Approximate Keys and Functional Dependencies in Incomplete Databases With Limited Domains–Algorithmic Perspective

Munqath Alatar
ITRDC, University of Kufa, Iraq
munqith.alattar@uokufa.edu.iq

Attila Sali
Alfréd Rényi Institute of Mathematics
and Department of Computer Science,
Budapest University of Technology and Economics
Budapest, Hungary
sali.attila@renyi.hu

Keywords: Strongly possible functional dependencies, Strongly possible keys, incomplete databases, approximate functional dependencies, approximate keys.

Received: February 1, 2023

A possible world of an incomplete database table is obtained by imputing values from the attributes (infinite) domain to the place of `NULL` *s. A table satisfies a possible key or possible functional dependency constraint if there exists a possible world of the table that satisfies the given key or functional dependency constraint. A certain key or functional dependency is satisfied by a table if all of its possible worlds satisfy the constraint. Recently, an intermediate concept was introduced. A strongly possible key or functional dependency is satisfied by a table if there exists a strongly possible world that satisfies the key or functional dependency. A strongly possible world is obtained by imputing values from the active domain of the attributes, that is from the values appearing in the table. In the present paper, we study approximation measures of strongly possible keys and FDs. Measure g_3 is the ratio of the minimum number of tuples to be removed in order that the remaining table satisfies the constraint. We introduce a new measure g_5, the ratio of the minimum number of tuples to be added to the table so the result satisfies the constraint. g_5 is meaningful because the addition of tuples may extend the active domains. We prove that if g_5 can be defined for a table and a constraint, then the g_3 value is always an upper bound of the g_5 value. However, the two measures are independent of each other in the sense that for any rational number $0 \leq \frac{p}{q} < 1$ there are tables of an arbitrarily large number of rows and a constant number of columns that satisfy $g_3 - g_5 = \frac{p}{q}$. A possible world is obtained usually by adding many new values not occurring in the table before. The measure g_5 measures the smallest possible distortion of the active domains. We study complexity of determining these approximate measures.*

Povzetek:

1 Introduction

The information in many industrial and research databases may usually be incomplete due to many reasons. For example, databases related to instrument maintenance, medical applications, and surveys [8]. This makes it necessary to handle the cases when some information missing from a database and are required by the user. Imputation (filling in) is one of the common ways to handle the missing values [13].

A new approach for imputing values in place of the missing information was introduced in [3], to achieve complete data tables, using only information already contained in the SQL table attributes (which are called the active domain of an attribute). Any total table obtained in this way is called a *strongly possible world*. We use only the data shown on the table to replace the missing information because in many cases there is no proper reason to consider any other attribute values than the ones that already exist in the table. Using this concept, new key and functional dependency constraints called strongly possible keys (spKeys) and strongly possible functional dependencies (spFDs) were defined in [5, 4] that are satisfied after replacing any missing value (`NULL`) with a value that is already shown in the corresponding attribute. In Section 2, we provide the formal definitions of spKeys and spFDs.

The present paper continues the work started in [5], where an approximation notion was introduced to calculate how close any given set of attributes can be considered as a key. Tuple removal may be necessary because the active domains do not contain enough values to be able to replace the `NULL` values so that the tuples are pairwise distinct on a candidate key set of attributes K. In the present paper, we study approximation measures of spKeys and spFDs by adding tuples. Adding a tuple with new unique values will add more values to the attributes' active domains, thus some unsatisfied constraints may get satisfied.

For example, Car_Model and $DoorNo$ is designed to form a key in the Cars Types table shown in Table 1 but the table does not satisfy the spKey $sp\langle Car\_Model, DoorNo\rangle$. Two tuples would need to be removed, but adding a new tuple with distinct door number value to satisfy $sp\langle Car\_Model, DoorNo\rangle$ is better than removing two tuples. In addition to that, we know that the car model and door number determines the engine type, then the added tuple can also have a new value in the $DoorNo$ attribute so that the table satisfy $(Car\_Model, DoorNo) \rightarrow_{sp} Engine\_Type$ rather than removing other two tuples.

Table 1: Cars Types Incomplete Table

Car_Model	**Door No**	**Engine_Type**
BMW I3	4 doors	$\perp$
BMW I3	$\perp$	electric
Ford explorer	$\perp$	V8
Ford explorer	$\perp$	V6

2 Definitions

Let $R = \{A_1, A_2, \dots A_n\}$ be a relation schema. The set of all the possible values for each attribute $A_i \in R$ is called the domain of A_i and denoted as $D_i = dom(A_i)$ for i = 1,2,... n. Then, for $X \subseteq R$, then $D_X = \prod_{\forall A_i \in K} D_i$.

An instance $T = (t_1, t_2, \dots t_s)$ over R is a list of tuples such that each tuple is a function $t : R \to \bigcup_{A_i \in R} dom(A_i)$ and $t[A_i] \in dom(A_i)$ for all A_i in R. By taking a list of tuples we use the *bag semantics* that allows several occurrences of the same tuple. Usage of the bag semantics is justified by that SQL allows multiple occurrences of tuples. Of course, the order of the tuples in an instance is irrelevant, so mathematically speaking we consider a *multiset of tuples* as an instance. For a tuple $t_r \in T$ and $X \subset R$, let $t_r[X]$ be the restriction of t_r to X.

It is assumed that $\perp$ is an element of each attribute's domain that denotes missing information. t_r is called V-total for a set V of attributes if $\forall A \in V$, $t_r[A] \neq \perp$. Also, t_r is a total tuple if it is R-total. t_1 and t_2 are *weakly similar* on $X \subseteq R$ denoted as $t_1[X] \sim_w t_2[X]$ defined by Köhler et.al. [12] if

$$\forall A \in X \quad (t_1[A] = t_2[A] \text{ or } t_1[A] = \perp \text{ or } t_2[A] = \perp).$$

Furthermore, t_1 and t_2 are *strongly similar* on $X \subseteq R$ denoted by $t_1[X] \sim_s t_2[X]$ if

$$\forall A \in X \quad (t_1[A] = t_2[A] \neq \perp).$$

For the sake of convenience we write $t_1 \sim_w t_2$ if t_1 and t_2 are weakly similar on R and use the same convenience for strong similarity. Let $T = (t_1, t_2, \dots t_s)$ be a table instance over R. Then, $T' = (t'_1, t'_2, \dots t'_s)$ is a *possible world* of T, if $t_i \sim_w t'_i$ for all $i = 1, 2, \dots s$ and T' is completely `NULL`-free. That is, we replace the occurrences of $\perp$ with a value from the domain D_i different from $\perp$ for all tuples and all attributes. Active domain of an attribute is the set of all the distinct values shown under the attribute except the `NULL`. Note that this was called the *visible domain* of the attribute in papers [3, 4, 5, 2].

Definition 1 *The* active domain *of an attribute A_i (VD_i^T) is the set of all distinct values except*

$\perp$ *that are already used by tuples in* T*:*

$$VD_i^T = \{t[A_i] : t \in T\} \setminus \{\perp\} \text{ for } A_i \in R.$$

To simplify notation, we omit the upper index T if it is clear from the context what instance is considered.

While a possible world is obtained by using the domain values instead of the occurrence of `NULL`, a strongly possible world is obtained by using the active domain values.

Definition 2 *A possible world* T' *of* T *is called a* strongly possible world (spWorld) *if* $t'[A_i] \in VD_i^T$ *for all* $t' \in T'$ *and* $A_i \in R$*.*

The concept of *strongly possible world* was introduced in [3]. Strongly possible worlds allow us to define *strongly possible keys (spKeys)* and *strongly possible functional dependencies (spFDs).*

Definition 3 *A strongly possible functional dependency, in notation* $X \rightarrow_{sp} Y$*, holds in table* T *over schema* R *if there exists a strongly possible world* T' *of* T *such that* $T' \models X \rightarrow Y$*. That is, for any* $t'_1, t'_2 \in T'$ $t'_1[X] = t'_2[X]$ *implies* $t'_1[Y] = t'_2[Y]$*. The set of attributes* X *is a strongly possible key, if there exists a strongly possible world* T' *of* T *such that* X *is a key in* T'*, in notation* $sp\langle X\rangle$*. That is, for any* $t'_1, t'_2 \in T'$ $t'_1[X] = t'_2[X]$ *implies* $t'_1 = t'_2$*.*

If $T = \{t_1, t_2, \ldots, t_p\}$ and $T' = \{t'_1, t'_2, \ldots, t'_p\}$ is an spWorld of it with $t_i \sim_w t'_i$, then t'_i is called an *sp-extension* or in short an *extension* of t_i. Let $X \subseteq R$ be a set of attributes and let $t_i \sim_w t'_i$ such that for each $A \in R$: $t'_i[A] \in VD(A)$, then $t'_i[X]$ is an *strongly possible extension of* t_i *on* X *(sp-extension)*

3 Related Work

Kivinen et. al. [11] introduced the measure g_3 for total tables. Giannella et al. [9] measure the approximate degree of functional dependencies. They developed the IFD approximation measure and compared it with the other two measures: g_3 (minimum number of tuples need to be removed so that the dependency holds) and τ (the probability of a correct guess of an FD satisfaction) introduced in [11] and [10] respectively. They developed analytical bounds on the measure differences and compared these measures analysis on five datasets. The authors show that when measures are meant to define the knowledge degree of X determines Y (prediction or classification), then IFD and τ measures are more appropriate than g_3. On the other hand, when measures are meant to define the number of "violating" tuples in an FD, then, g_3 measure is more appropriate than IFD and τ.

In [15], Jef Wijsen summarizes and discusses some theoretical developments and concepts in Consistent query answering CQA (when a user queries a database that is inconsistent with respect to a set of constraints). Database repairing was modeled by an acyclic binary relation $\leq_{db}$ on the set of consistent database instances, where $r_1 \leq_{db} r_2$ means that r_1 is at least as close to db as r_2. One possible distance is the number of tuples to be added and/or removed. In addition to that, Bertossi studied the main concepts of database repairs and CQA in [6], and emphasis on tracing back the origin, motivation, and early developments. J. Biskup and L. Wiese present and analyze an algorithm called preCQE that is able to correctly compute a solution instance, for a given original database instance, that obeys the formal properties of inference-proofness and distortion minimality of a set of appropriately formed constraints in [7].

4 Approximation of strongly possible integrity constraints

Definition 4 *Attribute set* K *is an approximate strongly possible key of ratio* a *in table* T*, in notation* $asp_a^- \langle K\rangle$*, if there exists a subset* S *of the tuples* T *such that* $T \setminus S$ *satisfies* $sp\langle K\rangle$*, and* $|S|/|T| \leq a$*. The minimum* a *such that* $asp_a^- \langle K\rangle$ *holds is denoted by* $g_3(K)$*.*

The measure $g_3(K)$ has a value between 0 and 1, and it is exactly 0 when $sp\langle K\rangle$ holds in T, which means we don't need to remove any tuples. For this, we used the g_3 measure introduced in [11], to determine the degree to which ASP key is approximate. For example, the g_3 measure of $sp\langle X\rangle$ on Table 2 is 0.5, as we are required to remove two out of four tuples to satisfy the key constraint as shown in Table 3.

The g_3 approximation measure for spKeys was introduced in [5]. In this section, we introduce a new approximation measure for spKeys.

Definition 5 *Attribute set K is an add-approximate strongly possible key of ratio b in table T, in notation $asp_b^+ \langle K \rangle$, if there exists a set of tuples S such that the table TS satisfies $sp \langle K \rangle$, and $|S|/|T| \leq b$. The minimum b such that $asp_b^+ \langle K \rangle$ holds is denoted by $g_5(K)$.*

The measure $g_5(K)$ represents the approximation which is the ratio of the number of tuples needed to be added over the total number of tuples so that $sp \langle K \rangle$ holds. The value of the measure $g_3(K)$ ranges between 0 and 1, and it is exactly 0 when $sp \langle K \rangle$ holds in T, which means we do not have to add any tuple. For example, the g_5 measure of $sp\langle X \rangle$ on Table 2 is 0.25, as it is enough to add one tuple to satisfy the key constraint as shown in Table 4.

Definition 6 *For the attribute sets X and Y, $\sigma : X \to_{sp} Y$ is a remove-approximate strongly possible functional dependency of ratio a in a table T, in notation*
$T \models \approx_a^- X \to_{sp} Y$, if there exists a set of tuples S such that the table $T \setminus S \models X \to_{sp} Y$, and $|S|/|T| \leq a$. Then, $g_3(\sigma)$ is the smallest a such that $T \models \approx_a^- \sigma$ holds.

Definition 7 *For the attribute sets X and Y, $\sigma : X \to_{sp} Y$ is an add-approximate strongly possible functional dependency of ratio b in a table T, in notation $T \models \approx_b^+ X \to_{sp} Y$, if there exists a set of tuples S such that the table $T \cup S \models X \to_{sp} Y$, and $|S|/|T| \leq b$. Then, $g_5(\sigma)$ is the smallest b such that $T \models \approx_b^+ \sigma$ holds.*

Let T be a table and $U \subseteq T$ be the set of the tuples that we need to remove so that the spKey holds in T, i.e, we need to remove $|U|$ tuples, while by adding a tuple with new values, we may make more than one of the tuples in U satisfy the spKey using the new added values for their `NULL`s. In other words, we may need to add a fewer number of tuples than the number of tuples we need to remove to satisfy an spKey in the same given table. For example, Table 2 requires removing two tuples to satisfy $sp \langle X \rangle$, while adding one tuple is enough.

Table 2: Incomplete Table to measure $sp\langle X \rangle$

X	
X_1	X_2
$\perp$	1
2	$\perp$
2	$\perp$
2	2

Table 3: The table after removing $(asp_a^- \langle X \rangle)$

X	
X_1	X_2
$\perp$	1
2	2

4.1 Relation between g_3 and g_5 measures

Results together with their proofs of this subsection were reported in the conference volume [1], so the proofs are not included here. The following Proposition is used to prove Proposition 2.

Proposition 1 *Let T be an instance over schema R and let $K \subseteq R$. If the K-total part of the table T satisfies the key $sp \langle K \rangle$, then there exists a minimum set of tuples U to be removed that are all non-K-total so that $T \setminus U$ satisfies $sp \langle K \rangle$.*

Proposition 2 *For any $K \subseteq R$ with $|K| \geq 2$, we have $g_3(K) \geq g_5(K)$.*

Apart form the previous inequality, the two measures are totally independent for spKeys.

Table 4: The table after adding $(asp_b^+ \langle X \rangle)$

X	
X_1	X_2
$\perp$	1
2	$\perp$
2	$\perp$
2	2
3	3

Theorem 1 *Let $0 \leq \frac{p}{q} < 1$ be a rational number. Then there exist tables over schema $\{A_1, A_2\}$ with arbitrarily large number of rows, such that $g_3(\{A_1, A_2\}) - g_5(\{A_1, A_2\}) = \frac{p}{q}$.*

Unfortunately, the analogue of Proposition 1 is not true for spFDs, so the proof of the following theorem is quiet involved.

Theorem 2 *Let T be a table over schema R, $\sigma: X \to_{sp} Y$ for some $X, Y \subseteq R$. Then $g_3(\sigma) \geq g_5(\sigma)$.*

Theorem 3 can be proven by a construction similar to the proof of Theorem 1.

Theorem 3 *For any rational number $0 \leq \frac{p}{q} < 1$ there exists tables with an arbitrarily large number of rows and bounded number of columns that satisfy $g_3(\sigma) - g_5(\sigma) = \frac{p}{q}$ for $\sigma: X \to_{sp} Y$.*

4.2 Complexity problems

Definition 8 *The* SPKey problem *is the following.*
Input Table T over schema R and $K \subseteq R$.
Question Is it true that $T \models sp\langle K\rangle$?
The SPKeySystem problem *is the following.*
Input Table T over schema R and $\mathcal{K} \subseteq 2^R$.
Question Is it true that $T \models sp\langle \mathcal{K}\rangle$?
The SPFD problem *is the following.*
Input Table T over schema R and $X, Y \subseteq R$.
Question Is it true that $T \models X \to_{sp} Y$?

The following was shown in [4].

Theorem 4 *SPKey∈P, SPkeySystem and SPFD are NP-complete*

However, the approximation measures raise new, interesting algorithmic questions.

Definition 9 *The* SpKey-g3 *problem is the following.*
Input Table T over schema R, $K \subseteq R$ and $0 \leq q < 1$.
Question Is it true that $g_3(K) \leq q$ in table T?
The SpKey-g5 *problem is the following.*
Input Table T over schema R, $K \subseteq R$ and $0 \leq q < 1$.
Question Is it true that $g_5(K) \leq q$ in table T?

Proposition 3 *The decision problem SpKey-g5 is in P.*

Proof: Let us assume that tuples $s_i: i = 1, 2, \ldots, p$ over schema R are such that $T \cup \{s_1, s_2, \ldots s_p\}$ is optimal, so $g_5(K) = \frac{p}{m}$. Then clearly we may replace s_i by $s'_i = (z_i, z_i, \ldots, z_i)$ for all $i = 1, 2, \ldots, p$ where z_i's are pairwise distinct new values not appearing in the (extended) table $T \cup \{s_1, s_2, \ldots s_p\}$ so that $T \cup \{s'_1, s'_2, \ldots s'_p\} \models sp\langle K\rangle$. Thus, if $g_5(K) \leq q$ is needed to be checked for a table T of m tuples, one may add $\lfloor q \cdot m \rfloor$ completely new tuples to obtain table T' and check whether $T' \models sp\langle K\rangle$ in polynomial time by Theorem 4. $\square$

Theorem 5 *Decision problem SpKey-g3 is in P.*

Proof: Consider the relational schema R and $K \subseteq R$. Furthermore, let T be an instance over R with `NULL`s. Let T' be the set of total tuples $T' = \{t' \in \Pi_{A \in K} VD^T(A): \exists t \in T$ such that $t[K] \sim_w t'[K]\}$, furthermore let $G = (T, T'; E)$ be the bipartite graph, called the *K-extension graph of T*, defined by $\{t, t'\} \in E \iff t[K] \sim_w t'[K]$. Finding a matching of G that covers all the tuples in T (if exists) provides the set of tuples in T' to replace the incomplete tuples in T with, to verify that K is an spKey.

It was shown in [5] that the g_3 approximation measure for strongly possible keys satisfies

$$g_3(K) = \frac{|T| - \nu(G)}{|T|}.$$

where $\nu(G)$ denotes the maximum matching size in the K-extension graph G. However, the size of G is usually exponential function of the size of the input of the decision problem SpKey-g3, as T' is usually exponentially large.

In order to make our algorithm run in polynomial time we only generate part of T'. Let $T = \{t_1, t_2 \ldots t_m\}$ and $\ell(t_i) = |\{t^\star \in \Pi_{A \in K} VD^T(A): t^\star \sim_w t_i[K]\}|$. Note that $\ell(t_i) = \prod_{A:\, t_i[A] = \perp} |VD^T(A)|$, hence these values can be calculated by scanning T once and using appropriate search tree data structures to hold values of active domains of each attribute. Sort tuples of T in non-decreasing $\ell(t_i)$ order, i.e. assume that $\ell(t_1) \leq \ell(t_2) \leq \ldots \leq \ell(t_m)$. Let $j = \max\{i: \ell(t_i) < i\}$ and $T_j = \{t_1, t_2, \ldots t_j\}$, furthermore $T^\star_j = \{t^\star : \exists t \in T_j : t^\star \sim_w t[K]\} \subseteq \Pi_{A \in K} VD^T(A)$. Note that $|T^\star_j| \leq \frac{1}{2} j(j-1)$. If $\forall i = 1, 2, \ldots, m: \ell(t_i) \geq i$, then define $j = 0$ and $T^\star_j = \emptyset$. Let $G^\star = (T_j, T^\star_j; E^\star)$ be the induced

subgraph of G on the vertex set $T_j \cup T_j^\star$. Note that $|T_j^\star| \le \frac{1}{2}j(j-1)$.
Claim $\nu(G) = \nu(G^\star) + |T \setminus T_j|$.
Proof of Claim: The inequality $\nu(G) \le \nu(G^\star) + |T \setminus T_j|$ is straightforward. On the other hand, a matching of size $\nu(G^\star)$ in $G^\star$ can greedily be extended to the vertices in $|T \setminus T_j|$, as $t_i \in T \setminus T_j$ has at least i neighbours (which can be generated in polynomial time).

Thus it is enough to determine $\nu(G^\star)$ in order to calculate $g_3(K)$, and that can be done in polynomial time using Augmenting Path method [14]. $\square$ Note that the proof above shows that the *exact value* of $g_3(K)$ can be determined in polynomial time. This gives the following corollary.

Definition 10 *The decision problem* SpKey-g3-equal-g5 *is defined as* Input *Table T over schema R, $K \subseteq R$.*
Question *Is $g_3(K) = g_5(K)$?*

Corollary 1 *The decision problem SpKey-g3-equal-g5 is in P.*

Example Let $R = \{A_1, A_2, A_3\}$, $K_1 = \{A_1, A_2\}, K_2 = \{A_2, A_3\}$.

$$T = \begin{array}{c|ccc} & A_1 & A_2 & A_3 \\ \hline t_1 & 1 & \perp & 1 \\ t_2 & 1 & 2 & 2 \\ t_3 & 2 & 1 & 1 \\ t_4 & 2 & 1 & 1 \\ \hline \end{array}$$

$T \setminus \{t_4\} \models sp\langle K_1\rangle$ and $T \setminus \{t_4\} \models sp\langle K_2\rangle$, but the spWorlds are different. In particular, this implies that for $\mathcal{K} = \{K_1, K_2\}$ we have $g_3(\mathcal{K}) > \max\{g_3(K) \colon K \in \mathcal{K}\}$ On the other hand, trivially $g_3(\mathcal{K}) \ge \max\{g_3(K) \colon K \in \mathcal{K}\}$ holds. This motivates the following definition.

Definition 11 *The problem* Max-g3 *defined as* Input *Table T over schema R, $\mathcal{K} \subseteq 2^R$.*
Question *Is $g_3(\mathcal{K}) = \max\{g_3(K) \colon K \in \mathcal{K}\}$?*

Theorem 6 *Let Table T over schema R and $\mathcal{K} \subseteq 2^R$. The decision problem Max-g3 is NP-complete.*

Proof: The problem is in NP, a witness consists of a set of tuples U to be removed, an index $j \colon \frac{|U|}{|T|} = g_3(K_j)$, also an spWorld T' of $T \setminus U$ such that each K_i is a key in T'. Verifying the witness can be done in three steps.

1. $g_3(K_j) \not\le \frac{|U|-1}{|T|}$ is checked in polynomial time using Theorem 5.
2. For all $i \ne j$ check that $g_3(K_i) \le \frac{|U|}{|T|}$ using again Theorem 5.
3. Using standard database algorithms check that $\forall i \colon K_i$ is a key in T'.

On the other hand, the SPKeySystem problem can be Karp-reduced to the present question as follows. First check for each $K_i \in \mathcal{K}$ separately whether $sp\langle K_i\rangle$ holds, this can be done in polynomial time. If $\forall i \colon T \models sp\langle K_i\rangle$ then give $\mathcal{K}$ and T as input for Max-g3. It will answer Yes iff $T \models sp\langle \mathcal{K}\rangle$. However, if $\exists i \colon T \not\models sp\langle K_i\rangle$, then give the example above as input for Max-g3. Clearly both problems have No answer. $\square$
According to Theorem 4, it is NP-complete to decide whether a given SpFD holds in a table. Here we show that approximations are also hard.

Definition 12 *The* SPFD-g3 (SPFD-g5) *problems are defined as follows.*
Input *A table T over schema R, $X, Y \subseteq R$, and positive rational number q.*
Question *Is $g_3(X \to_{sp} Y) \le q$? ($g_5(X \to_{sp} Y) \le q$?)*

Theorem 7 *Both decision problems SPFD-g3 and SPFD-g5 are NP-complete.*

Proof: To show that SPFD-g3∈NP one may take a witness consisting of a subset $U \subset T$, an spWorld $T^\star$ of $T \setminus U$ such that $T^\star \models X \to Y$ and $|U|/|T| \le q$. The validity of the witness can easily be checked in polynomial time. Similarly, to show that SPFD-g5∈NP one may take a set of tuples S over R and an spWorld $T^\star$ of$T \cup S$ such that $T^\star \models X \to Y$ and $|S|/|T| \le q$.

On the other hand, if $|T| = m$ and $q < 1/m$, then both SPFD-g3 and SPFD-g5 are equivalent with the original SPFD problem, since the smallest non-zero approximation measure is obtained if one tuple is needed to be deleted or added. According to Theorem 4, SPFD problem is NP-complete, thus so are SPFD-g3 and SPFD-g5. $\square$

Acknowledgement

Research of the second author was partially supported by the National Research, Development and Innovation Office (NKFIH) grants K–116769 and SNN-135643. This work was also supported by the BME- Artificial Intelligence FIKP grant of EMMI (BME FIKP-MI/SC) and by the Ministry of Innovation and Technology and the National Research, Development and Innovation Office within the Artificial Intelligence National Laboratory of Hungary.

References

[1] M. Al-Atar and A. Sali. Approximate keys and functional dependencies in incomplete databases with limited domains. In *Foundations of Information and Knowledge Systems 12th International Symposium, FoIKS 2022 Helsinki, Finland, June 20–23, 2022 Proceedings*, volume 13388 of *LNCS*, pages 147–167. Springer Nature Switzerland AG, 2022.

[2] M. Al-Atar and A. Sali. Strongly possible functional dependencies for sql. *Acta Cybernetica*, 2022.

[3] M. Alattar and A. Sali. Keys in relational databases with nulls and bounded domains. In *European Conference on Advances in Databases and Information Systems*, pages 33–50. Springer, 2019.

[4] M. Alattar and A. Sali. Functional dependencies in incomplete databases with limited domains. In *International Symposium on Foundations of Information and Knowledge Systems*, pages 1–21. Springer, 2020.

[5] M. Alattar and A. Sali. Strongly possible keys for sql. *Journal on Data Semantics*, 9(2):85–99, 2020.

[6] L. Bertossi. Database repairs and consistent query answering: Origins and further developments. In *Proceedings of the 38th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, pages 48–58, 2019.

[7] J. Biskup and L. Wiese. A sound and complete model-generation procedure for consistent and confidentiality-preserving databases. *Theoretical Computer Science*, 412(31):4044–4072, 2011.

[8] A. Farhangfar, L. A. Kurgan, and W. Pedrycz. A novel framework for imputation of missing values in databases. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 37(5):692–709, 2007.

[9] C. Giannella and E. Robertson. On approximation measures for functional dependencies. *Information Systems*, 29(6):483–507, 2004.

[10] L. A. Goodman and W. H. Kruskal. Measures of association for cross classifications. *Measures of association for cross classifications*, pages 2–34, 1979.

[11] J. Kivinen and H. Mannila. Approximate inference of functional dependencies from relations. *Theoretical Computer Science*, 149(1):129–149, 1995.

[12] H. Köhler, U. Leck, S. Link, and X. Zhou. Possible and certain keys for sql. *The VLDB Journal*, 25(4):571–596, 2016.

[13] W. Lipski Jr. On databases with incomplete information. *Journal of the ACM (JACM)*, 28(1):41–70, 1981.

[14] L. Lovász and M. Plummer. *Matching theory*, volume 367. American Mathematical Soc., 2009.

[15] J. Wijsen. Foundations of query answering on inconsistent databases. *ACM SIGMOD Record*, 48(3):6–16, 2019.