Acta Cybernetica **16** (2004) 459–472.

Distance Functional Dependencies in the Presence of Complex Values

Sebastian Link* and Klaus-Dieter Schewe*

Abstract

Distance functional dependencies (dFDs) have been introduced in the context of the relational data model as a generalisation of error-robust functional dependencies (erFDs). An erFD is a dependency that still holds, if errors are introduced into a relation, which cause the violation of an original functional dependency. A dFD with a distance $d = 2e + 1$ corresponds to an erFD with at most e errors in each tuple. Recently, an axiomatisation of dFDs has been obtained.

Database theory, however, does no longer deal only with flat relations. Modern data models such as the higher-order Entity-Relationship model (HERM), object oriented datamodels (OODM), or the eXtensible Meakup Language (XML) provide constructors for complex values such as finite sets, multisets and lists. In this article, dFDs with complex values are investigated. Based on a generalisation of the HAmming distance for tuples to complex values, which exploits a lattice structure on subattributes, the major achievement is a finite axiomatisation of the new class of dependencies.

Keywords. functional dependencies, complex value data models, error-robustness

1 Introduction

In [3] Demetrovics, Katona and Miklós introduced error-correcting keys in the RDM and generalised them to error-correcting functional dependencies in [4]. In both cases they studied the relationship of these dependencies to inclusion-free sets of attributes and derived combinatorical results on the size of the elements in such families. As these kinds of dependencies provide information about relations that is stable under the introduction of errors, we prefer to talk of *error-robust functional dependencies* (erFDs).

The work on error-robust functional dependencies is motivated by the fact that a database user may be confronted with a relation that contains errors. It is presumed that the user knows the structure of the relation schema, i.e. the attributes and

*Information Science Research Centre, Massey University, Private Bag 11222, Palmerston North, New Zealand, [s.link|k.d.schewe]@massey.ac.nz

the dependencies. There are various reasons for such errors to occur. For instance, a relation may have been transmitted through a noisy channel, so knowing about erFDs may help to localise the errors.

On the other hand, errors may have been introduced deliberately in order to hide and secure data. Then the knowledge about erFDs permits drawing conclusions about the errors. So the study of erFDs may lead to results on how reliable the used data hiding mechanism is. Another reason for an errornous relation may be that the data has been spoiled deliberately. Analogously to the case of using a noisy transmission channel the knowledge about erFDs may help to detect the errors.

In the conclusion of [4, p.92] the authors pose the question, whether erFDs in the RDM can be characterised, i.e., finitely axiomatised. As already shown in [4, Prop. 1.1] erFDs are subsumed by another class of dependencies, called *distance functional dependencies* (dFDs), where the distance refers to the Hamming distance of tuples projected to the left hand side of the dependency. More precisely, an erFD for the case of at most e errors in each tuple corresponds to a dFD with a distance of at most $2e + 1$. A finite axiomatisation of distance functional dependencies for the RDM including the more general case of disjunctive distance functional dependencies was achieved in [8].

Over the last decade the major focus of database theory has shifted from the relational data model to data models with complex values rather than just tuples. Examples are the higher-order Entity-Relationship model (HERM, [11]), object oriented datamodels (OODM, [10]), or most recently the eXtensible Meakup Language (XML, [1]). A natural question is, whether the theory of functional dependencies and distance functional dependencies can be carried over to these data models. For FDs this problem was addressed in [5] for finite sets, then generalised in [7] to capture sets, lists and multisets, and in [6] to capture sets and disjoint unions. In all these cases a finite axiomatisation could be achieved.

The aim of this paper is to generalise the notion and finite axiomatisation of error-robust functional dependencies. In Section 2 we summarise the results from [8] on dFDs in the relational data model, excluding the disjunctions. The Section 3 we introduce vthe fundamentals of nested attributes, which capture the gist of higher-order data models. We present some results from [7] that will be needed in this article. Section 4 introduces distance functional dependencies on nested attributes and a sound set of derivation rules for such dependencies. Finally, the completeness of this set of rules is proven.

2 Error-Robust Functional Dependencies in the RDM

We assume familarity with fundamental definitions of the RDM and functional dependencies in the RDM. One of many good sources is [9].

Suppose R is a relation schema and r, r' are R-relations. For $e \geq 0$ assume that r' results from r by introducing at most e errors per tuple. For simplicity neglect the case that r' has less elements than r, so that we can avoid considering

multisets of tuples instead of sets. We say that r satisfies the *e-error-robust functional dependency* (e-erFD) $X \to \{e\}Y$ with $X, Y \subseteq R$ iff the introduction of errors into r leading to r' would still allow to detect the functional dependency $X \to Y$. Formally, for any tuple $t' \in r'$ that corresponds to a tuple $t \in r$ there must not exist two tuples $t_1, t_2 \in r$, which both have a Hamming distance at most e from t, such that $t_1[Y] \neq t_2[Y]$ holds.

Recall that the *Hamming distance* of two tuples t_1 and t_2 (denoted as $\mathcal{H}(t_1, t_2)$) is the number of attributes B, on which $t_1[B] \neq t_2[B]$ holds.

Definition 1. Let $X, Y \subseteq R$ and $e \geq 0$. An *e-error-robust functional dependency* (e-erFD) is an expression $X \to \{e\}Y$. An R-relation r *satisfies* $X \to \{e\}Y$ iff for all R-relations r' such that there is a bijection σ between the tuples $t \in r$ and $\sigma(t) = t' \in r'$ with $\mathcal{H}(t, t') \leq e$ and all tuples $t_1, t_2 \in r$, $t' \in r'$ we have $\mathcal{H}(t_1[X], t'[X]) \leq e \wedge \mathcal{H}(t_2[X], t'[X]) \leq e \Rightarrow t_1[Y] = t_2[Y]$.

Obviously, for tuples $t_1, t_2 \in r$ with $\mathcal{H}(t_1[X], t_2[X]) \geq 2e + 1$ we obtain $t'_1[X] \neq t'_2[X]$, so these tuples cannot violate the functional dependency $X \to Y$ on r'. Conversely, for tuples $t_1, t_2 \in r$ with $\mathcal{H}(t_1[X], t_2[X]) < 2e + 1$ we may obtain $t'_1[X] = t'_2[X]$ in r', so that the tuples violate the functional dependency $X \to Y$ on r'. Using this simple fact we obtain the following easy result (see [8], also compare [4, Prop. 1.1]).

Proposition 1. *An R-relation r satisfies $X \to \{e\}Y$ iff $\mathcal{H}(t_1[X], t_2[X]) < 2e+1 \Rightarrow t_1[Y] = t_2[Y]$ holds for all tuples $t_1, t_2 \in r$.*

As in [4, p.87] we take advantage of Proposition 1 to define another class of dependencies, called *d-distance functional dependencies*, which will ease the task of finding a finite axiomatisation.

Definition 2. Let $X, Y \subseteq R$ and $d > 0$. A *d-distance functional dependency* (d-dFD) is an expression $X \to (d)Y$. An R-relation r *satisfies* $X \to (d)Y$ iff we have $\mathcal{H}(t_1[X], t_2[X]) < d \Rightarrow t_1[Y] = t_2[Y]$ for all tuples $t_1, t_2 \in r$.

As usual, we use the notation $\models_r X \to (d)Y$, if r satisfies the dFD. If Σ is a set of dFDs, we say that Σ *implies* $X \to (d)Y$ (notation: $\Sigma \models X \to (d)Y$) iff each relation r satisfying all dFDs in Σ also satisfies $X \to (d)Y$. We denote by Σ^* the *semantic hull* of Σ, i.e. the set of all dFDs implied by Σ, i.e. $\Sigma^* = \{X \to (d)Y \mid \Sigma \models X \to (d)Y\}$.

If we can find a finite, sound and complete set of rules and axioms that allows us to derive Σ^* out of Σ, then we also know how to obtain the semantic hull of a set of erFDs. This follows from the following obvious corollary of Proposition 1.

Corollary 1. *A relation r satisfies the erFD $X \to \{e\}Y$ iff r satisfies the dFD $X \to (2e + 1)Y$. In particular, 0-erFDs correspond to 1-dFDs.*

The main result on dFDs is the following theorem which was proven in a more general form in [8]. Here we use again the standard notation whereby $X, Y, Z, \ldots$ denote attribute sets, $A, B, C, \ldots$ denote attributes or attribute sets with just one attribute, and union is denoted by juxtaposition [9].

Theorem 1. *The following set $\mathfrak{R}$ of axioms and rules is sound and complete for the implication of dFDs in the RDM:*

(i) *the reflexivity axiom* $\dfrac{}{X \to (1)Y}\, Y \subseteq X$

(ii) *the weakening rule* $\dfrac{X \to (d+1)Y}{X \to (d)Y}$

(iii) *the strengthening rule* $\dfrac{X \to (d)Y}{X \to (d+1)Y}\ \mid X \mid < d$

(iv) *the union rule* $\dfrac{X \to (d)Y \quad X \to (d)Z}{X \to (d)YZ}$

(v) *the strong transitivity rule* $\dfrac{X \to (d)Y \quad YY' \to (d')Z}{X \to (d)Z}\ \mid Y' \mid < d'$

(vi) *the left strengthening rule* $\dfrac{X - A_1 \to (d)Y \ \cdots\ X - A_n \to (d)Y}{X \to (d+1)Y}\, X = \{A_1, \ldots, A_n\}$

(vii) *the left weakening rule* $\dfrac{X \to (d+1)Y}{X - A \to (d)Y}\, A \in X$

3 An Algebra of Nested Attributes

In this section we define our model of nested attributes, which covers the gist of higher-order datamodels including HERM, the OODM and XML. In particular, we investigate the structure of the set $\mathcal{S}(X)$ of subattributes of a given nested attribute X, which will give us a Brouwer algebra [6, 7].

3.1 Nested Attributes

We start with a definition of simple attributes and values for them.

Definition 3. A *universe* is a finite set $\mathcal{U}$ together with domains (i.e. sets of values) $dom(A)$ for all $A \in \mathcal{U}$. The elements of $\mathcal{U}$ are called *simple attributes*.

For the relational model a universe was enough, as a relation schema could be defined by a subset $R \subseteq \mathcal{U}$. For higher-order datamodels, however, we need nested attributes. In the following definition we use a set $\mathcal{L}$ of labels, and tacitly assume that the symbol λ is neither a simple attribute nor a label, i.e. $\lambda \notin \mathcal{U} \cup \mathcal{L}$, and that simple attributes and labels are pairwise different, i.e. $\mathcal{U} \cap \mathcal{L} = \emptyset$.

Definition 4. Let $\mathcal{U}$ be a universe and $\mathcal{L}$ a set of labels. 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 $X'_1, \ldots, X'_n \in \mathcal{N}$ we have $X(X'_1, \ldots, X'_n) \in \mathcal{N}$;
- 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*, $X(X'_1, \ldots, X'_n)$ a *record attribute*, $X\{X'\}$ a *set attribute*, $X[X']$ a *list attribute*, and $X\langle X'\rangle$ a *multiset attribute*. As record, set, list and multiset attributes have a unique leading label, say X, we often write simply X to denote the attribute.

We can now extend the association *dom* from simple to nested attributes, i.e. for each $X \in \mathcal{N}$ we will define a set of values $dom(X)$.

Definition 5. For each nested attribute $X \in \mathcal{N}$ we get a *domain* $dom(X)$ as follows:

- $dom(\lambda) = \{\top\}$;
- $dom(X(X'_1, \ldots, X'_n)) = \{(X_1 : v_1, \ldots, X_n : v_n) \mid v_i \in dom(X'_i) \text{ for } i = 1, \ldots, n\}$ with labels X_i for the attributes X'_i;
- $dom(X\{X'\}) = \{\{v_1, \ldots, v_n\} \mid v_i \in dom(X') \text{ for } i = 1, \ldots, n\}$, i.e. each element in $dom(X\{X'\})$ is a finite set with elements in $dom(X')$;
- $dom(X[X']) = \{[v_1, \ldots, v_n] \mid v_i \in dom(X') \text{ for } i = 1, \ldots, n\}$, i.e. each element in $dom(X[X'])$ is a finite list with elements in $dom(X')$;
- $dom(X\langle X'\rangle) = \{\langle v_1, \ldots, v_n\rangle \mid v_i \in dom(X') \text{ for } i = 1, \ldots, n\}$, i.e. each element in $dom(X\langle X'\rangle)$ is a finite multiset with elements in $dom(X')$.

Note that the relational model is covered, if only the tuple constructor is used. Thus, instead of a relation schema R we will now consider a nested attribute X, assuming that the universe $\mathcal{U}$ and the set of labels $\mathcal{L}$ are fixed. Instead of an R-relation r we will consider a finite set $r \subseteq dom(X)$.

3.2 Subattributes

In the dependency theory for the relational model we exploited projections on subsets X of a relation schema R. These are just special cases of projections on subattributes. Therefore, we will define a partial order $\geq$ on $\mathcal{N}$. However, this partial order will be defined on equivalence classes of attributes. We will identify nested attributes, if we can identify their domains.

Definition 6. $\equiv$ is the smallest *equivalence relation* on $\mathcal{N}$ satisfying the following properties:

- $\lambda \equiv X()$;
- $X(X'_1, \ldots, X'_n) \equiv X(X'_1, \ldots, X'_n, \lambda)$;
- $X(X'_1, \ldots, X'_n) \equiv X(X'_{\sigma(1)}, \ldots, X'_{\sigma(n)})$ for any permutation σ;

- $X(X'_1,\ldots,X'_n) \equiv X(Y_1,\ldots,Y_n)$ iff $X'_i \equiv Y_i$ for all $i = 1,\ldots,n$;
- $X\{X'\} \equiv X\{Y\}$ iff $X' \equiv Y$;
- $X[X'] \equiv X[Y]$ iff $X' \equiv Y$;
- $X\langle X'\rangle \equiv X\langle Y\rangle$ iff $X' \equiv Y$.

Basically, the equivalence definition states that λ in record attributes can be added or removed, and that order in record and union attributes does not matter.

In the following we identify $\mathcal{N}$ with the set $\mathcal{N}/_{\equiv}$ of equivalence classes. In particular, we will write $=$ instead of $\equiv$, and in the following definition we should say that Y is a subattribute of X iff $\tilde{X} \geq \tilde{Y}$ holds for some $\tilde{X} \equiv X$ and $\tilde{Y} \equiv Y$.

Definition 7. For $X, Y \in \mathcal{N}$ we say that Y is a *subattribute* of X, iff $X \geq Y$ holds, where $\geq$ is the smallest partial order on $\mathcal{N}$ satisfying the following properties:

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

Obviously, $X \geq Y$ induces a projection map $\pi^X_Y : dom(X) \to dom(Y)$. For $X \equiv Y$ we have $X \geq Y$ and $Y \geq X$ and the projection maps π^X_Y and π^Y_X are inverse to each other.

We use the notation $\mathcal{S}(X) = \{Z \in \mathcal{N} \mid X \geq Z\}$ to denote the *set of subattributes* of a nested attribute X. It has been shown that $\mathcal{S}(X)$ carries the structure of a Brouwer algebra [5, 6, 7].

Proposition 2. *The set $\mathcal{S}(X)$ of subattributes carries the structure of a Brouwer algebra, i.e. it is a distributive lattice with a meet-operation $\sqcap$, a join-operation $\sqcup$, a smallest element λ, a largest element X, and relative pseudo-complements $Y \leftarrow Z = \bigsqcap\{U \mid U \cup Y \geq Z\}$.*

Figure 1 as an example shows the Brouwer algebra $\mathcal{S}(X(X_1\{A\}, X_2[B]))$.

3.3 Ideals of Subattributes

We are dealing with several constructors for complex values at the same time. In order to cope with the problems arising from this fact, we need some additional notions that we will define in this subsection.

For the derivation rules for functional dependencies we need a notion of when two subattributes are "nearly disjoint". This property is called *semi-disjointness*.

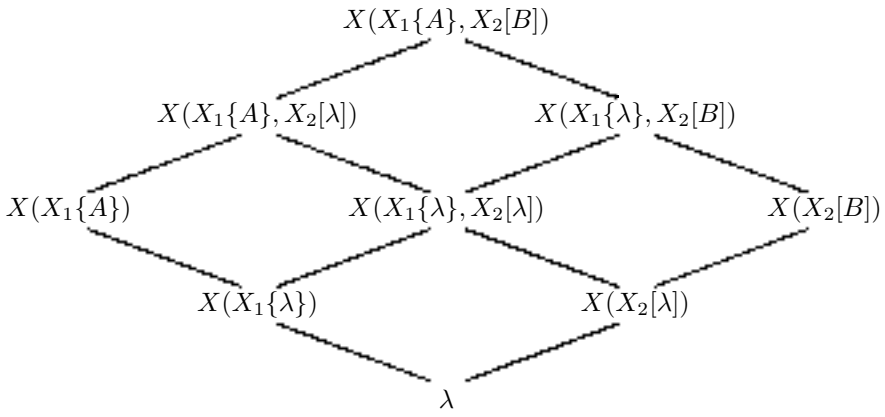

Figure 1: The lattice $\mathcal{S}(X(X_1\{A\}, X_2[B]))$

Definition 8. Two subattributes $Y, Z \in \mathcal{S}(X)$ are called *semi-disjoint* iff one of the following holds:

(i) $Y \geq Z$ or $Z \geq Y$;

(ii) $X = X(X_1, \ldots, X_n)$, $Y = X(Y_1, \ldots, Y_n)$, $Z = X(Z_1, \ldots, Z_n)$ and $Y_i, Z_i \in \mathcal{S}(X_i)$ are semi-disjoint for all $i = 1, \ldots, n$;

(iii) $X = X[X']$, $Y = X[Y']$, $Z = X[Z']$ and $Y', Z' \in \mathcal{S}(X')$ are semi-disjoint.

For the soundness proof in the next section we will need the following simple fact about projections to semi-disjoint attributes.

Lemma 1. *Let $t_1, t_2 \in dom(X)$ for some nested attribute $X \in \mathcal{N}$ such that $\pi^X_Y(t_1) = \pi^X_Y(t_2)$ and $\pi^X_Z(t_1) = \pi^X_Z(t_2)$ hold for semi-disjoint subattributes $Y, Z \in \mathcal{S}(X)$. Then also $\pi^X_{Y \sqcup Z}(t_1) = \pi^X_{Y \sqcup Z}(t_2)$ holds.*

Proof. We use induction on X to show $\pi^X_{Y \sqcup Z}(t_1) = \pi^X_{Y \sqcup Z}(t_2)$. The cases $X = \lambda$ and $X = A$ (i.e. a simple attribute) are trivial. There is also nothing to show for $Y \geq Z$ or $Z \geq Y$, as in these cases $Y \sqcup Z$ is one of Y or Z.

For $X = X(X_1, \ldots, X_n)$, semi-disjoint $Y = X(Y_1, \ldots, Y_n)$ and $Z = X(Z_1, \ldots, Z_n)$, and $t_j = (X_1 : t_{j1}, \ldots, X_n : t_{jn})$ $(j = 1, 2)$ we have $\pi^{X_i}_{Y_i}(t_{1i}) = \pi^{X_i}_{Y_i}(t_{2i})$ and $\pi^{X_i}_{Z_i}(t_{1i}) = \pi^{X_i}_{Z_i}(t_{2i})$, and Y_i, Z_i are semi-disjoint for all $i = 1, \ldots, n$. By induction $\pi^{X_i}_{Y_i \sqcup Z_i}(t_{1i}) = \pi^{X_i}_{Y_i \sqcup Z_i}(t_{2i})$, which implies $\pi^X_{Y \sqcup Z}(t_1) = \pi^X_{Y \sqcup Z}(t_2)$.

For $X = X[X']$ and semi-disjoint $Y = X[Y']$ and $Z = X[Z']$, the subattributes Y' and Z' of X' are also semi-disjoint. Furthermore, for $t_j = [t_{j1}, \ldots, t_{jn_j}]$ $(j = 1, 2)$ we must have $n_1 = n_2$ and $\pi^{X'}_{Y'}(t_{1k}) = \pi^{X'}_{Y'}(t_{2k})$ and $\pi^{X'}_{Z'}(t_{1k}) = \pi^{X'}_{Z'}(t_{2k})$ for all $k = 1, \ldots, n_1$. By induction we get also $\pi^{X'}_{Y' \sqcup Z'}(t_{1k}) = \pi^{X'}_{Y' \sqcup Z'}(t_{2k})$ for all $k = 1, \ldots, n_1$, i.e. $\pi^X_{Y \sqcup Z}(t_1) = \pi^X_{Y \sqcup Z}(t_2)$. □

As $\mathcal{S}(X)$ is a lattice, it makes sense to investigate ideals and filters. The following notion of an *HL-ideal* will be central for the completeness proof in the next section.

Definition 9. Let $X \in \mathcal{N}$. An *HL-ideal* on $\mathcal{S}(X)$ is a subset $\mathcal{F} \subseteq \mathcal{S}(X)$ with the following properties:

(i) $\lambda \in \mathcal{F}$;

(ii) if $Y \in \mathcal{F}$ and $Z \in \mathcal{S}(X)$ with $Y \geq Z$, then $Z \in \mathcal{F}$;

(iii) if $Y, Z \in \mathcal{F}$ are semi-disjoint, then $Y \sqcup Z \in \mathcal{F}$.

The key step in the completeness proof for dFDs in the RDM in [8] consists in the construction of a relation with exactly two tuples, which coincide exactly on a given set of attributes. While this is trivial for the RDM, the presence of complex values requires a similar result. However, instead of a set of attributes we now have to deal with an HL-ideal. This result — denoted *Central Lemma* in [6] — provides the major difficulty for the axiomatisation of functional dependencies in [5], [7] and [6].

The following theorem states this result for the case that we deal with records, lists, sets and multisets. The non-trivial lengthy proof is contained in [7].

Theorem 2. *Let $X \in \mathcal{N}$ and $\mathcal{F}$ be an* HL-ideal *on $\mathcal{S}(X)$. Then there exist $t_1, t_2 \in dom(X)$ with $\pi^X_Y(t_1) = \pi^X_Y(t_2)$ iff $Y \in \mathcal{F}$.*

4 Distance Functional Dependencies on Nested Attributes

Our major goal is to generalise Theorem 1 to dFDs on nested attributes. Therefore, we first have to generalise FDs and dFDs to this case. For the latter ones the major difficulty is to define a generalisation of the Hamming distance for complex values.

4.1 A Generalised Distance Function on Complex Values

Let us first define ordinary functional dependencies. As a set of attributes in the RDM corresponds to a single record attribute, the first idea is to replace sets of attributes by a single subattribute. While this is sufficient for records and lists, it is not a good idea for sets and multisets. The reason is that the well known extension rule in Armstrong's axiomatisation for FDs in the RDM does not generalise in this way [5]. Therefore, we have to consider sets of subattributes instead.

Definition 10. Let $X \in \mathcal{N}$. A *functional dependency* (FD) on $\mathcal{S}(X)$ is an expression $\mathcal{Y} \to \mathcal{Z}$ with $\mathcal{Y}, \mathcal{Z} \subseteq \mathcal{S}(X)$.

Let r be an instance of X. We say that r *satisfies the FD* $\mathcal{Y} \to \mathcal{Z}$ on $\mathcal{S}(X)$ (notation: $r \models \mathcal{Y} \to \mathcal{Z}$) iff for all $t_1, t_2 \in r$ with $\pi^X_Y(t_1) = \pi^X_Y(t_2)$ for all $Y \in \mathcal{Y}$ we also have $\pi^X_Z(t_1) = \pi^X_Z(t_2)$ for all $Z \in \mathcal{Z}$.

Now recall, that the difference between a FD and a dFD in the RDM was that we replaced the equality on the left hand side by a bound on the Hamming distance bewtwen two tuples with the distance $d = 1$ corresponding to the case of an ordinary FD. The Hamming distance counts the number of attributes, on which two tuples differ. These attributes form some kind of a "basis" in the sense that each subset of a relation schema can be constructed as a union of singleton sets containing just one attribute.

In order to generalise the distance notion to complex values, we therefore need a basis made of subattributes.

Definition 11. Let $X \in \mathcal{N}$. The *subattribute basis of* X is the smallest subset $\mathcal{B}(X) \subseteq \mathcal{S}(X)$ such that each $Y \in \mathcal{S}(X)$ can be written in the form $Y = \bigsqcup_{Y' \in \mathcal{B}_Y} Y'$ for some $\mathcal{B}_Y \subseteq \mathcal{B}(X)$.

The subattribute basis of a simple record attribute would just give us the simple attributes. Therefore, considering the subattribute basis suggests to be a good choice to replace the set of attributes in the definition of the distance function. However, in order to cope properly with sets ans multisets, we need to close $\mathcal{B}(X)$ under the join of attributes that are not semi-disjoint.

Definition 12. Let $X \in \mathcal{N}$. The *Hamming basis of* X is the smallest subset $\mathcal{C}(X) \subseteq \mathcal{S}(X)$ with $\mathcal{B}(X) \subseteq \mathcal{C}(X)$ such that for all non-semi-disjoint $Y, Z \in \mathcal{C}(X)$ we also have $Y \sqcup Z \in \mathcal{C}(X)$.

The following is an easy implication of Lemma 1.

Lemma 2. *Let $t_1, t_2 \in dom(X)$ for some nested attribute X and $Y \in \mathcal{X}$. If $\pi^X_{Y'}(t_1) = \pi^X_{Y'}(t_2)$ holds for all $Y' \in \mathcal{C}(Y)$, then also $\pi^X_Y(t_1) = \pi^X_Y(t_2)$ holds.*

Proof. According to the definition of the subattribute basis $\mathcal{B}(X)$ and the Hamming basis $\mathcal{C}(X)$ we can write Y in the form $Y = \bigsqcup_{Y' \in \mathcal{C}_Y} Y'$ for some subset $\mathcal{C}_Y \subseteq \mathcal{C}(X)$. By definition the elements in $\mathcal{C}_Y$ are pairwise semi-disjoint. As t_1 and t_2 coincide on all elements of $\mathcal{C}_Y$, the also coincide on Y by Lemma 1. □

Now we can use the Hamming basis of X to define the distance of two complex values $t_1, t_2 \in dom(X)$.

Definition 13. Let $X \in \mathcal{N}$ and $t_1, t_2 \in dom(X)$. The *Hamming distance* $\mathcal{H}(t_1, t_2)$ between t_1 and t_2 is defined as $\mathcal{H}(t_1, t_2) = |\{Y \in \mathcal{C}(X) \mid \pi^X_Y(t_1) \neq \pi^X_Y(t_2)\}|$, i.e. as the number of subattributes in the Hamming basis, on which t_1 and t_2 differ.

This leads us straightforward to the generalisation of dFDs on a nested attribute.

Definition 14. Let $X \in \mathcal{N}$ be a nested attribute and $d \geq 1$. A *d-distance functional dependency* (dFD) on $\mathcal{S}(X)$ is an expression of the form $\mathcal{Y} \to (d)\mathcal{Z}$ with $\mathcal{Y}, \mathcal{Z} \subseteq \mathcal{S}(X)$.

Let r be an instance of X. We say that r *satisfies the dFD* $\mathcal{Y} \to (d)\mathcal{Z}$ on $\mathcal{S}(X)$ (notation: $r \models \mathcal{Y} \to (d)\mathcal{Z}$) iff for all $t_1, t_2 \in r$ with $\mathcal{H}(\pi^X_Y(t_1), \pi^X_Y(t_2)) < d$ for all $Y \in \mathcal{Y}$ we also have $\pi^X_Z(t_1) = \pi^X_Z(t_2)$ for all $Z \in \mathcal{Z}$.

As before, we use $\models$ to denote implication of dFDs and Σ^* to denote the semantic hull of a set Σ of dFDs.

4.2 Sound Derivation Rules

Using the definitions from the last subsection we will show now that derivation rules similar to the ones in Theorem 1 are sound for the implication of dFDs on nested attributes. Before we can define this set of derivation rules, we need a few more notation.

For $Y \in \mathcal{Y} \subseteq \mathcal{S}(X)$ let $\downarrow Y = \{Y' \mid Y' \text{ maximal with } Y \geq Y'\}$. Furthermore, if $\mathcal{Y} = \{Y_1, \ldots, Y_k\}$, write $\downarrow \mathcal{Y} = \{\{Y_1', \ldots, Y_k'\} \mid Y_i' \in \downarrow Y_i \text{ for one } i \text{ and } Y_j' = Y_j \text{ for all } j \neq i\}$. In particular we have a mapping $Y_i \mapsto Y_i'$ and we can define $k_i = |\mathcal{C}(Y_i)| - |\mathcal{C}(Y_i')|$. We use this to define $k(\mathcal{Y}, \mathcal{Y}') = \max k_i$ for $\mathcal{Y}' = \{Y_1', \ldots, Y_k'\} \in \downarrow \mathcal{Y}$.

Theorem 3. *Let $X \in \mathcal{N}$ be a nested attribute. The following rules are sound for the implication of dFDs on $\mathcal{S}(X)$:*

reflexivity axiom: $$\frac{}{\mathcal{Y} \to (1)\mathcal{Z}}\ \mathcal{Z} \subseteq \mathcal{Y} \tag{1}$$

lambda axiom: $$\frac{}{\emptyset \to (d)\{\lambda\}} \tag{2}$$

subattribute axiom: $$\frac{}{\{Y\} \to (1)\{Z\}}\ Y \geq Z \tag{3}$$

join axiom: $$\frac{}{\{Y, Z\} \to (1)\{Y \sqcup Z\}}\ Y, Z \text{ semi-disjoint} \tag{4}$$

weakening rule: $$\frac{\mathcal{Y} \to (d+1)\mathcal{Z}}{\mathcal{Y} \to (d)\mathcal{Z}} \tag{5}$$

strengthening rule: $$\frac{\mathcal{Y} \to (d)\mathcal{Z}}{\mathcal{Y} \to (d+1)\mathcal{Z}}\ \max\{|\mathcal{C}(Y)| \mid Y \in \mathcal{Y}\} < d \tag{6}$$

union rule: $$\frac{\mathcal{Y} \to (d)\mathcal{Z}_1 \quad \mathcal{Y} \to (d)\mathcal{Z}_2}{\mathcal{Y} \to (d)\mathcal{Z}_1 \cup \mathcal{Z}_2} \tag{7}$$

strong transitivity rule: $$\frac{\mathcal{Y} \to (d)\mathcal{Z} \quad \mathcal{Z} \cup \mathcal{Z}' \to (d')\mathcal{U}}{\mathcal{Y} \to (d)\mathcal{U}}\ \max\{|\mathcal{C}(Z)| \mid Y \in \mathcal{Z}'\} < d' \tag{8}$$

left strengthening rule: $$\frac{\mathcal{Y}_1 \to (d-k_1)\mathcal{Z}\ \ldots\ \mathcal{Y}_m \to (d-k_m)\mathcal{Z}}{\mathcal{Y} \to (d)\mathcal{Z}} \tag{9}$$
for $\downarrow \mathcal{Y} = \{\mathcal{Y}_1, \ldots, \mathcal{Y}_m\}$ and $k_i = k(\mathcal{Y}, \mathcal{Y}_i)$

left weakening rule: $$\frac{\mathcal{Y} \to (d+k)\mathcal{Z}}{\mathcal{Y}' \to (d)\mathcal{Z}}\ \mathcal{Y}' \in \downarrow \mathcal{Y}, k = k(\mathcal{Y}, \mathcal{Y}') \tag{10}$$

Proof. In the following let r be an instance of X, i.e. $r \subseteq dom(X)$. The soundness of the weakening rule (5) is obvious.

For the soundness of the reflexivity axiom (1) let $t_1, t_2 \in r$ with $\mathcal{H}(\pi_Y^X(t_1), \pi_Y^X(t_2)) < 1$ for all $Y \in \mathcal{Y}$. That is, t_1 and t_2 coincide on all subattributes in $\mathcal{C}(Y)$ for all $Y \in \mathcal{Y}$. As $\mathcal{Z} \subseteq \mathcal{Y}$ holds, they must also coincide on all subattributes in $\mathcal{C}(Z)$ for all $Z \in \mathcal{Z}$.

The soundness of the lambda-axiom (2) is obvious, as any $t_1, t_2 \in r$ coincide on λ.

If t_1 and t_2 coincide on all subattributes in $\mathcal{C}(Y)$, they also coincide on Y by Lemma 2, and as $Y \geq Z$, they must also coincide on Z, which proves the soundness of the subattribute-axiom (3).

Similarly, $\mathcal{H}(\pi_Y^X(t_1), \pi_Y^X(t_2)) < 1$ and $\mathcal{H}(\pi_Z^X(t_1), \pi_Z^X(t_2)) < 1$ implies that t_1 and t_2 coincide on all subattributes in $\mathcal{C}(Y) \cup \mathcal{C}(Z)$. By Lemma 2 they must also coincide on Y and Z. As Y, Z are semi-disjoint, we obtain $\pi_{Y \sqcup Z}^X(t_1) = \pi_{Y \sqcup Z}^X(t_2)$ by Lemma 1, which proves the soundness of the join-axiom (4).

For the soundness of the strengthening rule (6) take $t_1, t_2 \in r$ with $\mathcal{H}(\pi_Y^X(t_1), \pi_Y^X(t_2)) < d + 1$ for all $Y \in \mathcal{Y}$. As $\mathcal{H}(\pi_Y^X(t_1), \pi_Y^X(t_2)) < |\mathcal{C}(Y)| < d$ for all $Y \in \mathcal{Y}$, the premise implies $\pi_Z^X(t_1) = \pi_Z^X(t_2)$ for all $Z \in \mathcal{Z}$ as claimed.

For the soundness of the union rule (7) take $t_1, t_2 \in r$ with $\mathcal{H}(\pi_Y^X(t_1), \pi_Y^X(t_2)) < d$ for all $Y \in \mathcal{Y}$. The premises of the rule imply $\pi_Z^X(t_1) = \pi_Z^X(t_2)$ for all $Z \in \mathcal{Z}_j$ $(j = 1, 2)$, which trivially implies $\pi_Z^X(t_1) = \pi_Z^X(t_2)$ for all $Z \in \mathcal{Z}_1 \cup \mathcal{Z}_2$.

In order to prove the soundness of the strong transitivity rule (8) take again $t_1, t_2 \in r$ with $\mathcal{H}(\pi_Y^X(t_1), \pi_Y^X(t_2)) < d$ for all $Y \in \mathcal{Y}$. The first premise of the rule implies $\pi_Z^X(t_1) = \pi_Z^X(t_2)$ for all $Z \in \mathcal{Z}$. For $Z' \in \mathcal{Z}'$ we have $\mathcal{H}(\pi_{Z'}^X(t_1), \pi_{Z'}^X(t_2)) < |\mathcal{C}(\mathcal{Z}')| < d'$. Hence $\mathcal{H}(\pi_Z^X(t_1), \pi_Z^X(t_2)) < d'$ for all $Z \in \mathcal{Z} \cup \mathcal{Z}'$. The second premise of the rule gives the desired $\pi_U^X(t_1) = \pi_U^X(t_2)$ for all $U \in \mathcal{U}$.

Now take again $t_1, t_2 \in r$ with $\mathcal{H}(\pi_Y^X(t_1), \pi_Y^X(t_2)) < d$ for all $Y \in \mathcal{Y}$. Unless $\pi_Y^X(t_1) = \pi_Y^X(t_2)$ there must exist some $\mathcal{Y}_i \in\downarrow \mathcal{Y}$ with $\mathcal{H}(\pi_{Y'}^X(t_1), \pi_{Y'}^X(t_2)) < d - k_i$ for all $Y' \in \mathcal{Y}_i$, and we can apply the corresponding premise of the left strengthening rule (9) to conclude $\pi_Z^X(t_1) = \pi_Z^X(t_2)$ for all $Z \in \mathcal{Z}$, which proves the soundness of this rule.

Finally, for the soundness of the left weakening rule (10) take again $t_1, t_2 \in r$ with $\mathcal{H}(\pi_{Y'}^X(t_1), \pi_{Y'}^X(t_2)) < d$ for all $Y' \in \mathcal{Y}'$. Hence, $\mathcal{H}(\pi_Y^X(t_1), \pi_Y^X(t_2)) < d + 1$ for all $Y \in \mathcal{Y}$. Applying the premise of the rule leads to $\pi_Z^X(t_1) = \pi_Z^X(t_2)$ for all $Z \in \mathcal{Z}$ as claimed. □

4.3 Completeness

As usual, given a set of axioms and rules $\mathfrak{R}$, and a set Σ of dFDs, we let Σ^+ denote the *syntactic hull* of Σ, i.e. the set of all dFDs that can be derived from Σ using the axioms and rules in $\mathfrak{R}$. In the following we take $\mathfrak{R}$ as the axioms and rules from Theorem 3. This theorem already states the soundness of $\mathfrak{R}$, i.e. $\Sigma^+ \subseteq \Sigma^*$.

A set of axioms and rules is called *complete* iff $\Sigma^* \subseteq \Sigma^+$ holds. Our final goal is to show the completeness of the rules in $\mathfrak{R}$. Theorem 2 will turn out to be the key for the completeness proof in this section.

Theorem 4. *The set $\mathfrak{R}$ of axioms and rules from Theorem 3 is complete for the implication of dFDs on nested attributes.*

Proof. Let $X \in \mathcal{N}$ be a nested attribute, and let Σ denote a set of dFDs on $\mathcal{S}(X)$. In order to show $\Sigma^* \subseteq \Sigma^+$ let $\mathcal{Y} \to (d)\mathcal{Z} \notin \Sigma^+$.

Let d be minimal with this property. Then, according to rule (6) we can assume that $|\mathcal{C}(Y)| \geq d-1$ for at least one $Y \in \mathcal{Y}$. Otherwise, we would have $\max\{|\mathcal{C}(Y)| \mid Y \in \mathcal{Y}\} < d-1$. As d is minimal, we have $\mathcal{Y} \to (d-1)\mathcal{Z} \in \Sigma^+$, and applying the strengthening rule (6) would result in the contradiction $\mathcal{Y} \to (d)\mathcal{Z} \in \Sigma^+$.

Due to the union rule (7) there must be some $Z \in \mathcal{Z}$ with $\mathcal{Y} \to (d)\{Z\} \notin \Sigma^+$. Then, applying the left strengthening rule (9) k times with $k \leq d-1$ — which is possible, as $|\mathcal{C}(Y)| \geq d-1$ for at least one $Y \in \mathcal{Y}$ — we find some $\mathcal{Y}' \in{\downarrow^k}\, \mathcal{Y}$ with $\mathcal{Y}' \to (1)\{Z\} \notin \Sigma^+$.

Now take ${\mathcal{Y}'}^+ = \{U \mid \mathcal{Y}' \to (1)\{U\} \in \Sigma^+\}$, so $Z \notin {\mathcal{Y}'}^+$, but due to the reflexivity axiom (1) we have $\mathcal{Y}' \subseteq {\mathcal{Y}'}^+$.

Obviously, due to the lambda axiom (2), the subattribute axiom (3) and the join axiom (4) ${\mathcal{Y}'}^+$ is an HL-ideal in the Brouwer algebra $\mathcal{S}(X)$. Applying Theorem 2 to ${\mathcal{Y}'}^+$ we obtain an instance $r = \{t_1, t_2\}$ such that $\pi_U^X(t_1) = \pi_U^X(t_2)$ holds iff $U \in {\mathcal{Y}'}^+$.

Hence, $r \not\models \mathcal{Y}' \to (1)\{Z\}$, and applying the sound left weakening rule (10) k times we obtain $r \not\models \mathcal{Y} \to (d)\{Z\}$. From the soundness of the reflexivity axiom (1) we further obtain $r \not\models \mathcal{Y} \to (d)\mathcal{Z}$.

We now show $r \models \Sigma$. So let $\mathcal{U} \to (d')\mathcal{V} \in \Sigma$. We consider two cases:

(i) Assume $\max\{|\mathcal{C}(U)| \mid U \in \mathcal{U}\} < d'$. Due to the lambda rule (2) we have $\mathcal{Y}' \to (1)\{\lambda\} \in \Sigma^+$. Using the strong transitivity rule (8) with $\mathcal{Z}' = \mathcal{U}$, we obtain $\mathcal{Y}' \to (1)\mathcal{V} \in \Sigma^+$, hence $\mathcal{V} \subseteq {\mathcal{Y}'}^+$. Due to the construction of r we obtain $\pi_V^X(t_1) = \pi_V^X(t_2)$ for all $V \in \mathcal{V}$, which shows $r \models \mathcal{U} \to (d')\mathcal{V}$.

(ii) Next assume $\max\{|\mathcal{C}(U)| \mid U \in \mathcal{U}\} \geq d'$. We show $r \models \mathcal{U}' \to (1)\mathcal{V}$, whenever $\mathcal{U}' \in{\downarrow^{k'}}\, \mathcal{U}$ with $k' \leq d'-1$, and $\mathcal{U}' \to (1)\mathcal{V} \in \Sigma^+$ results from applying the left weakening rule (10) k' times.

Then the soundness of the left strengthening rule (9) implies $r \models \mathcal{U} \to (d')\mathcal{V}$ as desired.

We distinguish again two subcases:

(a) If $\mathcal{U}' \not\subseteq {\mathcal{Y}'}^+$, we have $\pi_{U'}^X(t_1) \neq \pi_{U'}^X(t_2)$ for at least one $U' \in \mathcal{U}'$, which immediately implies $r \models \mathcal{U}' \to (1)\mathcal{V}$.

(b) If $\mathcal{U}' \subseteq {\mathcal{Y}'}^+$, we have $\mathcal{Y}' \to (1)\{U'\} \in \Sigma^+$ for all $U' \in \mathcal{U}'$. Using the union rule (7) we conclude $\mathcal{Y}' \to (1)\mathcal{U}' \in \Sigma^+$, and further $\mathcal{Y}' \to (1)\mathcal{V} \in \Sigma^+$ by applying the strong transitivity rule (8).
Hence $\mathcal{V} \subseteq {\mathcal{Y}'}^+$, which implies $r \models \mathcal{U}' \to (1)\mathcal{V}$ as desired.

Now $r \models \Sigma^*$, but $r \not\models \mathcal{Y} \to (d)\mathcal{Z}$. Hence $\mathcal{Y} \to (d)\mathcal{Z} \notin \Sigma^*$, which completes the proof. □

5 Conclusion

In this article we presented a finite axiomatisation of distance functional dependencies on nested attributes. This result generalises a corresponding result for the RDM that was achieved (in a more general form) in [8].

The major tasks to solve this problem were generalising the Hamming distance from tuples to arbitrary complex values, and constructing values that coincide exactly on a given ideal of subattributes. The latter problem was solved in [7] with solutions to a subcase contained in [5].

For the generalisation of the Hamming distance we used the "Hamming basis", which results from the subattribute basis by adding the joins of all non-semi-disjoint subattributes. This preserves the Hamming distance on flat tuples as a special case. That is, the new Hamming distance counts the number of subattributes in the Hamming basis, on which two values differ.

Alternatively, we could have chosen all subattributes instead of just those in the Hamming basis. Looking through the proofs in this article, this would not have affected the finite axiomatisation. However, we would have obtained a distance function with significant jumps.

We might still feel that the new distance function is still too coarse, as it cannot express counting. For instance, two sets with elements in the domain of a simple attribute either have distance 0, i.e. they are equal, or 1, i.e. they are different but both non-empty, or 2, i.e. they are different and one of thre sets is empty. However, the same problem appears already with functional dependencies, and thus, has to be solved in a larger context.

References

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

[2] W.W. Armstrong. Dependency Structures of Database Relationships. *Information Processing* vol. 74: 580-583, 1974.

[3] J. Demetrovics, G.O.H. Katona, D. Miklós. Error-Correcting Keys in Relational Databases. in K.-D. Schewe, B. Thalheim (Eds.). *Foundations of Information and Knowledge Systems. First International Symposium, FoIKS 2000*. Springer-Verlag, LNCS vol. 1762: 88-93. Berlin 2000.

[4] J. Demetrovics, G.O.H. Katona, D. Miklós. Functional Dependencies in Presence of Errors. in T. Eiter, K.-D. Schewe (Eds.). *Foundations of Information and Knowledge Systems. Second International Symposium, FoIKS 2002*. Springer-Verlag, LNCS vol. 2284: 85-92. Berlin 2002.

[5] S. Hartmann, A. Hoffmann, S. Link, K.-D. Schewe. Axiomatizing Functional Dependencies in the Higher-Order Entity-Relationship Model. *Information Processing Letters* vol. 87 (2003): 133-137.

[6] S. Hartmann, S. Link, K.-D. Schewe. Weak Functional Dependencies in Higher-Order Datamodels – The Case of the Union Constructor. in D. Seipel, J. M. Turull Torres (Eds.). *Foundations of Information and Knowledge Systems. Third International Symposium, FoIKS 2004*. Springer-Verlag LNCS vol. 2942: 117-134. Berlin 2004.

[7] S. Hartmann, S. Link, K.-D. Schewe. Axiomatisation of Functional Dependencies in the Presence of Records, Lists, Sets and Multisets. Massey University 2003. submitted for publication.

[8] S. Hartmann, S. Link, K.-D. Schewe, B.Thalheim. Error-Robust Functional Dependencies. Massey University 2002. submitted for publication.

[9] J. Paredaens, P. De Bra, M. Gyssens, D. Van Gucht. *The Structure of the Relational Database Model*. EATCS Monographs on Theoretical Computer Science. Springer-Verlag, Berlin Heidelberg 1989.

[10] K.-D. Schewe and B. Thalheim. Fundamental concepts of object oriented databases. *Acta Cybernetica* vol. 11 (4): 49-85, 1993.

[11] B. Thalheim. *Entity-Relationship Modeling: Foundations of Database Technology*. Springer-Verlag, Berlin Heidelberg 2000.

Received October, 2002