Journal of Universal Computer Science, vol. 11, no. 12 (2005), 2063-2075
submitted: 5/10/05, accepted: 15/11/05, appeared: 28/12/05 © J.UCS

Functional Dependencies with Counting on Trees[1]

Klaus-Dieter Schewe
(Massey University, Department of Information Systems & Information Science
Research Centre, Private Bag 11 222, Palmerston North, New Zealand
k.d.schewe@massey.ac.nz)

Abstract: The paper presents an axiomatisation for functional dependencies on trees that are defined using constructors for records, lists, sets and multisets. A simple form of restructuring permitting lists to be mapped onto multisets and multisets onto sets is added to the theory. Furthermore, the theory handles dependencies on sets treated as multisets. This adds the possibility to use the count of elements in the dependencies.

Key Words: functional dependencies, complex value databases, counting attributes, axiomatisation

Category: F.4.1, H.2.1

1 Introduction

Dependency theory is an important branch of database theory [Abiteboul et al., 1995]. In the context of the relational data model around 90 classes of dependencies have been investigated, and most problems (except the real hard ones) have been solved. Thalheim in [Thalheim, 1991] gives a good account on relational dependency theory, but for more than a decade not much more database research has been devoted to dependency theory.

Recently, there has been a revived interest in dependency theory for post-relational databasess, in particular for HERM [Thalheim, 2000] and XML [Abiteboul et al., 2000]. The work in [Arenas and Libkin, 2004], [Vincent et al., 2004] and [Hartmann et al., 2006] considers functional dependencies (FDs) on trees. Usually, FDs are considered to be the simplest class of dependencies, though in the non-relational theories they can no longer be considered simple. While the first two cited papers investigate paths in XML trees, the third approach exploits constructors for sets, lists and multisets, and then investigates Brouwerian algebras of subattributes in order to axiomatise FDs. All three lines of research lead to different, not yet unified theories, though the work in [Wang and Topor, 2005] tried to create a class of FDs that subsumes the other existing definitions.

The work in this paper continues the line of research in [Hartmann et al., 2006], and further investigates the axiomatisation problem. The finite axiomatisation has been extended in [Sali and Schewe, 2006] to cover also a disjoint union constructor. However, so called "counter subattributes" that permit counting the

[1] C. S. Calude, H. Ishihara (eds.). *Constructivity, Computability, and Logic. A Collection of Papers in Honour of the 60th Birthday of Douglas Bridges.*

elements in a list or multiset or indicating, whether sets are empty or not, respectively, had to be excluded. In fact, as shown in [Sali and Schewe, 2005] it is shown that taking these subattributes into account leads to non-existence of finite axiomatisation. However, if the class of FDs is extended to weak functional dependencies, i.e. disjunctions of FDs, there is again an axiomatisation for the price of a significant increase in the complexity of the completeness proof. The reason for this complexity is that the union constructor induces non-trivial equivalences between subattributes. For instance, a set attribute with a union attribute for its elements can be identified with a record attribute with set attributes in each component.

As remarked in [Sali and Schewe, 2005] it may well be possible to add deliberately further subattribute relationships, e.g. a set attribute can be considered a subattribute of a multiset attribute, which can be considered a subattribute of a list attribute. Furthermore, a set value can always be considered as a multiset value, which permits to apply FDs on multiset attributes to set values. This gives rise to FDs with counting, as counting the elements in sets would be enabled.

In this paper we present an axiomatisation of FDs with counting, but for sake of not repeating the complex technical proof work in [Sali and Schewe, 2005] we exclude the union constructor. We repeat the preliminaries of our model of nested attributes and subattributes in Section 2, to which we add the new restructuring rules. In Section 3 we discuss FDs and the extensions regarding counting. Then we present a set of derivation rules and proof their soundness and completeness.

2 Preliminaries

Let $\mathcal{U}$ be a finite set, the elements of which we will call *simple attributes*. Further assume that for each $A \in \mathcal{U}$ we are given a countably infinite set $dom(A)$, which we call the *domain* of A or the *sets of values* of attribute A. Take another set $\mathcal{L}$ of *labels* with $\mathcal{U} \cap \mathcal{L} = \emptyset$ and assume that the symbol λ is neither a simple attribute nor a label, i.e. $\lambda \notin \mathcal{U} \cup \mathcal{L}$.

Definition 1. The set $\mathcal{N}$ of *(nested) attributes* over $\mathcal{U}$ and $\mathcal{L}$ is the smallest set with $\lambda \in \mathcal{N}$, $\mathcal{U} \subseteq \mathcal{N}$, and satisfying the following properties:

- for $X \in \mathcal{L}$ and $X'_1, \dots, X'_n \in \mathcal{N}$ we have $X(X'_1, \dots, X'_n) \in \mathcal{N}$;
- for $X \in \mathcal{L}$ and $X' \in \mathcal{N}$ we have $X\{X'\} \in \mathcal{N}$, $X[X'] \in \mathcal{N}$, and $X\langle X' \rangle \in \mathcal{N}$.

We call λ a *null attribute*, $X(X'_1, \dots, X'_n)$ a *record attribute*, $X\{X'\}$ a *set attribute*, $X[X']$ a *list attribute*, and $X\langle X' \rangle$ a *multiset attribute*. We can then extend the association *dom* from simple to nested attributes.

Definition 2. For each nested attribute $X \in \mathcal{N}$ we get a *domain* $dom(X)$ as follows:

- $dom(\lambda) = \{\top\}$;
- $dom(X(X'_1, \dots, X'_n)) = \{(v_1, \dots, v_n) \mid v_i \in dom(X'_i) \text{ for } i = 1, \dots, n\}$;
- $dom(X\{X'\}) = \{\{v_1, \dots, v_k\} \mid k \in \mathbb{N} \text{ and } v_i \in dom(X') \text{ for } i = 1, \dots, k\}$, i.e. each element in $dom(X\{X'\})$ is a finite set with (pairwise different) elements in $dom(X')$;
- $dom(X[X']) = \{[v_1, \dots, v_k] \mid k \in \mathbb{N} \text{ and } v_i \in dom(X') \text{ for } i = 1, \dots, k\}$, i.e. each element in $dom(X[X'])$ is a finite (ordered) list with (not necessarily different) elements in $dom(X')$;
- $dom(X\langle X'\rangle) = \{\langle v_1, \dots, v_k\rangle \mid k \in \mathbb{N} \text{ and } v_i \in dom(X') \text{ for } i = 1, \dots, k\}$, i.e. each element in $dom(X\langle X'\rangle)$ is a finite multiset with elements in $dom(X')$, or in other words each $v \in dom(X')$ has a *multiplicity* $m(v) \in \mathbb{N}$ in a value in $dom(X\langle X'\rangle)$.

In order to define functional dependencies on a nested attribute $X \in \mathcal{N}$ it is crucial that we can define projection mappings on $dom(X)$. For this we use the notion of *subattribute*, i.e. we define a partial order $\geq$ on nested attributes in such a way that whenever $X \geq Y$ holds, we obtain a canonical projection $\pi^X_Y : dom(X) \to dom(Y)$. However, this partial order has to be defined on equivalence classes of attributes, as some domains may be identified.

Equivalence of attributes is simply induced by the rule that order in record attributes is not important, i.e. $X(X'_1, \dots, X'_n) \equiv X(X'_{\sigma(1)}, \dots, X'_{\sigma(n)})$ holds for any permutation $\sigma \in \mathbf{S}_n$, λ can be added or removed in record attributes, and $\lambda \equiv X()$ and $X[\lambda] \equiv X\langle\lambda\rangle$ hold. Then, whenever an attribute X' appearing in the definition of another nested attribute X is replaced by an equivalent one, the result is equivalent to X – for further formal details see [Sali and Schewe, 2006]. In the following we identify $\mathcal{N}$ with the set of equivalence classes.

Definition 3. For $X, Y \in \mathcal{N}$ we say that Y is a *subattribute* of X (notation: $X \geq Y$), if $\geq$ is the smallest partial order on $\mathcal{N}$ satisfying the following properties:

- $X \geq \lambda$ for all $X \in \mathcal{N}$;
- $X(Y_1, \dots, Y_n) \geq X(X'_{\sigma(1)}, \dots, X'_{\sigma(m)})$ for some injective $\sigma : \{1, \dots, m\} \to \{1, \dots, n\}$ and $Y_{\sigma(i)} \geq X'_{\sigma(i)}$ for all $i = 1, \dots, m$;
- $X\{Y\} \geq X\{X'\}$, whenever $Y \geq X'$ holds;
- $X[Y] \geq X[X'] \geq X\langle X'\rangle$, whenever $Y \geq X'$ holds;
- $X\langle Y\rangle \geq X\langle X'\rangle \geq X\{X'\}$, whenever $Y \geq X'$ holds.

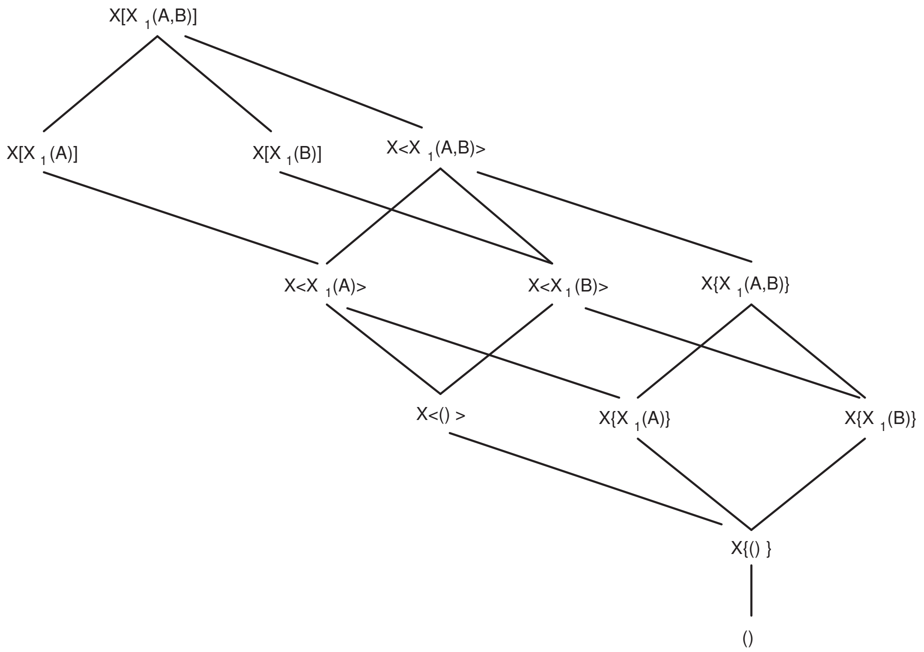

Figure 1: Brouwerian algebra for $\mathcal{S}(X[X_1(A, B)])$

For $X \geq Y$ we obtain the desired canonical projection $\pi_Y^X : dom(X) \rightarrow dom(Y)$. Obviously, for record attributes the projection throws away some components and recursively applies projection functions to the remaining ones, while for lists, sets and multisets a projection function is applied to all elements. Note that we also included the projection of a list onto a multiset by simply fogetting the order, and of a multiset to a set by forgetting multiplicities. It is rather obvious to see that $\geq$ induces the structure of a Brouwerian algebra on $\mathcal{S}(X) = \{Y \in \mathcal{N} \mid X \geq Y\}$, the set of subattributes of the attribute X. That is, $\mathcal{S}(X)$ is a distributive lattice with λ as bottom element, X as top element, and relative pseudo-complements $Y \leftarrow Z = \bigsqcap\{U \mid Y \sqcup U \geq Z\}$. Figure 1 illustrates such an algebra $\mathcal{S}(X[X_1(A, B)])$.

3 Functional Dependencies

In this section we will define functional dependencies on $\mathcal{S}(X)$, then extend them to include the possibility of counting for sets, and finally derive a some sound and complete system of derivation rules. We will use *instances* of X, which are finite sets $r \subseteq dom(X)$.

Definition 4. Let $X \in \mathcal{N}$. A *functional dependency* (FD) on $\mathcal{S}(X)$ is an expression $\mathcal{Y} \to \mathcal{Z}$ with $\mathcal{Y}, \mathcal{Z} \subseteq \mathcal{S}(X)$. An instance r of X *satisfies* $\mathcal{Y} \to \mathcal{Z}$ (notation: $r \models \mathcal{Y} \to \mathcal{Z}$) iff for all $t_1, t_2 \in r$ with $\pi_Y^X(t_1) = \pi_Y^X(t_2)$ for all $Y \in \mathcal{Y}$ we also have $\pi_Z^X(t_1) = \pi_Z^X(t_2)$ for all $Z \in \mathcal{Z}$.

Our Definition 3 of subattributes already includes the projection of lists to multisets and of multisets to sets. In addition, a set value may always be considered as a multiset value. Therefore, for a set attribute $X\{X'\}$ we may widen the notion of functional dependency to include "lifted" dependencies $\mathcal{Y} \to \mathcal{Z}$ with $\mathcal{Y}, \mathcal{Z} \subseteq \mathcal{S}(X^{\langle\rangle})$, where we define $X^{\langle\rangle} = X\langle X' \rangle$.

In doing so, X becomes a subattribute of $X^{\langle\rangle}$, and the "normal" FDs on X are just a subset of the lifted ones. Furthermore, as $\pi_X^{X^{\langle\rangle}}(v) = v$ holds for all $v \in dom(X)$, the notion of satisfiability of such FDs remains unchanged. In other words, it is sufficient to consider the "lifted" FDs just as FDs on $X^{\langle\rangle}$.

The extension adds FDs on sets including the possibility of counting, but for the problem of axiomatisation it is sufficient to look at the FDs as defined above. For this we first need the notion of reconsilable subattributes.

Definition 5. Two subattributes $Y, Z \in \mathcal{S}(X)$ are called *reconsilable* iff one of the following holds:

1. $Y \geq Z$ or $Z \geq Y$;
2. $X = X[X']$, $Y = X[Y']$, $Z = X[Z']$ and $Y', Z' \in \mathcal{S}(X')$ are reconsilable;
3. $X = X(X_1, \dots, X_n)$, $Y = X(Y_1, \dots, Y_n)$, $Z = X(Z_1, \dots, Z_n)$ and $Y_i, Z_i \in \mathcal{S}(X_i)$ are reconsilable for all $i = 1, \dots, n$.

Note that for the set- and multiset-constructor we can only obtain reconsilability for subattributes in a $\geq$-relation.

Theorem 6. *The following axioms and rules are sound for the implication of FDs on* $\mathcal{S}(X)$*:*

reflexivity axiom:

$$\frac{}{\mathcal{Y} \to \mathcal{Z}}\ \mathcal{Z} \subseteq \mathcal{Y} \tag{1}$$

subattribute axiom:

$$\frac{}{\{Y\} \to \{Z\}}\ Y \geq Z \tag{2}$$

join axiom:

$$\frac{}{\{Y, Z\} \to \{Y \sqcup Z\}}\ Y, Z \textit{ reconsilable} \tag{3}$$

λ axiom:

$$\overline{\emptyset \to \{\lambda\}} \tag{4}$$

extension rule:

$$\frac{\mathcal{Y} \to \mathcal{Z}}{\mathcal{Y} \to \mathcal{Y} \cup \mathcal{Z}} \tag{5}$$

transitivity rule:

$$\frac{\mathcal{Y} \to \mathcal{Z} \quad \mathcal{Z} \to \mathcal{U}}{\mathcal{Y} \to \mathcal{U}} \tag{6}$$

Proof. The proof is trivial except for the join axiom (3). Solet $t_1, t_2 \in r$ for some instance $r \subseteq dom(X)$ with $\pi_Y^X(t_1) = \pi_Y^X(t_2)$ and $\pi_Z^X(t_1) = \pi_Z^X(t_2)$ for reconsilable subattributes $Y, Z \in \mathcal{S}(X)$.

- In case $Y \geq Z$ we have $Y \sqcup Z = Y$ and thus $\pi_{Y \sqcup Z}^X(t_1) = \pi_{Y \sqcup Z}^X(t_2)$.
- In case $X = X[X']$ we must have $Y = X[Y']$ and $Z = X[Z']$ with reconsilable subattributes $Y', Z' \in \mathcal{S}(X')$. Furthermore, $t_1 = [t_{1,1}, \dots, t_{1,n}]$ and $t_2 = [t_{2,1}, \dots, t_{2,m}]$. This gives $n = m$, $\pi_{Y'}^{X'}(t_{1,j}) = \pi_{Y'}^{X'}(t_{2,j})$ and $\pi_{Z'}^{X'}(t_{1,j}) = \pi_{Z'}^{X'}(t_{2,j})$ for all $j = 1, \dots, n$.

  By induction we obtain $\pi_{Y' \sqcup Z'}^{X'}(t_{1,j}) = \pi_{Y' \sqcup Z'}^{X'}(t_{2,j})$ for all $j = 1, \dots, n$. From this and $Y \sqcup Z = X[Y' \sqcup Z']$ follows $\pi_{Y \sqcup Z}^X(t_1) = \pi_{Y \sqcup Z}^X(t_2)$.
- In case $X = X(X_1, \dots, X_n)$ we must have $Y = X(Y_1, \dots, Y_n)$ and $Z = X(Z_1, \dots, Z_n)$ with reconsilable subattributes $Y_i, Z_i \in \mathcal{S}(X_i)$ for $i = 1, \dots, n$. Furthermore, $t_1 = (t_{1,1}, \dots, t_{1,n})$ and $t_2 = (t_{2,1}, \dots, t_{2,n})$, which implies $\pi_{Y_i}^{X_i}(t_{1,i}) = \pi_{Y_i}^{X_i}(t_{2,i})$ and $\pi_{Z_i}^{X_i}(t_{1,i}) = \pi_{Z_i}^{X_i}(t_{2,i})$ for all $i = 1, \dots, n$. By induction we obtain $\pi_{Y_i \sqcup Z_i}^{X_i}(t_{1,i}) = \pi_{Y_i \sqcup Z_i}^{X_i}(t_{2,i})$ for all $i = 1, \dots, n$. From this and $Y \sqcup Z = X(Y_1 \sqcup Z_1, \dots, Y_n \sqcup Z_n)$ follows $\pi_{Y \sqcup Z}^X(t_1) = \pi_{Y \sqcup Z}^X(t_2)$. $\square$

In order to show that the axioms and rules in Theorem 6 are also complete, we consider *conincidence ideals*.

Definition 7. A *coincidence ideal* on $\mathcal{S}(X)$ is a subset $\mathcal{F} \subseteq \mathcal{S}(X)$ with the following properties:

1. $\lambda \in \mathcal{F}$;
2. if $Y \in \mathcal{F}$ and $Z \in \mathcal{S}(X)$ with $Y \geq Z$, then $Z \in \mathcal{F}$;
3. if $Y, Z \in \mathcal{F}$ are reconsilable, then $Y \sqcup Z \in \mathcal{F}$.

The name "coincidence ideals" was chosen, because these ideals characterise sets of subattributes, on which two complex values coincide – the proof of this simple fact is analogous to the soundness proof above. The corresponding definition in [Sali and Schewe, 2005] is much lengthier, as it contains a lot of additional properties that only make sense, if the union constructor is present. We first show a simple decendence lemma for coincidence ideals.

Lemma 8. *Let $\mathcal{F}$ be a coincidence ideal on $\mathcal{S}(X)$.*

1. *If $X = X(X_1', \dots, X_n')$, then $\mathcal{F}_i = \{Y_i \in \mathcal{S}(X_i') \mid X(\lambda, \dots, Y_i, \dots, \lambda) \in \mathcal{F}\}$ is a coincidence ideal.*
2. *If $X = X[X']$, such that X' is not a union attribute, and $\mathcal{F} \neq \{\lambda\}$, then $\mathcal{G} = \{Y \in \mathcal{S}(X') \mid X[Y] \in \mathcal{F}\}$ is a coincidence ideal.*

Proof. We only show property 3 of Definition 7 – the other two properties are trivial.

If $Y_i^{(1)}, Y_i^{(2)} \in \mathcal{F}_i$ are reconsilable, then also $X(\lambda, \dots, Y_i^{(j)}, \dots, \lambda) \in \mathcal{F}$ ($j = 1, 2$) are reconsilable, which gives $X(\lambda, \dots, Y_i^{(1)} \sqcup Y_i^{(2)}, \dots, \lambda) \in \mathcal{F}$ for their join. By definition $Y_i^{(1)} \sqcup Y_i^{(2)} \in \mathcal{F}_i$ follows.

If $Y^{(1)}, Y^{(2)} \in \mathcal{G}$ are reconsilable, then also $X[Y^{(j)}$ ($j = 1, 2$) are reconsilable, which gives $X[Y^{(1)} \sqcup Y^{(2)}] \in \mathcal{F}$ for their join. By definition $Y^{(1)} \sqcup Y^{(2)} \in \mathcal{G}$ follows. $\square$

Unfortunately, this decendence property does not hold for the set- and the multiset-constructors. So, we need a direct construction for these constructors, which will use *distinguished values*.

Definition 9. Let X be a nested attribute such that the union-constructor only appears in X inside a list-constructor. For each $Y \in \mathcal{S}^r(X)$ we define the *distinguished value* $\tau_Y^X \in dom(X)$ as follows:

1. $\tau_\lambda^\lambda = \top$;
2. $\tau_A^A = a$ and $\tau_\lambda^A = a'$ for a simple attribute A and $a, a' \in dom(A)$, $a \neq a'$;
3. $\tau_{X(Y_1,\dots,Y_n)}^{X(X_1,\dots,X_n)} = (X_1 : \tau_{Y_1}^{X_1}, \dots, X_n : \tau_{Y_n}^{X_n})$;
4. $\tau_{X\{Y\}}^{X\{X'\}} = \{\tau_Y^{X'}\}$ and $\tau_\lambda^{X\{X'\}} = \emptyset$;
5. $\tau_{X\langle Y\rangle}^{X\langle X'\rangle} = \langle \tau_Y^{X'}, \tau_Y^{X'} \rangle$, $\tau_{X\{Y\}}^{X\langle X'\rangle} = \langle \tau_Y^{X'} \rangle$, and $\tau_\lambda^{X\langle X'\rangle} = \langle\rangle$;
6. $\tau_{X[Y]}^{X[X']} = [\tau_Y^{X'}, \tau_Y^{X'}, \tau_Y^{X'}]$, $\tau_{X\langle Y\rangle}^{X[X']} = [\tau_Y^{X'}, \tau_Y^{X'}]$, $\tau_{X\{Y\}}^{X[X']} = [\tau_Y^{X'}]$, and $\tau_\lambda^{X[X']} = []$.

Note that item 3 in Definition 9 includes the case $X(\lambda, \dots, \lambda) = \lambda$. Using these distinguished values we first show some elementary properties for them, which are used in a second step to prove the main result for the case of the set- and the multiset-constructors. A proof without the restructuring rules was given in [Hartmann et al., 2006].

Lemma 10. *Let X be a nested attribute such that the union-constructor appears in X only immediately inside a list-constructor. Let $\mathcal{G} \subseteq \mathcal{S}(X)$ be an ideal on X. Then we have:*

1. *If we have $\pi^X_Y(\tau^X_Z) = \pi^X_Y(\tau^X_Y)$, then $Z \geq Y$.*
2. *For $Y, Z \in \mathcal{S}(X)$ and $Z^\sharp = (Y \leftarrow Z) \leftarrow (Y \sqcap Z)$ we have $\pi^X_Y(\tau^X_Z) = \pi^X_Y(\tau^X_{Z^\sharp})$.*

Proof. For the first statement there is nothing to show for $Y = \lambda$, $Z \geq Y$ or $Y = X\{\lambda\}$. We then use structural induction on X:

For a simple attribute $X = A$ we have $Y = A$ and $Z = \lambda$, so $\pi^X_Y(\tau^X_Z) = a' \neq a = \pi^X_Y(\tau^X_Y)$.

For $X = X(X_1, \dots, X_n)$, $Y = X(Y_1, \dots, Y_n)$ and $Z = X(Z_1, \dots, Z_n)$ we have $\pi^X_Y(\tau^X_Z) = (\pi^{X_1}_{Y_1}(\tau^{X_1}_{Z_1}), \dots, \pi^{X_1}_{Y_1}(\tau^{X_n}_{Z_n}))$ and $\pi^X_Y(\tau^X_Y) = (\pi^{X_1}_{Y_1}(\tau^{X_1}_{Y_1}), \dots, \pi^{X_1}_{Y_1}(\tau^{X_n}_{Y_n}))$, hence $\pi^{X_i}_{Y_i}(\tau^{X_i}_{Z_i}) = \pi^{X_i}_{Y_i}(\tau^{X_i}_{Y_i})$ for all $i = 1, \dots, n$. By induction we get $Z_i \geq Y_i$ for all $i = 1, \dots, n$, thus $Z \geq Y$.

For $X = X\{X'\}$ and $Y = X\{Y'\}$ we must have $Z = X\{Z'\}$ or $Z = \lambda$. Then we get $\pi^X_Y(\tau^X_Z) = \{\pi^{X'}_{Y'}(\tau^{X'}_{Z'})\}$ in the first case, and $\pi^X_Y(\tau^X_Z) = \emptyset$ in the second case. Furthermore, we get $\pi^X_Y(\tau^X_Y) = \{\pi^{X'}_{Y'}(\tau^{X'}_{Y'})\}$. Hence $Z \neq \lambda$, and by induction we get $Z' \geq Y'$, which implies $Z \geq Y$.

Analogously, for $X = X\langle X'\rangle$ and $Y = X\langle Y'\rangle$ we must have $Z = X\langle Z'\rangle$, $Z = X\{Z'\}$ or $Z = \lambda$, which gives $\pi^X_Y(\tau^X_Z) = \langle \pi^{X'}_{Y'}(\tau^{X'}_{Z'}), \pi^{X'}_{Y'}(\tau^{X'}_{Z'})\rangle$ in the first case, $\pi^X_Y(\tau^X_Z) = \langle \pi^{X'}_{Y'}(\tau^{X'}_{Z'})\rangle$ in the second, and $\pi^X_Y(\tau^X_Z) = \langle\rangle$ in the third case. As we have $\pi^X_Y(\tau^X_Y) = \langle \pi^{X'}_{Y'}(\tau^{X'}_{Y'}), \pi^{X'}_{Y'}(\tau^{Y'}_{Z'})\rangle$, we must have $Z = X\langle Z'\rangle$ and by induction $Z' \geq Y'$, which implies $Z \geq Y$. The case for $X = X\langle X'\rangle$ and $Y = X\{Y'\}$ is handled analogously.

Finally, the list case, i.e. $X = X[X']$ also follows analogously.

For the second statement there is nothing to prove for $Y = \lambda$ or $Y \geq Z$. The latter one gives $Z^\sharp = (Y \leftarrow Z) \leftarrow (Y \sqcap Z) = \lambda \leftarrow Z = Z$. Now proceed by induction on X and assume $\lambda \neq Y \ngeq Z$. Note that the cases $X = \lambda$ and X a simple attribute are already covered.

For $X = X(X_1, \dots, X_n)$, $Y = X(Y_1, \dots, Y_n)$ and $Z = X(Z_1, \dots, Z_n)$ we have by induction $\pi^{X_i}_{Y_i}(\tau^{X_i}_{Z_i}) = \pi^{X_i}_{Y_i}(\tau^{X_i}_{Z_i^\sharp})$ for all $i = 1, \dots, n$ with $Z_i^\sharp = (Y_i \leftarrow Z_i) \leftarrow (Y_i \sqcap Z_i)$. This implies

$$\begin{aligned}\pi^X_Y(\tau^X_Z) &= (X_1 : \pi^{X_1}_{Y_1}(\tau^{X_1}_{Z_1}), \dots, \pi^{X_n}_{Y_n}(\tau^{X_n}_{Z_n})) \\ &= (X_1 : \pi^{X_1}_{Y_1}(\tau^{X_1}_{Z_1^\sharp}), \dots, \pi^{X_n}_{Y_n}(\tau^{X_n}_{Z_n^\sharp})) = \pi^X_Y(\tau^X_{Z^\sharp}) \, .\end{aligned}$$

For $X = X\{X'\}$ and $Y = X\{Y'\}$ we must have $Z = X\{Z'\}$ with $Y' \not\geq Z'$. By induction we get $\pi^{X'}_{Y'}(\tau^{X'}_{Z'}) = \pi^{X'}_{Y'}(\tau^{X'}_{Z'^\sharp}))$ with $Z'^\sharp = (Y' \leftarrow Z') \leftarrow (Y' \sqcap Z')$. This implies

$$\pi^X_Y(\tau^X_Z) = \{\pi^{X'}_{Y'}(\tau^{X'}_{Z'})\} = \{\pi^{X'}_{Y'}(\tau^{X'}_{Z'^\sharp})\} = \pi^X_Y(\tau^X_{Z^\sharp}) \ .$$

For $X = X\langle X'\rangle$ and $Y = X\{Y'\}$ we must have $Z = X\{Z'\}$ with $Y' \not\geq Z'$ or $Z = X\langle Z'\rangle$. In both cases we have $Z'^\sharp = (Y' \leftarrow Z') \leftarrow (Y' \sqcap Z')$, and by induction $\pi^{X'}_{Y'}(\tau^{X'}_{Z'}) = \pi^{X'}_{Y'}(\tau^{X'}_{Z'^\sharp}))$. As $Z^\sharp = X\{Z'^\sharp\}$ in the first case, and $Z^\sharp = X\langle Z'^\sharp\rangle$ in the second one, this implies

$$\pi^X_Y(\tau^X_Z) = \{\pi^{X'}_{Y'}(\tau^{X'}_{Z'})\} = \{\pi^{X'}_{Y'}(\tau^{X'}_{Z'^\sharp})\} = \pi^X_Y(\tau^X_{Z^\sharp}) \ .$$

For $X = X\langle X'\rangle$ and $Y = X\langle Y'\rangle$ we must have $Z = X\{Z'\}$ or $Z = X\langle Z'\rangle$ with $Y' \not\geq Z'$. By induction we have $\pi^{X'}_{Y'}(\tau^{X'}_{Z'}) = \pi^{X'}_{Y'}(\tau^{X'}_{Z'^\sharp}))$ with $Z'^\sharp = (Y' \leftarrow Z') \leftarrow (Y' \sqcap Z')$. This implies (with $x = 1$ or 2 in the first or second case, respectively):

$$\pi^X_Y(\tau^X_Z) = \langle\underbrace{\pi^{X'}_{Y'}(\tau^{X'}_{Z'})}_{x\text{-times}}\rangle = \langle\underbrace{\pi^{X'}_{Y'}(\tau^{X'}_{Z'^\sharp})}_{x\text{-times}}\rangle = \pi^X_Y(\tau^X_{Z^\sharp}) \ .$$

The case for lists is analogous to the one for multisets, except that we get three cases involving lists of length one, two or three. $\square$

We now use Lemma 10 to prove the main result for coincidence ideals. Note that an analogous result for the relational data model would have been completely trivial.

Theorem 11. *Let $\mathcal{F} \subseteq \mathcal{S}(X)$ be a coincidence ideal. Then there exist $t_1, t_2 \in dom(X)$ such that $\pi^X_Y(t_1) = \pi^X_Y(t_2)$ holds iff $Y \in \mathcal{F}$.*

Proof. We use induction on X. The case $X = \lambda$ is trivial.

For a simple attribute $X = A$ we either have $\mathcal{F} = \{\lambda\}$ or $\mathcal{F} = \{A, \lambda\}$. In the former case take $t_1 = a$ and $t_2 = a'$ for $a, a' \in dom(A)$ with $a \neq a'$. In the latter case take $t_1 = t_2 = a$.

For $X = X(X_1, \dots, X_n)$ take the coincidence ideals $\mathcal{F}_i$ on X_i constructed in Lemma 8. By induction we find $t_{1i}, t_{2i} \in dom(X_i)$ with $\pi^{X_i}_{Y_i}(t_{1i}) = \pi^{X_i}_{Y_i}(t_{2i})$ iff $Y_i \in \mathcal{F}_i$. So take $t_1 = (X_1 : t_{11}, \dots, X_n : t_{1n})$ and $t_2 = (X_1 : t_{21}, \dots, X_n : t_{2n})$. For $Y = X(Y_1, \dots, Y_n) \in \mathcal{F}$ we have

$$\begin{aligned}\pi^X_Y(t_1) &= (X_1 : \pi^{X_1}_{Y_1}(t_{11}), \dots, X_n : \pi^{X_n}_{Y_n}(t_{1n}))\\ &= (X_1 : \pi^{X_1}_{Y_1}(t_{21}), \dots, X_n : \pi^{X_n}_{Y_n}(t_{2n})) = \pi^X_Y(t_2) \ .\end{aligned}$$

For $Y = X(Y_1, \dots, Y_n) \notin \mathcal{F}$ there is at least one $Y_i \notin \mathcal{F}_i$, which gives

$$\begin{aligned}\pi^X_Y(t_1) &= (X_1 : \pi^{X_1}_{Y_1}(t_{11}), \dots, X_n : \pi^{X_n}_{Y_n}(t_{1n}))\\ &\neq (X_1 : \pi^{X_1}_{Y_1}(t_{21}), \dots, X_n : \pi^{X_n}_{Y_n}(t_{2n})) = \pi^X_Y(t_2) \ .\end{aligned}$$

For $X = X\{X'\}$ consider first the case $\mathcal{F} = \{\lambda\}$. For this take $t_1 = \{v\}$ with $v \in dom(X')$ and $t_2 = \emptyset$. For $Y = X\{Y'\} \notin \mathcal{F}$ we get $\pi^X_Y(t_1) = \{\pi^{X'}_{Y'}(v)\} \neq \emptyset = \pi^X_Y(t_2)$.

So assume now $\mathcal{F} \neq \{\lambda\}$. In this case let $\mathcal{G} = \{Y \in \mathcal{S}(X') \mid X\{Y\} \in \mathcal{F}\}$, which is an ideal on $\mathcal{S}(X')$, but not a coincidence ideal. Then define $t_1 = \{\tau^{X'}_Z \mid Z \in \mathcal{S}(X')\}$ and $t_2 = \{\tau^{X'}_Z \mid Z \in \mathcal{G}\}$. For $Y \in \mathcal{G}$, i.e. $X\{Y\} \in \mathcal{F}$, we have $Z^\sharp = (Y \leftarrow Z) \leftarrow (Y \sqcap Z) \leq Y \sqcap Z \leq Y \in \mathcal{G}$ and $\pi^{X'}_Y(\tau^{X'}_Z) = \pi^{X'}_Y(\tau^{X'}_{Z^\sharp})$ by item 2 of Lemma 10. This implies

$$\pi^X_{X\{Y\}}(t_1) = \{\pi^{X'}_Y(\tau^{X'}_Z) \mid Z \in \mathcal{S}(X')\} = \{\pi^{X'}_Y(\tau^{X'}_Z) \mid Z \in \mathcal{G}\} = \pi^X_{X\{Y\}}(t_2).$$

For $Y \notin \mathcal{G}$, i.e. $X\{Y\} \notin \mathcal{F}$, assume $\pi^X_{X\{Y\}}(t_1) = \pi^X_{X\{Y\}}(t_2)$. Then in particular $\pi^{X'}_Y(\tau^{X'}_Y) = \pi^{X'}_Y(\tau^{X'}_Z)$ holds for some $Z \in \mathcal{G}$. Item 1 of Lemma 10 implies $Z \geq Y$, from which we get the contradiction $Y \in \mathcal{G}$.

For $X = X\langle X'\rangle$ and $\mathcal{F} = \{\lambda\}$ take $t_1 = \langle v\rangle$ with $v \in dom(X')$ and $t_2 = \langle\rangle$. For $Y_1 = X\{Y'\} \notin \mathcal{F}$ or $Y_2 = X\langle Y'\rangle \notin \mathcal{F}$ we get $\pi^X_{Y_1}(t_1) = \{\pi^{X'}_{Y'}(v)\} \neq \emptyset = \pi^X_{Y_1}(t_2)$, and $\pi^X_{Y_2}(t_1) = \langle\pi^{X'}_{Y'}(v)\rangle \neq \langle\rangle = \pi^X_{Y_2}(t_2)$.

So assume now $\mathcal{F} \neq \{\lambda\}$. In this case let $\mathcal{G}_o = \{Y \in \mathcal{S}(X') \mid X\langle Y\rangle \in \mathcal{F}\}$ and $\mathcal{G}_u = \{Y \in \mathcal{S}(X') \mid X\{Y\} \in \mathcal{F}\}$, which are both ideals on $\mathcal{S}(X')$ with $\mathcal{G}_o \subseteq \mathcal{G}_u$, but not coincidence ideals. Let $Y_1, \dots, Y_k$ be the minimal elements in the filter $\mathcal{S}(X') - \mathcal{G}_o$, and let $Y_{k+1}, \dots, Y_m$ be the minimal elements in $\mathcal{S}(X') - \mathcal{G}_u$. For each Y_i $(i = 1, \dots, m)$ let $Y'_{i1}, \dots, Y'_{ix_i}$ be the maximal proper subattributes of Y_i. Then Y_i generates a Boolean algebra $\mathbb{B}_i \subseteq \mathcal{S}(X')$ with top element Y_i, bottom element $Y'_{i_1} \sqcap \dots \sqcap Y'_{ix_i}$, and containing all Y'_{ij} $(j = 1, \dots, x_i)$ [Hartmann et al., 2006, Lemma 22].

For $X \in \mathbb{B}_i$ let $d_i(Z)$ be the distance of Z from Y_i in $\mathbb{B}_i$. Then for $i = 1, \dots, k$ define $t_{1i} = \langle\tau^{X'}_Z \mid Z \in \mathbb{B}_i, d_i(Z) \text{ even}\rangle$, and $t_{2i} = \langle\tau^{X'}_Z \mid Z \in \mathbb{B}_i, d_i(Z) \text{ odd}\rangle$. For $i \in \{k+1, \dots, m\}$ let $\hat{\imath} \in \{1, \dots, k\}$ with $Y_{\hat{\imath}} \leq Y_i$. If the distance between $Y_{\hat{\imath}}$ and Y_i is odd, define $t_{1i} = \langle\tau^{X'}_Z \mid Z \in \mathbb{B}_i, d_i(Z) \text{ even}\rangle$ and $t_{2i} = \langle\tau^{X'}_Z \mid Z \in \mathbb{B}_i, d_i(Z) \text{ odd}\rangle$. Otherwise define $t_{1i} = \langle\tau^{X'}_Z \mid Z \in \mathbb{B}_i, d_i(Z) \text{ odd}\rangle$ and $t_{2i} = \langle\tau^{X'}_Z \mid Z \in \mathbb{B}_i, d_i(Z) \text{ even}\rangle$. Finally, take multiset union $\uplus$ to define $t_j = \biguplus_{i=1}^m t_{ji}$ $(j = 1, 2)$.

As shown in [Hartmann et al., 2006, Lemma 23] $\pi^X_{X\langle Y\rangle}(t_{1i}) \neq \pi^X_{X\langle Y\rangle}(t_{2i})$ holds iff $Y_i \leq Y$, i.e. t_{1i} and t_{2i} coincide exactly on the multiset subattributes outside the principal filter generated by $X\langle Y_i\rangle$. From this, using [Hartmann et al., 2006, Lemma 24] we conclude that $\pi^X_{X\langle Y\rangle}(t_1) = \pi^X_{X\langle Y\rangle}(t_2)$ holds iff $y \not\geq Y_i$ for all $i = 1, \dots, m$. Due to the choice of the Y_i and the fact that for $i \in \{k+1, \dots, m\}$ there always exists some $\hat{\imath} \in \{1, \dots, k\}$ with $Y_{\hat{\imath}} \leq Y_i$ we obtain $\pi^X_{X\langle Y\rangle}(t_1) = \pi^X_{X\langle Y\rangle}(t_2)$ iff $Y \in \mathcal{G}_o$.

Now take a maximal $Y \in \mathcal{G}_u - \mathcal{G}_o$. Then Y is a maximal proper subattribute of some Y_i with $i \in \{k+1, \dots, m\}$, and due to the construction we have $t_{1\hat{\imath}} \subseteq t_{2i}$ and $t_{2\hat{\imath}} \subseteq t_{1i}$. As we have $\pi^X_{X\{Y\}}(t_{1i}) \neq \pi^X_{X\{Y\}}(t_{2i})$ iff $Y \geq Y_i$, we conclude

$\pi^X_{X\{Y\}}(t_1) = \pi^X_{X\{Y\}}(t_2)$ iff $Y \not\geq Y_i$ for all $i = k+1, \dots, m$. Due to the choice of the Y_i we finally obtain $\pi^X_{X\{Y\}}(t_1) = \pi^X_{X\{Y\}}(t_2)$ iff $Y \in \mathcal{G}_u$, and hence $\pi^X_Z(t_1) = \pi^X_Z(t_2)$ iff $Z \in \mathcal{F}$.

Finally, let $X = X[X']$. If we have $\mathcal{F} = \{\lambda\}$, take $t_1 = [v]$ with $v \in dom(X')$ and $t_2 = []$. For $Y_1 = X\{Y'\} \notin \mathcal{F}$, $Y_2 = X\langle Y'\rangle \notin \mathcal{F}$ or $Y_3 = X[Y'] \notin \mathcal{F}$ we get $\pi^X_{Y_1}(t_1) = \{\pi^{X'}_{Y'}(v)\} \neq \emptyset = \pi^X_{Y_1}(t_2)$, $\pi^X_{Y_2}(t_1) = \langle\pi^{X'}_{Y'}(v)\rangle \neq \langle\rangle = \pi^X_{Y_2}(t_2)$, and $\pi^X_{Y_3}(t_1) = [\pi^{X'}_{Y'}(v)] \neq [] = \pi^X_{Y_3}(t_2)$.

So assume now $\mathcal{F} \neq \{\lambda\}$. In this case take $\mathcal{F}' = \mathcal{S}(X\langle X'\rangle) \cap \mathcal{F}$, which is a coincidence ideal on $\mathcal{S}(X\langle X'\rangle)$. Using the construction above we obtain multisets $t'_1, t'_2 \in dom(X\langle X'\rangle)$ with $\pi^{X\langle X'\rangle}_Z(t'_1) = \pi^{X\langle X'\rangle}_Z(t'_2)$ iff $Z \in \mathcal{F}'$. Now let $Y_1, \dots, Y_k$ be the maximal element in $\mathcal{S}(X')$ such that $X[Y_i] \in \mathcal{F}$ for all $i = 1, \dots, k$. In particular, $X\langle Y_i\rangle \in \mathcal{F}'$.

For $k = 0$ there is nothing to show. For $k = 1$ we can order the elements in t'_1 and t'_2 in such a way that we obtain lists t_1 and t_2 with $\pi^X_{X[Y_1]}(t_1) = \pi^X_{X[Y_1]}(t_2)$. Then obviously $\pi^X_Z(t_1) = \pi^X_Z(t_2)$ holds iff $Z \in \mathcal{F}$.

Now assume $k > 1$. Then $Y_1, \dots, Y_k$ must be pairwise not reconsilable. As the elements in t'_1 and t'_2 all have the form $\tau = \tau^{X'}_Y$ for some $Y \in \mathcal{S}(X')$, we will now replace them by elements $\hat{\tau}$ defined with respect to $Y_1, \dots, Y_k$ such that for $\hat{t}_j = \langle\hat{\tau} \mid \tau \in t_j\rangle$ $(j = 1, 2)$ we still have $\pi^{X\langle X'\rangle}_Z(\hat{t}_1) = \pi^{X\langle X'\rangle}_Z(\hat{t}_2)$ iff $Z \in \mathcal{F}'$, and $\hat{t}_1$ and $\hat{t}_2$ can be ordered in a way that the resulting lists t_1 and t_2 satisfy $\pi^X_{X[Y_i]}(t_1) = \pi^X_{X[Y_i]}(t_2)$ for all $i = 1, \dots, k$, which will complete our proof.

In order to achieve these two properties we temporarily identify list values with record values – all occurring lists have the length one, two or three with identical elements. Furthermore, we may flatten nested record values. Then split t'_1 and t'_2 into multisets containing only tuples of the same length, so without loss of generality we may assume that all τ have the form $(\tau_1, \dots, \tau_n)$, and each Y_i has the form $X'(Y_{i1}, \dots, Y_{in})$. Now take ℓ such that all $X'(Y_{i(\ell+1)}, \dots, Y_{in})$ are pairwise reconsilable. In particular, for $j \leq \ell$ the Y_{ij} $(i = 1, \dots, k)$ must be set or multiset attributes. Split t'_1 and t'_2 into submultisets containing tuples with equal projections on the last $n - \ell$ components.

Let these elements be $\tau^{(j)}_i = (v^{(j)}_{1i}, \dots, v^{(j)}_{\ell i}, v_{\ell+1}, \dots, v_n)$ with $i = 1, \dots, m$ and $j = 1, 2$. Now define $\widehat{\tau^{(j)}_i} = (\bigcup^m_{i=1} v^{(j)}_{1i}, \dots, \bigcup^m_{i=1} v^{(j)}_{\ell i}, v_{\ell+1}, \dots, v_n)$, where $\bigcup$ has to be understood as either set or multiset union. This obviously satisfies the first of the two desired properties.

For the second desired property we know that t'_1 and t'_2 in such a way that we obtain lists $t^{(i)}_1$ and $t^{(i)}_2$ such that $\pi^X_{X[Y_i]}(t^{(i)}_1) = \pi^X_{X[Y_i]}(t^{(i)}_2)$, but the order may be different for different $i \in \{1, \dots, k\}$. However, elements τ_1, τ_2 that must appear in different positions for different $i \in \{1, \dots, k\}$ give rise to the same $\hat{\tau}_1 = \hat{\tau}_2$, which gives us the second desired property and hence the theorem. $\square$

With this central result on coincidence ideals we are now able to finalise the proof for the axiomatisation showing that the axioms and rules in Theorem 6 are complete. The idea of the proof is simply to follow the corresponding proof for the relational model. However, the use of Theorem 11 will be central, while in the relational model the corresponding construction was trivial.

Theorem 12. *The set of axioms and rules in Theorem 6 is complete for the implication of FDs on* $\mathcal{S}(X)$.

Proof. Let Σ be a set of FDs on $\mathcal{S}(X)$ and assume $\mathcal{Y} \to \mathcal{Z} \notin \Sigma^+$. Then there exists a subattribute $Z \in \mathcal{Z}$ with $\mathcal{Y} \to \{Z\} \notin \Sigma^+$. Thus, $Z \notin \bar{\mathcal{Y}} = \{Z' \mid \mathcal{Y} \to \{Z'\} \in \Sigma^+\}$. We show that $\mathcal{F} = \bar{\mathcal{Y}}$ is a coincidence ideal on $\mathcal{S}(X)$:

1. $\lambda \in \mathcal{F}$ follows immediately from the reflexivity axiom (1), the λ axiom (4), and the transitivity rule (6).
2. For $Z_1 \in \mathcal{F}$ and $Z_1 \geq Z_2$ the subattribute axiom (2) and the transitivity rule (6) imply $Z_2 \in \mathcal{F}$.
3. For reconsilable $Z_1, Z_2 \in \mathcal{F}$ the join axiom (3) and the transitivity rule (6) imply $Z_1 \sqcup Z_2 \in \mathcal{F}$.

Now apply Theorem 11, which gives us $r = \{t_1, t_2\} \subseteq dom(X)$ with $\pi_Y^X(t_1) = \pi_Y^X(t_2)$ iff $Y \in \mathcal{F}$. In particular, $\pi_Y^X(t_1) = \pi_Y^X(t_2)$ for all $Y \in \mathcal{Y}$, and $\pi_Z^X(t_1) \neq \pi_Z^X(t_2)$. That is, $r \not\models \mathcal{Y} \to \{Z\}$ and hence also $r \not\models \mathcal{Y} \to \mathcal{Z}$.

Finally, we show $r \models \Sigma$, for which we distinguish two cases:

– If $\mathcal{U} \subseteq \mathcal{F}$, then $\pi_U^X(t_1) = \pi_U^X(t_2)$ for all $U \in \mathcal{U}$. The reflexivity axiom and the transitivity rule allow us to derive $\mathcal{Y} \to \mathcal{V} \in \Sigma^+$, which means $\mathcal{V} \subseteq \mathcal{F}$ and thus $\pi_V^X(t_1) = \pi_V^X(t_2)$ for all $V \in \mathcal{V}$, i.e. $r \models \mathcal{U} \to \mathcal{V}$.

– If $\mathcal{U} \subseteq \mathcal{F}$, then there is some $U \in \mathcal{U}$ with $\pi_U^X(t_1) \neq \pi_U^X(t_2)$, which immediately implies $r \models \mathcal{U} \to \mathcal{V}$.

Hence we get $r \models \Sigma^*$, which finally gives $\mathcal{Y} \to \mathcal{Z} \notin \Sigma^*$. □

4 Concluding Remarks

In this paper we took up an open problem formulated in [Sali and Schewe, 2005] and added non-trivial restructuring rules to Brouwerian algebras of nested attributes. This addition adds counting to functional dependencies (FDs), as counting the number of elements in sets, multisets and lists would be enabled. We then presented a finite axiomatisation for such extended FDs, which generalises the main result in [Hartmann et al., 2006].

We did, however, exclude the union constructor, which was added in [Sali and Schewe, 2005] and which widens the possibilities to exploit counting subattributes in functional dependencies. It is an open problem, whether the axiomatisation for weak functional dependencies in [Sali and Schewe, 2005] can also be generalised to this case.

References

[Abiteboul et al., 2000] Abiteboul, S., Buneman, P., and Suciu, D. (2000). *Data on the Web: From Relations to Semistructured Data and XML*. Morgan Kaufmann Publishers.

[Abiteboul et al., 1995] Abiteboul, S., Hull, R., and Vianu, V. (1995). *Foundations of Databases*. Addison-Wesley.

[Arenas and Libkin, 2004] Arenas, M. and Libkin, L. (2004). A normal form for XML documents. *ACM Transactions on Database Systems*, 29(1):195–232.

[Hartmann et al., 2006] Hartmann, S., Link, S., and Schewe, K.-D. (2006). Axiomatisation of functional dependencies in the presence of records, lists, sets and multisets. *Theoretical Computer Science*. to appear.

[Sali and Schewe, 2005] Sali, A. and Schewe, K.-D. (2005). Weak functional dependencies on trees with restructuring. submitted for publication.

[Sali and Schewe, 2006] Sali, A. and Schewe, K.-D. (2006). Counter-free keys and functional dependencies in higher-order datamodels. *Fundamenta Informaticae*. to appear.

[Thalheim, 1991] Thalheim, B. (1991). *Dependencies in Relational Databases*. Teubner-Verlag.

[Thalheim, 2000] Thalheim, B. (2000). *Entity-Relationship Modeling: Foundations of Database Technology*. Springer-Verlag.

[Vincent et al., 2004] Vincent, M., Liu, J., and Liu, C. (2004). Strong functional dependencies and their application to normal forms in XML. *ACM Transactions on Database Systems*, 29(3):445–462.

[Wang and Topor, 2005] Wang, J. and Topor, R. (2005). Removing XML data redundancies using functional and equality-generating dependencies. In Dobbie, G. and Williams, H., editors, *Database Technologies 2005 – Sixteenth Australasian Database Conference*, volume 39 of *CRPIT*, pages 65–74. Australian Computer Society.