v_4\}$	$\times$	$\times$	$\times$	$\times$			$\times$
$\{v_2\}$							$\times$
$\{c, v_5, v_6\}$					$\times$		$\times$
$\{v_7\}$						$\times$	$\times$
$\{d\}$							$\times$

$\mathbb{K}_2$	R_1	R_2	S	$\doteq$
$(\{v_2\}, \{a, b, v_1, v_3, v_4\})$		$\times$		
$(\{a, b, v_1, v_3, v_4\}, \{c, v_5, v_6\})$	$\times$	$\times$	$\times$	
$(\{c, v_5, v_6\}, \{v_7\})$	$\times$	$\times$		
$(\{v_2\}, \{v_2\})$				$\times$
$(\{a, b, v_1, v_3, v_4\}, \{a, b, v_1, v_3, v_4\})$				$\times$
$(\{c, v_5, v_6\}, \{c, v_5, v_6\})$				$\times$
$(\{v_7\}, \{v_7\})$				$\times$
$(\{d\}, \{d\})$				$\times$

The standard graph of this model is given below. In the left, we sketch which vertices and edges are added by which step of Definition 9.



If we translate a model to a graph and then translate this graph back into a model, in general we do not get the same graph back, but at least a semantically equivalent graph:

Lemma 3 ($\mathcal{M}$ and $\mathcal{M}^{\mathfrak{G}_\mathcal{M}}$ are Semantically Equivalent).
Let $\mathcal{A} = (\mathcal{G}, \mathcal{C}, \mathcal{R})$ be an alphabet and let $\mathcal{M}$ be a contextual structure over $\mathcal{A}$.

1. *It holds $\mathcal{M} \models \mathcal{M}^{\mathfrak{G}_\mathcal{M}}$ and $\mathcal{M}^{\mathfrak{G}_\mathcal{M}} \models \mathcal{M}$.*
2. *If $\mathcal{M}$ satisfies furthermore $M_0 = \mathcal{C}$ and $M_i = \mathcal{R}_i$ for all $i \geq 1$, then $\mathcal{M}$ and $\mathcal{M}^{\mathfrak{G}_\mathcal{M}}$ are even isomorphic.*

Proof:
Let

$$f :=:= \begin{cases} \mathcal{M} \to \mathcal{M}^{\mathfrak{G}_\mathcal{M}} \\ g \mapsto [v_g]\theta_{\mathfrak{G}_\mathcal{M}} \end{cases}$$

(with the denotation of Definition 9 for v_g). It can easily be checked that f is bijective, and that we have $\mathcal{M} \models_{f^{-1}} \mathcal{M}^{\mathfrak{G}_\mathcal{M}}$ and $\mathcal{M}^{\mathfrak{G}_\mathcal{M}} \models_f \mathcal{M}$, hence 1 is fulfilled. For 2, we have that for each i, the contexts $\mathbb{K}_i$ and $\mathbb{K}_i^{\mathfrak{G}_\mathcal{M}}$ have the same attributes, so the conditions which are satisfied by f yield that $\mathcal{M}$ and $\mathcal{M}^{\mathfrak{G}_\mathcal{M}}$ are isomorphic. $\Box$

From 2. of Lemma 3 we conclude that each model $\mathcal{M}$ with $M_0 = \mathcal{C}$ and $M_i = \mathcal{R}_i$ for all $i \geq 1$ is already isomorphic to a standard model of a graph. Together with 1., we see that each class in the quasiorder (CS,$\models$) contains at least one standard model (but this is not uniquely determined: Each class contains infinitely many pairwise non-isomorphic standard models).

In the following, we will provide an corresponding result of the last lemma for graphs, that is we will show that $\mathfrak{G}$ and $\mathfrak{G}_{\mathcal{M}^{\mathfrak{G}}}$ are equivalent. In contrast to the last lemma, we will prove that $\mathfrak{G}$ and $\mathfrak{G}_{\mathcal{M}^{\mathfrak{G}}}$ are *syntactically* equivalent. As we know from [Da02] that the calculus for concept graphs with cuts is sound, we know that the restricted calculus $\vdash_{pos}$ we consider in this paper is sound, too. In particular, when we have shown that $\mathfrak{G}$ and $\mathfrak{G}_{\mathcal{M}^{\mathfrak{G}}}$ are syntactically equivalent, we know that are these graphs are semantically equivalent as well.

Before we prove the equivalence, we need a simple lemma.

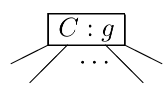
Lemma 4.
Let $\mathfrak{G} := (V, E, \nu, \kappa, \rho)$ be a concept graph, let v_1, v_2 be two new vertices, e_1, e_2 be two new edges, and let $\mathfrak{G}' := (V', E', \nu', \kappa', \rho')$ be defined as follows:

- $V' := V \dot\cup \{v_1, v_2\}$, $E' := E \dot\cup \{e_1, e_2\}$, $\nu' := \nu \dot\cup \{(e_1, (v, v_1)), (e_2, (v, v_2))\}$
- $\kappa' := \kappa|_{V \setminus \{v\} \dot\cup E} \dot\cup \{(v, \top), (v_1, \top), (v_2, \kappa(v)), (e_1, \doteq), (e_2, \doteq)\}$, *and*
- $\rho' := \rho|_{V \setminus \{v\}} \dot\cup \{(v, *), (v_1, \rho(v)), (v_2, *)\}$.

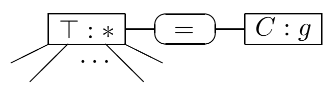
Then we have

$$\mathfrak{G} \vdash_{pos} \mathfrak{G}' \quad \text{and} \quad \mathfrak{G}' \vdash_{pos} \mathfrak{G}\ .$$

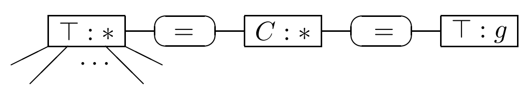
Proof:
We start with:



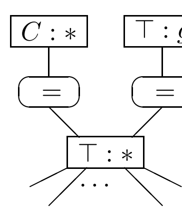
Corollary 11.4. in [Da02] yields:



Corollary 11.3. in [Da02] yields:



Lemma 11.5. in [Da02] yields:



□

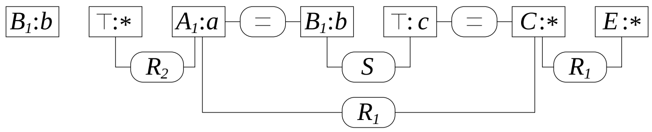
Now we are prepared to prove the syntactical equivalence of $\mathfrak{G}$ and $\mathfrak{G}_{\mathcal{M}^{\mathfrak{G}}}$.

Theorem 2 ($\mathfrak{G}$ and $\mathfrak{G}_{\mathcal{M}^{\mathfrak{G}}}$ are Syntactically Equivalent).
Let $\mathfrak{G}$ be a concept graph. Then we have

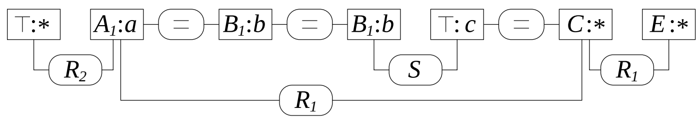
$$\mathfrak{G} \vdash_{pos} \mathfrak{G}_{\mathcal{M}^{\mathfrak{G}}} \quad \text{and} \quad \mathfrak{G}_{\mathcal{M}^{\mathfrak{G}}} \vdash_{pos} \mathfrak{G}\ .$$

Proof:

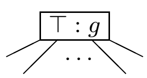
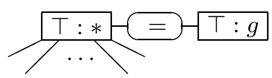
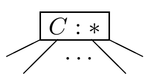
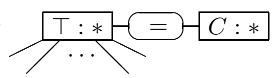
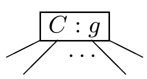
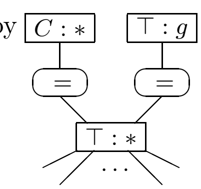
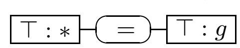
We will exemplify the proof with the example for standard graphs above, that is, we start with



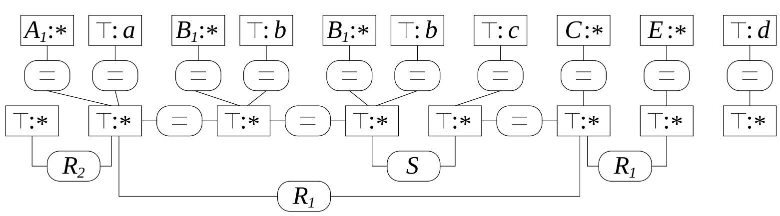
Let $[v]\theta_{\mathfrak{G}} \cap V = \{v_1, \ldots, v_n\}$. With the rule 'identity-insertion' and with Lemma 11.5 of [Da02], we can add or remove identity links such that there is an identity link between $v_i, v_j \in V \cap [v]\theta_{\mathfrak{G}}$ iff $j = i + 1$. One possible result for our example is:



Now we do the following:

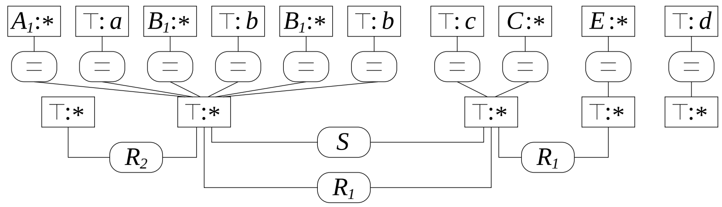
1. Each vertex  is replaced by 
2. Each vertex  ($C \neq \top$) is replaced by 
3. Each vertex  is replaced with Lemma 4 by 
4. For each $G \in \mathcal{G}$ with $g \notin \rho[V]$ we add 

For our example, we get:

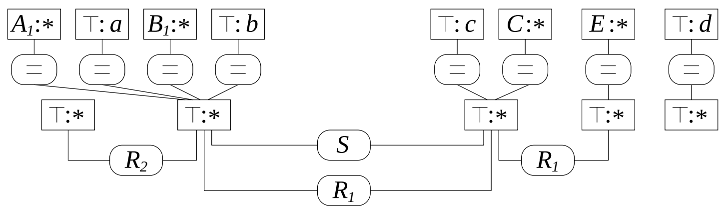


For each class $[v]\theta_{\mathfrak{G}} := \{v_1, \ldots, v_n\}$, we merge v_1 into v_2, v_2 into v_3, …, v_{n-1} into v_n. After this step, each class $[v]\theta_{\mathfrak{G}}$ corresponds to exactly one vertex in the graph we have constructed so far. Let W be the set of these vertices.

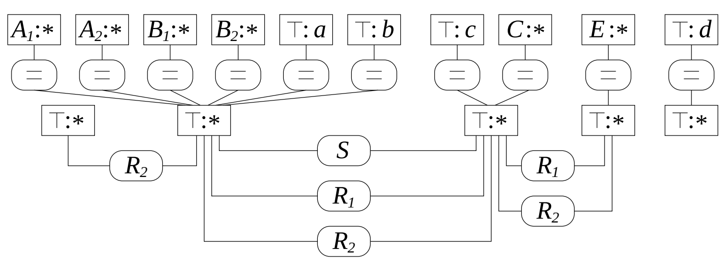
In our example, we get:



Now we erase all repeated instances of structures $-(=)-\boxed{C : *}$ and of structures $-(=)-\boxed{\top : g}$ which are linked to the same vertex $w = \boxed{\top : *}$ with $w \in W$. Analogously, all repeated instances of edges are erased. The opposite direction can be carried out with the iteration-rule.



For each vertex $w \in W$, we now do the following: If a structure $-(=)-\boxed{C : g}$ is linked to w and if $C < D < \top$, we add a structure $-(=)-\boxed{D : g}$ which is linked to w as well (supposed the new structure did not exist already). Analogously, if we have an edge e which is incident with vertices from W and which is labelled with the relation name $R := \kappa(e)$, and if S is an relation name with $R < S$, we add an edge f with $\kappa(f) = S$ and $f|_1 = e|_1, \ldots, f|_k = e|_k$ (with $|e| = |f| = k$), supposed an edge like this did not exists already. This can be done with an application the iteration- and of the generalization-rule. The opposite direction can be carried out with the erasure-rule.



It is easy to see that the resulting graph is $\mathfrak{G}_{\mathcal{M}^{\mathfrak{G}}}$. As all steps in the proof can be carried out in both directions, we are done. □

The class of all graphs over a given alphabet $\mathcal{A}$, together with the semantical entailment relation $\models$ is a quasiorder. The same holds for the class of all models. With the last theorem, we are now prepared to show that these quasiorders are isomorphic structures. More precisely, we have the following corollary:

Corollary 2 ((CS,$\models$) and (CG,$\models$) are isomorphic quasiorders).
The mappings $\mathcal{M} \mapsto \mathfrak{G}_{\mathcal{M}}$ and $\mathfrak{G} \mapsto \mathcal{M}^{\mathfrak{G}}$ are, up to equivalence, mutually inverse isomorphisms between the quasiordered sets (CS,$\models$) and (CG,$\models$).

Proof: As we know that $\vdash_{pos}$ is sound, the last theorem yields that $\mathfrak{G} \models \mathfrak{G}_{\mathcal{M}^{\mathfrak{G}}}$ and $\mathfrak{G}_{\mathcal{M}^{\mathfrak{G}}} \models \mathfrak{G}$ hold as well. We have furthermore

$$\mathcal{M}_1 \models \mathcal{M}_2 \overset{\text{L. 3}}{\Longleftrightarrow} \mathcal{M}^{\mathfrak{G}_{\mathcal{M}_1}} \models \mathcal{M}^{\mathfrak{G}_{\mathcal{M}_2}} \overset{\text{C. 1}}{\Longleftrightarrow} \mathfrak{G}_{\mathcal{M}_1} \models \mathfrak{G}_{\mathcal{M}_2} .$$

These results together with Lemma 3 and Corollary 1 yield this corollary. □

5 Transformation-Rules for Models

We still have to show that $\vdash_{pos}$ is complete. Although we have the last corollary, this cannot be derived from the results we have so far.

In order to prove the completeness of $\vdash_{pos}$, we will introduce four transformation rules for models:[10] *removing an element, doubling an element, exchanging attributes, and restricting the incidence relations.* This rules form a calculus for models, which will be denoted by $\vdash$. We will show that $\vdash_{pos}$ for graphs and $\vdash$ for models are complete. The main idea is the following: If we have two models $\mathcal{M}^a$ and $\mathcal{M}^b$ with $\mathcal{M}^a \models \mathcal{M}^b$, we will show that $\mathcal{M}^a$ can successively transformed to $\mathcal{M}^b$ with the rules for the models, and each transformation carries over to the standard graphs, i. e. we get simultaneous $\mathfrak{G}_{\mathcal{M}^a} \vdash_{pos} \mathfrak{G}_{\mathcal{M}^b}$. This is enough to prove the completeness of $\vdash_{pos}$ as well.

In the following, we will define the transformation rules for models and show that each rule is sound in the system of models, and that it carries over to the set of graphs, together with the calculus $\vdash_{pos}$.

Definition 10 (Removing an Element from a Contextual Structure).
Let $\mathcal{M} := (\vec{\mathbb{K}}, \lambda)$ be a contextual $\mathcal{A}$-structure and let $g \in G_0 \backslash \lambda_{\mathcal{G}}[\mathcal{G}]$. We define a power context family $\vec{\mathbb{K}}'$ as follows:

1. $G'_0 = G_0 \backslash \{g\}$, $G'_i = G_i \cap (G'_0)^i$ *for all* $i \geq 1$,
2. $M'_i = M_i$ *for all* i,
3. $I'_i = I_i \cap (G'_i \times M_i)$ *for all* i.

For the contextual structure $\mathcal{M} := (\vec{\mathbb{K}}', \lambda)$, we say that $\mathcal{M}'$ is obtained from $\mathcal{M}$ by removing the element g.

Lemma 5 (Removing an Element).
Let $\mathcal{M} := (\vec{\mathbb{K}}, \lambda)$ be a model, let $\mathcal{M}' := (\vec{\mathbb{K}}', \lambda')$ be obtained from $\mathcal{M}$ by removing the element $g \in G_0$. Let $\mathcal{M}^a := (\vec{\mathbb{K}}^a, \lambda^a)$ be a model, let $f : G^a_0 \to G_0$ with $g \notin f[G^a_0]$ and $\mathcal{M} \models_f \mathcal{M}_a$, and $id : G_0 \backslash \{g\} \to G_0$ the identity-mapping. Then we have $\mathcal{M}' \models_f \mathcal{M}^a$, $\mathcal{M} \models_{id} \mathcal{M}'$ and $\mathfrak{G}_{\mathcal{M}} \vdash_{pos} \mathfrak{G}_{\mathcal{M}'}$

[10] We have the implicit agreement that isomorphic models are identified. Isomorphism between power context families and between models is defined as usual.

Proof:

It is easy to check that we have $\mathcal{M}' \models_f \mathcal{M}^a$ and $\mathcal{M} \models_{id} \mathcal{M}'$.

With the denotations from Definition 9), $\mathfrak{G}_{\mathcal{M}'}$ can be derived from $\mathfrak{G}_{\mathcal{M}}$ by easing all edges v_g is incident with, by erasing all vertices and edges $v_{g,G}$ and $e_{g,G}$ with $G \in \mathcal{G}$ and $\lambda_{\mathcal{G}}(G) = g$, by erasing all vertices and edges $v_{g,C}$ and $e_{g,C}$ with $C \in \mathcal{C}$ and $g \in \mathrm{Ext}(\lambda_{\mathcal{C}}(C))$, and by erasing v_g. □

Definition 11 (Doubling of an Element in a Contextual Structure).

Let $\mathcal{M} := (\vec{\mathbb{K}}, \lambda)$ be a contextual $\mathcal{A}$-structure and let $g, g' \in G_0$. For each tuple $\vec{h} = (g_1, \ldots, g_i) \in G_i$, we set $\vec{h}[g'/g] := (g'_1, \ldots, g'_i)$ with $g'_j := \begin{cases} g_j & g_j \neq g \\ g' & g_j = g \end{cases}$.

If $\mathcal{M}' := (\vec{\mathbb{K}}', \lambda')$ is a contextual structure over $\mathcal{A}$ such that there is a $g \in G_0$ with

1. *$G'_0 = G_0 \dot\cup \{g'\}$ for a $g' \notin G_0$ and $G'_i = G_i \dot\cup \{\vec{h}[g'/g] \mid \vec{h} \in G_i\}$ for all $i \geq 1$,*
2. *$M'_i = M_i$ for all i,*
3. *$I'_0 = I_0 \dot\cup \{(g', m) \mid g I_0 m\}$, and $I'_i = I_i \dot\cup \{(\vec{h}[g'/g], m) \mid \vec{h} I_i m\}$ for all $i \geq 1$,*
4. *$\lambda'_{\mathcal{G}}$ fulfills $\lambda'_{\mathcal{G}}(G) = \lambda_{\mathcal{G}}(G)$ for all $G \in \mathcal{G}$ with $\lambda_{\mathcal{G}}(G) \neq G$, and for all $G \in \mathcal{G}$ with $\lambda_{\mathcal{G}}(G) = G$ we have $\lambda'_{\mathcal{G}}(G) \in \{g, g'\}$,*
5. *$\lambda'_{\mathcal{C}}(C) = (\, (Int(\lambda_{\mathcal{C}}(C))^{I'_0}, Int(\lambda_{\mathcal{C}}(C))\,)$ for all $C \in \mathcal{C}$, and*
6. *$\lambda'_{\mathcal{R}}(R) = (\, (Int(\lambda_{\mathcal{R}}(R))^{I'_i}, Int(\lambda_{\mathcal{R}}(R))\,)$ for all $R \in \mathcal{R}_i$,*

then we say that $\mathcal{M}'$ is obtained from $\mathcal{M}$ by doubling the element *g.*

As the definition of *doubling an element* is fairly technical, we provide an example for this rule. Let $\mathcal{A} := (\{a, b, c\}, \{A, B, \top\}, \{R, \dot=\})$, where R is a 4-ary relation name, and let the following contextual structure $\mathcal{M}$ over $\mathcal{A}$ be given (we have added an additional column to show how the mapping $\lambda_{\mathcal{G}}$ assigns object names to objects):

$\lambda_{\mathcal{G}}$
a, b
c

$\mathbb{K}_0$	A	B	C	$\top$
g	×	×		×
h			×	×

$\mathbb{K}_2$	$\dot=$
(g, g)	×
(h, h)	×

$\mathbb{K}_4$	R
(g, h, g, h)	×

.

Then the following contextual structure is one of three possible structures which can be obtained from $\mathcal{M}$ by doubling the element g:

$\lambda_{\mathcal{G}}$
a
b
c

$\mathbb{K}_0$	A	B	C	$\top$
g	×	×		×
g'	×	×		×
h			×	×

$\mathbb{K}_2$	$\dot=$
(g, g)	×
(g', g')	×
(h, h)	×

$\mathbb{K}_4$	R
(g, h, g, h)	×
(g', h, g', h)	×

.

As we loose the information $\lambda_{\mathcal{G}}(a) = \lambda_{\mathcal{G}}(b)$, we see that this contextual structure contains less information than $\mathcal{M}$.

Furthermore we note that if $(\vec{\mathbb{K}}', \lambda'), (\vec{\mathbb{K}}'', \lambda'')$ are two contextual structures which are obtained from a contextual structure $(\vec{\mathbb{K}}, \lambda)$ by doubling the element $g \in G_0$, then they can (up to isomorphism) only differ between $\lambda'_{\mathcal{G}}$ and $\lambda''_{\mathcal{G}}$.

Lemma 6 (Doubling an Element).

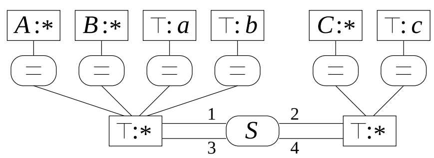
Let $\mathcal{M} := (\vec{\mathbb{K}}, \lambda)$ be a model and let $\mathcal{M}' := (\vec{\mathbb{K}}', \lambda')$ be obtained from $\mathcal{M}$ by doubling the element $g \in G_0$. Then we have $\mathcal{M} \models \mathcal{M}'$ and $\mathfrak{G}_{\mathcal{M}} \vdash_{pos} \mathfrak{G}_{\mathcal{M}'}$ .If we have a model $\mathcal{M}^a := (\vec{\mathbb{K}}^a, \lambda^a)$ with $\mathcal{M} \models \mathcal{M}_a$, then we can chose $\mathcal{M}'$ such that we have $\mathcal{M}' \models \mathcal{M}^a$.

Proof:

Let $g' \in G'_0$ be the new element which is obtained from doubling $g \in G_0$, i. e. $G'_0 = G_0 \dot{\cup} \{g'\}$. It is easy to see that $f' : G'_0 \to G_0$ with $f'|_{G_0} = id$ and $f'(g') = g$ fulfills all conditions of Definition 8, i. e. we have $\mathcal{M} \models_{f'} \mathcal{M}'$.

Now let $f_a : G^a_0 \to G_0$ be a mapping with $\mathcal{M} \models_f \mathcal{M}^a$. Let f'_a be an arbitrary mapping with $f'_a(h) = f_a(h)$, if $f_a(h) \neq g$, and $f'_a(h) \in \{g, g'\}$, if $f_a(h) = g$. Now condition 4. of Definition 11 enables us to chose $\lambda'_{\mathcal{G}}$ from $\mathcal{M}'$ such that condition 1. of Definition 8 is satisfied. Conditions 2. and 3. of Definition 8 are trivially satisfied. So we have $\mathcal{M}' \models_{f'_a} \mathcal{M}^a$.

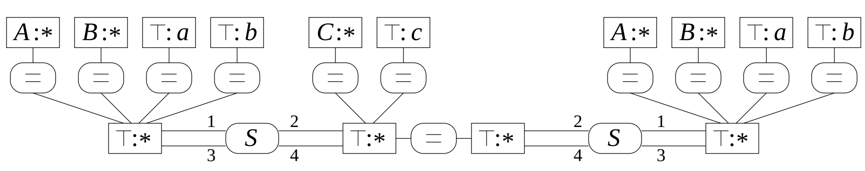
It remains to show that $\mathfrak{G}_{\mathcal{M}} \vdash_{pos} \mathfrak{G}_{\mathcal{M}'}$ holds. We will exemplify the proof with the example after Definition 11, that is, we start with



We consider the subgraph which contains the following vertices and edges (we use the denotation of Definition 9):

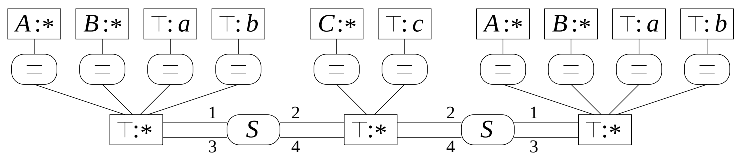
1. v_g, all $v_{g,G}$ and $e_{g,G}$ (with $G \in \mathcal{G}$ and $\lambda_{\mathcal{G}}(G) = g$), all $v_{g,C}$ and $e_{g,C}$ (with $C \in \mathcal{C}$ and $g \in \text{Ext}(\lambda_{\mathcal{C}}(C))$.
2. All edges $e_{g_1,\ldots,g_i,R}$ such that $g_j = g$ for one j (with $R \in \mathcal{R}$ and $(g_1, \ldots, g_i) \in \text{Ext}(\lambda_{\mathcal{R}}(R))$). The set of these edges shall be denoted by F.
3. All vertices v_h which are incident with an edge $e \in F$. The set of these vertices shall be denoted by W.

This subgraph is iterated, and an new identity link is inserted between w and its copy for each $w \in W$.[11] The copies of the vertices $v_{g,G}$ and edges $e_{g,G}$ will be denoted by $v'_{g,G}$ and $e'_{g,G}$, respectively. For our example, we get:

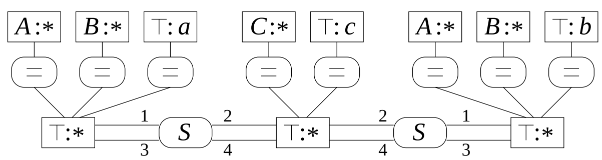


[11] Using the technical implementation of the iteration-rule in Definition 11.1. of [Da02], we insert an identity link between $(w, 1)$ and $(w, 2)$.

For every $w \in W$, the copy of w is merged 'back' into w. For our example, we get:



For g, we erase all vertices $v_{g,G}$ and edges $e_{g,G}$, where $\lambda'_{\mathcal{G}}(G) \neq g$. Analogously for g', we erase all vertices $v'_{g,G}$ and edges $e'_{g,G}$, where $\lambda'_{\mathcal{G}}(G) \neq g'$. For our example, we obtain the following graph:



This is $\mathfrak{G}_{\mathcal{M}'}$. □

Definition 12 (Restricting the Incidence Relations).

Let $\mathcal{M} := (\vec{\mathbb{K}}, \lambda)$ *be a contextual* $\mathcal{A}$*-structure. If* $\mathcal{M}' := (\vec{\mathbb{K}}', \lambda)$ *is a contextual structure over* $\mathcal{A}$ *with*

$$G'_i = G_i \text{ for all } i, \quad M'_i = M_i \text{ for all } i, \quad \text{and } I'_i \subseteq I_i \text{ for all } i \, ,$$

then we say that $\mathcal{M}'$ *is obtained from* $\mathcal{M}$ *by* restricting the incidence relations.

Lemma 7 (Restricting the Incidence Relations).

If $\mathcal{M}' := (\vec{\mathbb{K}}', \lambda)$ *is obtained from* $\mathcal{M} := (\vec{\mathbb{K}}, \lambda)$ *by restriction the incidence relations, we have* $\mathcal{M} \models_{id} \mathcal{M}'$ *and* $\mathfrak{G}_{\mathcal{M}} \vdash_{pos} \mathfrak{G}_{\mathcal{M}'}$.

Proof:

It is easy to see that $\mathfrak{G}_{\mathcal{M}'}$ is a subgraph of $\mathfrak{G}_{\mathcal{M}}$, hence $\mathfrak{G}_{\mathcal{M}'}$ can be derived from $\mathfrak{G}_{\mathcal{M}}$ by erasing all edges and vertices which are not in $\mathfrak{G}_{\mathcal{M}'}$. □

Definition 13 (Exchanging Attributes and Standardization).

Let $\mathcal{M} := (\vec{\mathbb{K}}, \lambda)$ *be a contextual* $\mathcal{A}$*-structure. If* $\mathcal{M}' := (\vec{\mathbb{K}}', \lambda')$ *is a contextual* $\mathcal{A}$*-structure which satisfies*

1. $G'_i := G_i$ *for all* i,
2. $gI_0\lambda_{\mathcal{C}}(C) \Longleftrightarrow gI'_0\lambda'_{\mathcal{C}}(C)$ *for all* $g \in G_0$ *and* $C \in \mathcal{C}$,
3. $\vec{g}I_i\lambda_{\mathcal{R}}(R) \Longleftrightarrow \vec{g}I'_i\lambda'_{\mathcal{C}}(R)$*, for all* $i \geq 1$, $\vec{g} \in G_i$ *and* $R \in \mathcal{R}_i$*, and*
4. $\lambda'_{\mathcal{G}} := \lambda_{\mathcal{G}}$,

then we say that $\mathcal{M}'$ *is obtained from* $\mathcal{M}$ by exchanging attributes of $\mathcal{M}$. *If* $\mathcal{M}'$ *additionally satisfies* $M'_0 := \mathcal{C}$ *and* $M'_i := \mathcal{R}_i$ *for all* $i \geq 1$*, then we say that* $\mathcal{M}'$ *is obtained from* $\mathcal{M}$ *by* standardization of $\mathcal{M}$.

This rule is the only rule which does not weaken the informational content of a model. In particular, it can be carried out in both directions.

Lemma 8 (Exchanging Attributes and Standardization).

If $\mathcal{M}' := (\vec{\mathbb{K}}', \lambda')$ is obtained from $\mathcal{M} := (\vec{\mathbb{K}}, \lambda)$ by exchanging attributes, we have $\mathcal{M} \models_{id} \mathcal{M}'$, $\mathcal{M}' \models_{id} \mathcal{M}$, and $\mathfrak{G}_{\mathcal{M}} = \mathfrak{G}_{\mathcal{M}'}$ .Furthermore exists a standardization of $\mathcal{M}$ for each contextual structure $\mathcal{M}$.

Proof: Trivial

The four rules form a calculus for models, i. e. we have the following definition:

Definition 14 (Calculus for Contextual Structures).

The calculus for contextual structures over the alphabet $\mathcal{A} := (\mathcal{G}, \mathcal{C}, \mathcal{R})$ consists of the following rules:

Removing an element, doubling an element, exchanging attributes, and restricting the incidence relations.

If $\mathcal{M}^a$, $\mathcal{M}^b$ are two models, and if there is a sequence $(\mathcal{M}^1, \mathcal{M}^2, \dots, \mathcal{M}^n)$ with $\mathcal{M}^1 = \mathcal{M}^a$ and $\mathcal{M}^b = \mathcal{M}^n$ such that each $\mathcal{M}^{i+1}$ is derived from $\mathcal{M}^i$ by applying one of the rules of the calculus, we say that $\mathcal{M}^b$ can be derived from *$\mathcal{M}^a$, which is denoted by $\mathcal{M}^a \vdash \mathcal{M}^b$.*

Now we are prepared to show that the transformation rules for models are complete and respected by the construction of standard graphs.

Theorem 3 ($\vdash$ is Complete and Respected by Standard Graphs).

Let $\mathcal{M}^a := (\vec{\mathbb{K}}^a, \lambda^a)$, $\mathcal{M}^b := (\vec{\mathbb{K}}^b, \lambda^b)$ be two contextual structures such that $\mathcal{M}^a \models \mathcal{M}^b$. Then we have $\mathcal{M}^a \vdash \mathcal{M}^b$ and $\mathfrak{G}_{\mathcal{M}^a} \vdash_{pos} \mathfrak{G}_{\mathcal{M}^b}$.

Proof:

First Lemma 8 allows us to assume w.l.o.g. that $\mathcal{M}^a$ and $\mathcal{M}^b$ are standardized, i. e. that we have $M_0{}^a = M_0{}^b = \mathcal{C}$ and $M_i{}^a = M_i{}^b = \mathcal{R}_i$ for all $i \geq 1$.

Let $f : G_0{}^b \to G_0{}^a$ with $\mathcal{M}^a \models_f \mathcal{M}^b$.

Assume that f is not injective, i. e. there are $g_1, g_2 \in G_0{}^b$ with $f(g_1) = f(g_2)$. Then we can double $f(g_1) \in G_0{}^a$ to obtain from $\mathcal{M}^a$ a contextual model $\mathcal{M}^c$ with a new element h. Similar to the proof of Lemma 6, we can choose $\lambda^c_{\mathcal{C}}$ such that the a mapping $f' : G_0{}^b \to G_0{}^c$ with $f'|_{G_0{}^b \backslash \{g_2\}} = f$ and $f(g_2) = h$ which fulfills $\mathcal{M}^c \models_{f'} \mathcal{M}$. If f' is not injective again, we repeat this step as often as necessary until we finally obtain a contextual structure $\mathcal{M}^1$ and an injective mapping $f_1 : G_0{}^b \to G_0{}^1$ with $\mathcal{M}^1 \models_{f_1} \mathcal{M}^b$. As $\mathcal{M}^1$ is obtained from $\mathcal{M}^a$ by doubling several elements, we have $\mathcal{M}^a \vdash \mathcal{M}^1$ by definition of $\vdash$, and Lemma 6 yields $\mathfrak{G}_{\mathcal{M}^a} \vdash_{pos} \mathfrak{G}_{\mathcal{M}^1}$.

If f_1 is not surjective, we can remove with the rule rule 'removing an element' gradually all objects $g \in G_0{}^1 \backslash f_1[G_0{}^b]$ from $\mathcal{M}^1$ to obtain from $\mathcal{M}^1$ a contextual structure $\mathcal{M}^2$ with $\mathcal{M}^1 \vdash \mathcal{M}^2$. Lemma 5 yields $\mathfrak{G}_{\mathcal{M}^1} \vdash_{pos} \mathfrak{G}_{\mathcal{M}^2}$ and $\mathcal{M}^2 \models_{f_1} \mathcal{M}^b$. Furthermore we now have that f_2 is bijective.

It is clear that isomorphic contextual structures yield isomorphic standard graphs, so we can finally assume that f is the identity-mapping, in particular we have $G_0^2 = G_0^b$ (remember that have the implicit agreement that isomorphic models are identified).

Conditions 1.–3. of Definition 8, which are satisfied by id, can now be stated as follows:

1. For all $G \in \mathcal{G}$ we have $\lambda_{\mathcal{G}}^b(G) = \lambda_{\mathcal{G}}^a(G)$,
2. $\mathcal{C} = {M_0}^2 = {M_0}^b$, and for all $C \in \mathcal{C}$ we have $C^{{I_0}^2} \subseteq C^{{I_0}^b}$, and
3. for $i \geq 1$, we have $\mathcal{R}_i = {M_i}^2 = {M_i}^b$, and for all $R \in \mathcal{R}_i$ we have $R^{{I_i}^2} \subseteq R^{{I_i}^b}$.

Now it is easy to see that $\mathcal{M}^b$ can be obtained from $\mathcal{M}^2$ by restricting the incidence relations, thus Lemma 7 yields $\mathcal{M}^2 \vdash \mathcal{M}^b$ and $\mathfrak{G}_{\mathcal{M}^2} \vdash_{pos} \mathfrak{G}_{\mathcal{M}^b}$.

As $\vdash_{pos}$ (for graphs) and $\vdash$ (for models) are transitive, we conclude $\mathcal{M}^a \vdash \mathcal{M}^b$ and $\mathfrak{G}_{\mathcal{M}^a} \vdash_{pos} \mathfrak{G}_{\mathcal{M}^b}$. □ This theorem yields that the calculus $\vdash_{pos}$ on the graphs is complete as well:

Corollary 3 (Both Calculi are Complete).
Let $\mathcal{M}_1, \mathcal{M}_2$ be models and $\mathfrak{G}_1, \mathfrak{G}_2$ be graphs. We have:
$\mathcal{M}_1 \models \mathcal{M}_2 \iff \mathcal{M}_1 \vdash \mathcal{M}_2$ *and* $\mathfrak{G}_1 \models \mathfrak{G}_2 \iff \mathfrak{G}_1 \vdash_{pos} \mathfrak{G}_2$.

Proof:
The direction '⇐' of the first equivalence follows immediately from Lemmata 6, 5, 8, and 7, and the direction '⇒' is a part of Theorem 3.

The direction '⇐' of the second equivalence is already proven in [Da02], so it remains to show '⇒'. We have:

$$\mathfrak{G}_1 \models \mathfrak{G}_2 \overset{\text{C. 1}}{\iff} \mathcal{M}^{\mathfrak{G}_1} \models \mathcal{M}^{\mathfrak{G}_2} \overset{\text{C. 2}}{\implies} \mathfrak{G}_{\mathcal{M}^{\mathfrak{G}_1}} \models \mathfrak{G}_{\mathcal{M}^{\mathfrak{G}_2}} \overset{\text{T. 2}}{\iff} \mathfrak{G}_1 \models \mathfrak{G}_2 \ ,$$

thus we are done. □

6 Conclusion

In Prediger ([Pr98a]), the notion of standard models is adaquate only for concept graphs without generic markers. For concept graphs with generic markers, a standard model of a graph may encode less information than the graph. Thus, in this case, the reasoning on concept graphs cannot be carried over to the models completely. This is a gap we have bridged with our notion of standard models, which extends Predigers approach. Moreover, reasoning on models can be carried out in two different ways: By the semantical entailment relation $\models$ on models (see Definition 8), and by transformation rules between models (see Section 5).

In [Da02], an adaquate calculus for concept graphs with cuts is provided. Thus, if we have two concept graphs without cuts $\mathfrak{G}_a$ and $\mathfrak{G}_b$ with $\mathfrak{G}_a \models \mathfrak{G}_b$, we have a proof for $\mathfrak{G}_a \vdash \mathfrak{G}_b$, that is: We have a sequence $(\mathfrak{G}_1, \mathfrak{G}_2, \ldots, \mathfrak{G}_n)$ with $\mathfrak{G}_1 = \mathfrak{G}_a$, $\mathfrak{G}_b = \mathfrak{G}_n$ such that each $\mathfrak{G}_{i+1}$ is derived from $\mathfrak{G}_i$ by applying one of the rules of the calculus. But now we have even more: From Corollary 3, we conclude that we can find a proof $(\mathfrak{G}_1, \mathfrak{G}_2, \ldots, \mathfrak{G}_n)$ such that *all* graphs $\mathfrak{G}_1, \ldots, \mathfrak{G}_n$ are concept graphs without cuts. This result cannot be directly derived from [Da02].

In this sense, this work is a completion of both [Da02]) and [Pr98a].

References

[Da02] F. Dau: *The Logic System of Concept Graphs with Negations and its Relationship to Predicate Logic*. PhD-Thesis, Darmstadt University of Technology, 2002. To appear in Springer Lecture Notes on Computer Science.

[CM92] M. Chein, M.-L. Mugnier: *Conceptual Graphs: Fundamental Notions.* Revue d'Intelligence Artificielle 6, 1992, 365–406.

[Mu00] M.-L. Mugnier: *Knowledge Representation and Reasonings Based on Graph Homomophism.* In: B. Ganter, G. W. Mineau (Eds.): Conceptual Structures: Logical, Linguistic and Computational Issues. LNAI 1867, Springer Verlag, Berlin–New York 2000, 172–192.

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

[Pr98a] S. Prediger: *Kontextuelle Urteilslogik mit Begriffsgraphen. Ein Beitrag zur Restrukturierung der mathematischen Logik.* Aachen, Shaker Verlag 1998.

[Pr98b] S. Prediger: *Simple Concept Graphs: A Logic Approach.* In: M. -L. Mugnier, M. Chein (Eds.): Conceptual Structures: Theory, Tools and Applications. LNAI 1453, Springer Verlag, Berlin–New York 1998, 225–239.

[Sh99] S. J. Shin: *The Iconic Logic of Peirce's Graphs.* Bradford Book, Massachusetts, 2002.

[So84] J. F. Sowa: *Conceptual Structures: Information Processing in Mind and Machine.* Addison Wesley Publishing Company Reading, 1984.

[So00] J. F. Sowa: *Knowledge Representation: Logical, Philosophical, and Computational Foundations.* Brooks Cole Publishing Co., Pacific Grove, CA, 2000.

[Wi97] R. Wille: *Conceptual Graphs and Formal Concept Analysis.* In: D. Lukose et al. (Eds.): Conceptual Structures: Fulfilling Peirce's Dream. LNAI 1257, Springer Verlag, Berlin–New York 1997, 290–303.

[Wi02] R. Wille: *Existential Concept Graphs of Power Context Families.* In: U. Priss, D. Corbett, G. Angelova (Eds.): Conceptual Structures: Integration and Interfaces. LNAI 2393, Springer Verlag, Berlin–New York 2002, 382–396.