Strongly Possible Keys in Incomplete Databases with Limited Domains

Munqath Alattar
Department of Computer Science and Information Theory
Budapest University of Technology and Economics
Budapest, Hungary
Email: m.attar@cs.bme.hu

Attila Sali
Alfréd Rényi Institute of Mathematics
Hungarian Academy of Sciences
Budapest, Hungary
Email: sali.attila@renyi.mta.hu

***Abstract*—Missing values that may occur in the key attributes of a database table is an extensive problem and handling it is an important and challenging task, as the records need to contain distinct and total values in their key part. The existing effective approaches include an imputation operation for each occurrence of a null in the key part of the data. In this paper, we assume the situation when the attributes domains are not known. For that, a new concept of keys called *strongly possible keys* in databases with null values is introduced. It lies between possible keys and certain keys introduced by Köhler et. al. earlier. The definition uses only information extractable from the database table. Furthermore, an approximation concept of the strongly possible key is provided.**

Keywords–Strongly possible keys; null values; approximation of keys.

I. INTRODUCTION

A basic approach to treat null values in keys of relational databases is an imputation operation for each occurrence of a null in the key part of the data with a value from the attribute domain as explained by [1]. We investigate the situation when the attributes' domains are not known. For that, we only consider what we have in the given data and extract the values to be imputed from the data itself for each attribute so that the resulting complete dataset after the imputation would not contain two tuples having the same value in their key. Köhler et al. [1] used possible worlds by replacing each occurrence of a null with a value from the corresponding attribute's (possibly infinite) domain. They defined a possible key as a key that is satisfied by some possible world of a non total database table and a certain key as a key that is satisfied by every possible world of the table. In many cases, we have no proper reason to assume existence of any other attribute value than the ones already existing in the table. Such examples could be types of cars, diagnoses of patients, applied medications, dates of exams, course descriptions, etc. We define a strongly possible key as a key that is satisfied by some possible world that is obtained by replacing each occurrence of null value from the corresponding attribute existing values. We call this kind of a possible world a strongly possible world. This is a data mining type approach; our idea is that we are given a raw table with nulls and we would like to identify possible key sets based on the data only.

The remainder of the paper is organized as follows. In Section 2, some definitions are stated. In Section 3, strongly possible keys, their discovery, and characterization of the implication problem of systems of strongly possible keys are provided. Approximation measures are studied in Section 4. Section 5 presents concluding remarks and future research directions.

II. 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 referred as D_i = $dom(A_i)$ for i = 1,2,... n. And if $X \subseteq R$ then $D_X = \prod_{\forall A_i \in K} D_i$. An instance T = (t_1,t_2, ... t_s) over R is a set of tuples that each tuple is a function $t : R \rightarrow \bigcup_{A_i \in R} dom(A_i)$ and $t[A_i]$ is in the $dom(A_i)$ for all A_i in R. For a tuple $t_r \in T$, let $t_r[A_i]$ be the restriction of the r^{th} tuple of T to A_i.

In practice, data models may contain an unknown information about the value of some tuple $t_j[A_i]$ for j = 0,1,... s that is denoted by $\perp$. t_1 and t_2 are *weakly similar* on $X \subseteq R$ denoted as $t_1[X] \sim_w t_2[X]$ as defined by Köhler [1] 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 $X \subseteq T$ 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 the same for strong similarity. For a null-free table, a set of attributes $K \subset R$ is a *key* if there are no two distinct tuples in the table that share the same values in all the attributes of K:

$$t_a[K] \neq t_b[K] \; \forall \; 0 \leq a, b \leq s \text{ such that } a \neq b$$

The concepts of possible and certain keys were defined by Köhler et al [1]. Let T' = (t'_1, t'_2, ... t'_s) be a table that represents a total version of T which is obtained by replacing the occurrences of $\perp$ in all attributes $t[A_i]$ with a value from the domain D_i different from $\perp$ for each i. T' is called a *possible world* of T. In a possible world T', t'_i is weakly similar to t_i and T' is completely null-free table. A *possible key* K denoted as $p\langle K\rangle$, is a key for some possible world T' of T, so that:

$$t'_1[K] \neq t'_2[K], \;\; \forall t'_1, t'_2 \in T'$$

Similarly, a *certain key* K referred as $c\langle K\rangle$, is a key for every possible world T' of T. The *visible domain* of an

 ISBN: 978-1-61208-715-3

attribute A (VD_A) is the set of all distinct values except $\perp$ that are already used by tuples in T:

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

The term visible domain refers to the data that already exist in a given dataset. For example, if we have a dataset with no information about the attributes' domains definitions, then we use the data itself to define their own structure and domains. This may provide more realistic results when extracting the relationship between data so it is more reliable to consider only what information we have in a given dataset.

A possible world T' is called *strongly possible world* if $T' \subseteq VD_1 \times VD_2 \times \ldots \times VD_n$.

A subset $K \subseteq R$ is a *strongly possible key* (in notation $sp\langle K\rangle$) in T if $\exists$ a strongly possible world $T' \subseteq VD_1 \times VD_2 \times \ldots \times VD_n$ such that K is a key in T'.

III. RESULTS

Table I implies $sp\langle AB\rangle$ as a strongly possible key because there is a strongly possible world in Table II where AB is a key. On the other hand, Table I implies neither $sp\langle AC\rangle$ nor $sp\langle BC\rangle$ because there is no strongly possible world T' that has AC or BC as keys.

TABLE I. A DATASET WITH NULLS

A	B	C	D
3	2	$\perp$	0
15	1	2	10
$\perp$	2	2	$\perp$

TABLE II. A STRONGLY POSSIBLE WORLD OF TABLE I

A	B	C	D
3	2	2	0
15	1	2	10
15	2	2	10

Let Σ be a set of strongly possible keys and θ a single strongly possible key over a relation schema R. Σ logically implies θ, denoted as $\Sigma \models \theta$ if for every instance T over R satisfying every strongly possible key in Σ we have that T satisfies θ.

Theorem 1: $\Sigma \models sp\langle K\rangle \iff \exists Y \subseteq K$ s.t. $sp\langle Y\rangle \in \Sigma$.

Proof: $\Leftarrow$: $\exists T'$ s.t. $t'_i[Y] \neq t'_j[Y], \forall i \neq j$, so $t'_i[K] \neq t'_j[K], \forall i \neq j$ holds, as well.

$\Rightarrow$: Suppose indirectly that $sp\langle Y\rangle \notin \Sigma \ \forall Y \subseteq K$. Consider the following instance consisting of two tuples $t_1 = (0, 0, \ldots, 0)$, $t_2[K] = (\perp, \perp, \ldots, \perp)$, and $t_2[R \setminus K] = (1, 1, \ldots 1)$ as in Table III . Then, the only possible t'_2 in T' is $t'_2(0, 0, \ldots, 0, 1, 1, \ldots, 1)$. Furthermore, $\forall Z$ where $sp\langle Z\rangle \in \Sigma$, there must be $z \in Z \setminus K$, thus $t'_1[Z] \neq t'_2[Z]$ but $t'_1[K] = t'_2[K]$ showing that (t_1, t_2) satisfies every strongly possible key constraints from Σ, but does not satisfy $sp\langle K\rangle$. ∎

TABLE III. INCOMPLETE DATA INSTANCE

	K	$R \setminus K$
t_1	0 0 0 0	00000000
t_2	$\perp\perp\perp\perp$	11111111

Note 1: If $\Sigma \models \neg sp\langle K\rangle$ and $Y \subseteq K$ then $\Sigma \models \neg sp\langle Y\rangle$.

Note 2: If $\Sigma \models sp\langle K\rangle$, then $\Sigma \models p\langle K\rangle$ but the reverse is not necessarily true, since $D_K \supseteq VD_K$ could be proper containment so K could be made a key by imputing values from $D_K \setminus VD_K$. For example, in Table III, it is shown that $\neg sp\langle K\rangle$ holds, but $p\langle K\rangle$ may hold in some T' if there is at least one other value in the domain of K rather than the zeros to be placed instead of the nulls in the second tuple so that $t'_1[K] \neq t'_2[K]$ results.

Note 3: If $\Sigma \models c\langle K\rangle$, then $\Sigma \models sp\langle K\rangle$. As certain keys hold in any possible world, they hold also if this possible world is created using visible domain.

Note 4: For a single attribute A, $sp\langle A\rangle \iff t[A] \nsim_w t'[A] \ \forall t, t'$ s.t. $t \neq t'$, i.e., if there are no nulls occurrences in A.

In other words, a single attribute with a null value cannot be a strongly possible key. That is because replacing an occurrence of null with a visible domain value results in duplicated values for that attribute.

Let us consider a schema $R = \{A_1, A_2, \ldots, A_n\}$ and let $\mathcal{K} = \{K_1, K_2, \ldots K_p\}$ be a collection of attribute sets and $T = \{t_1, t_2, \ldots, t_s\}$ be an instance with possible null occurrences. Our main question here is whether $\Sigma = \{sp\langle K_1\rangle, sp\langle K_2\rangle, \ldots, sp\langle K_p\rangle\}$ holds in T? Let $E_i = \{t' \in VD_1 \times VD_2 \times \ldots \times VD_n : t' \sim_w t_i\}$. Let $S \subseteq VD_1 \times VD_2 \times \ldots \times VD_n$ be the union $S = E_1 \cup E_2 \cup \ldots \cup E_s$ and define bipartite graph $G = (T, S; E)$ by $\{t, t'\} \in E \iff t \sim_w t'$ for $t \in T$ and $t' \in S$. Let $(S, \mathcal{M}_0)$ be the transversal matroid (see [2]) defined by G on S, that is a subset $X \subseteq S$ satisfies $X \in \mathcal{M}_0$ if X can be matched into T. Furthermore, consider the partitions

$$S = S_1^j \cup S_2^j \cup \ldots \cup S_{p_j}^j \tag{1}$$

induced by K_j for $j = 1, 2, \ldots, p$ such that S_i^j's are maximal sets of tuples from S that agree on K_j. Let $(S, \mathcal{M}_j)$ be the partition matroid given by (1). We can formulate the following theorem.

Theorem 2: Let T be an instance over schema $R = \{A_1, A_2, \ldots, A_n\}$ and let $\mathcal{K} = \{K_1, K_2, \ldots K_p\}$ be a collection of attribute sets. $\Sigma = \{sp\langle K_1\rangle, sp\langle K_2\rangle, \ldots, sp\langle K_p\rangle\}$ holds in T if and only if the matroids $(S, \mathcal{M}_j)$ have a common independent set of size $|T|$ for $j = 0, 1, \ldots p$

Proof: An independent set T' of size $|T|$ in matroid $(S, \mathcal{M}_0)$ means that tuples in T' form a strongly possible world for T. That they are independent in $(S, \mathcal{M}_j)$ means that K_j is a key in T', that is $sp\langle K_j\rangle$ holds.

Conversely, if $\Sigma = \{sp\langle K_1\rangle, sp\langle K_2\rangle, \ldots, sp\langle K_p\rangle\}$ holds in T, then there exists a strongly possible world $T' = \{t'_1, t'_2, \ldots, t'_s\} \subseteq VD_1 \times VD_2 \times \ldots \times VD_n$ such that $t_i \sim_w t'_i$. This means that $T' \subseteq S$ and that T' is independent in transversal matroid $(S, \mathcal{M}_0)$. $sp\langle K_j\rangle$ holds implies that tuples t'_i are pairwise distinct on K_j, that is T' is independent in partition matroid $(S, \mathcal{M}_j)$. ∎

Unfortunately, Theorem 2 does not give a good algorithm to decide the satisfaction of a system Σ of strongly possible keys, because as soon as Σ contains at least two constraints, then we would have to calculate the size of the largest common independent set of at least three matroids, known to be an NP-complete problem [3].

 ISBN: 978-1-61208-715-3

In case of a single strongly possible key $sp\langle K\rangle$ constraint, Theorem 2 requires to compute the largest common independent set of two matroids, which can be solved in polynomial time [4]. However, we can reduce the problem to the somewhat simpler problem of matchings in bipartite graphs.

If we want to decide whether $sp\langle K\rangle$ holds or not, we can forget about the attributes that are not in K since we need distinct values on K as a matching from $VD_{A_1}\times VD_{A_2}\times\ldots\times VD_{A_b}$ to $T=\{t_1,t_2\ldots t_r\}|_K$ where $K=\{A_1,A_2\ldots A_b\}$. Thus, we may construct a table T' that is formed by finding all the possible combinations of the visible domains of $T|_K$ that are weakly similar to some tuple in $T|_K$.

$$T'=\{t':\exists t\in T: t'[K]\sim_w t[K]\}\subseteq VD_1\times VD_2\times\ldots\times VD_b$$

Finding the matching between T and T' that covers all the tuples in T (if it exists) will result in the set of tuples in T' that needs to be replaced in T so that K is a strongly possible key.

Let $c_v(A)$ denote the number of tuples that have value v in attribute A, that is $c_v(A)=|\{t\in T\colon t[A]=v\}|$. Next are some necessary conditions to have a strongly possible key.

Proposition 1: Let $K\subseteq R$ be a set of attributes. If $sp\langle K\rangle$ holds, then

1) No two tuples t_i,t_j are strongly similar in K.
2) $|T|\leq\prod_{\forall A\in K}|VD_A|$.
3) $\forall B\in K$, number of nulls in $B\leq\sum_{\forall v\in VD_B}\Big(\frac{\prod_{\forall A\in K}|VD_A|}{|VD_B|}-c_v(B)\Big)$.
4) For all $v\in VD_B$ we have $c_v(B)\leq\frac{\prod_{\forall A\in K}|VD_A|}{|VD_B|}$

Proof: The first condition is obviously required so that K is a strongly possible key, where the strong similarity means that the two tuples are total and equal to each other in the key part and this violates the general key definition. In addition to that, for any set of attributes, the maximum number of distinct combination of their values is the size of the multiplication of their visible domain, and this proves (2). Moreover, to prove conditions (3) and (4), when K is $sp\langle K\rangle$ in T then there should exist a T' with no two tuples having the same values in all K attributes after filling all their nulls. So for each set of tuples S that has the same value v in the attribute B, the number of distinct combinations of the other attributes is the multiplication of their VD's, means the number of tuples in S should not be more than $\prod_{\forall A\in(K\setminus B)}VD_A$. Thus, the number of times value v can be used to replace a null in attribute B is at most $\frac{\prod_{\forall A\in K}|VD_A|}{VD_B}-c_v(B)$. ■

Note that $sp\langle K\rangle$ holds if a matching covering T exists in the bipartite graph $G=(T,T';E)$ defined as above, $\{t,t'\}\in E\iff t[K]\sim_w t[K]'$. We can apply Hall's Theorem to obtain

$$\forall X\subseteq T,\ \text{we have } |N(X)|\geq|X|$$

$$\text{for } N(X)=\{t'\colon\exists t\in X \text{ such that } t[K]'\sim_w t[K]\}$$

IV. Strongly Possible Keys Approximation

To measure the degree of how much a strongly possible key holds in a given dataset, we use the g_3 measure introduced in [5]. g_3 is based on the idea that the degree to which ASP key is approximate is determined by the minimum number of tuples that need be removed from T so that K becomes an ASP key. To find the tuples that we need to remove, we suggest to construct the maximum matching in graph $G=(T,T';E)$.

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

where $\nu(G)$ denotes the maximum size of a matching in graph G.

Let $\mathcal{M}$ be the collection of connected components in graph G that hold the strongly possible key condition, i.e., there is a matching cover all T tuples in that set ($\forall_{M\in\mathcal{M}}\ \nexists X\subseteq M\cap T$ such that $|X|>N(X)$). Let $C\subseteq G$ be defined as $C=G\setminus\bigcup_{\forall M\in\mathcal{M}}M$ and let $\mathcal{M}'$ be the set of connected components of C. In addition to that, we use the term V_M to denote the set of vertices of T in a component M. So, the maximum matching can be written as $\sum_{M\in\mathcal{M}}(|V_M|)+\sum_{\forall M'\in\mathcal{M}'}\nu(M')$. Therefore we can reformulate the g_3 measure as:

$$g_3(K)=\frac{|T|-(\sum_{M\in\mathcal{M}}(|V_M|)+\sum_{M'\in\mathcal{M}'}\nu(M'))}{|T|}$$

Figure 2 shows 7 tables that represent the key part only of the data where each table has more than one attribute. Tables A, B and C have $2n$ tuples, tables E and F have n tuples, and table D has $n+l$ tuples while table G has kn tuples. Table D includes a variable $0\leq\beta\leq\frac{n}{2}$. We intend to use these cases to illustrate the differences and give a bound of g_3/g_3^c where it is always true that $g_3-g_3^c\geq 0$. The graphs show the weak similarity relationship between the data tuples and the visible domains combinations. The visible domains combinations are shown on Figure 1. For example, in table A, the first two tuples of T in the left side of the graph can have a unique weakly similar tuples in T' for each, while for the rest, every two tuples in T form a connected component that have only one weakly similar tuple in T'. On other hand, all the tuples of table E form connected component of size n that have a weakly similar relation (matching) to one tuple in T'.

Measuring the strongly possible keys approximation can be more appropriate by take into consideration the effect of each connected component in the graph on the matching. More specifically, $\mathcal{M}$ represents the sets of tuples that do not require any tuple to be removed to get a strongly possible key, while the components of $\mathcal{M}'$ represent the sets of tuples that contain some tuples which need to be removed to have a strongly possible key. We consider the components of $\mathcal{M}$ to get their effect doubled in the approximation measure because they represent a part of the data that is not affected by any tuples removal. So, we propose a derived version of g_3 measure named g_3^c that considers the effects of these components.

$$g_3^c(K)=\frac{|T|-(\sum_{M\in\mathcal{M}}(|V_M|)+\sum_{M'\in\mathcal{M}'}\nu(M'))}{|T|+\sum_{M\in\mathcal{M}}|V_M|}$$

Theorem 3: For any table T and set of attributes K we have either $g_3(K)=g_3^c(K)$ or $1<g_3(K)/g_3^c(K)<2$. Furthermore, for any rational number $1\leq\frac{p}{q}<2$ there exist tables of arbitrarily large number of tuples with $g_3(K)/g_3^c(K)=\frac{p}{q}$.

Proof: $g_3(K)$ and $g_3^c(K)$ are different only in the denominator part. The number of tuples of the components in $\mathcal{M}$ can't be more than the total number of tuples in the table, so $0\leq\sum_{M\in\mathcal{M}}|V_M|\leq|T|$ and $\sum_{M\in\mathcal{M}}|V_M|=|T|$ iff every

(A)	$t'_i = (i-1,0)$ $i = 1,2,\ldots,n+1$
(B)	$t'_i = (i-1,0)$ $i = 1,2,\ldots,n+1$
(C)	$t'_i = (i-1,0)$ $i = 1,2,\ldots,n+1$
(D)	$t'_i = (i,0,0)$ for $i = 1,2,\ldots,n-\beta,$ and $t'_{n-\beta+j} = (0,0,j-1)$ for $j = 1,2,\ldots,l+1$
(E)	$t'_1 = (0,0)$
(F)	$t'_i = (i,0)$ $i = 1,2,\ldots,n-1$
(G)	$t'_{jn+i} = (jn+i+j,0)$ for $i = 1,2,\ldots,n-j-1$ and $j = 0,1,\ldots,k-1$

Figure 1. Visible Domains Combinations of Tables of Figure 2

tuple is a member of some connected component in $\mathcal{M}$. In the latter case $g_3(K) = g_3^c(K)$, otherwise the denominator of $g_3^c(K)$ is less than twice the denominator of $g_3(K)$ that proves the inequlaties of the ratio. Table E proves that $g_3(K) = g_3^c(K)$ can hold for arbitrarily large tables. Now let $1 < \frac{p}{q} < 2$ be given with $\frac{p}{q} = 1 + \frac{p'}{q'}$. Consider Table D where

$$g_3(K)/g_3^c(K) = \left(\frac{\beta-1}{n+l}\right) / \left(\frac{\beta-1}{n+2l}\right)$$

which can simply be written as $1 + \frac{l}{n+l}$. Now taking $n = \alpha(q'-p')$, $l = \alpha p'$ and any β between 2 and $\lfloor \frac{n}{2} \rfloor$ we obtain that

$$g_3(K)/g_3^c(K) = 1 + \frac{p'}{q'}.$$

■

Note that $g_3(K)$ ranges between $1/n$ and $1/2$ depending on the choice of β.

V. Conclusion and Future Directions

The main contributions of this paper are as follows:

- We introduced and defined strongly possible keys over database relations that contain some occurrences of nulls.
- We provided some properties, observations, and number of necessary conditions so that a strongly possible key holds in a given dataset. We show that deciding whether a given set of attributes is a strongly possible key can be done by application of matchings in bipartite graph, so Hall's condition is naturally applied.
- We showed that deciding whether a given system of sets of attributes is a system of possible keys for a given table can be done using matroid intersection. However, we need at least three matroids, and matroid intersection of three or more matroids is NP-complete, which suggests that our problem is also NP-complete.
- We studied systems of strongly possible keys and we gave characterization of the implication problem.
- An approximation concept of the strongly possible key was introduced to measure how close approximation of a strongly possible key holds in a data relation, using g_3 measure. We derived the measure g_3^c from g_3 and gave bounds of the two measures.

Strongly possible keys are special cases of possible keys of relational schemata with each attribute having finite domain. So, future research is needed to decide what properties of implication, axiomatization of inference remain valid in this setting. Note that the main results in [1] consider that at least one attribute has infinite domain.

We plan to extend our research from keys to functional dependencies. Weak and strong functional dependencies were introduced in [6]. A wFD $X \to_w Y$ holds if there is a possible world T' that satisfies FD $X \to Y$, while sFD $X \to_s Y$ holds if every possible world satisfies FD $X \to Y$. Our strongly possible world concept naturally induces an intermediate concept of functional dependency. Future research on possible keys of finite domains might extend our results on strongly possible keys.

Finally, Theorem 2 defines a matroid intersection problem. It would be interesting to know whether this particular question is NP-complete, which we strongly believe it is.

References

[1] H. Köhler, U. Leck, S. Link, and X. Zhou, "Possible and certain keys for sql," The VLDB Journal, vol. 25, 2016, pp. 571–596.

[2] D. Welsh, Matroid Theory. Academic Press, New York, 1976.

[3] M. Garey and D. Johnson, Computers and Intractability. A Guide to the Theory of NP-Completeness. Freeman, New York, 1979.

[4] E. Lawler, "Matroid intersection algorithms," Mathematical Programming, vol. 9, 1975, pp. 31–56.

[5] J. Kivinen and H. Mannila, "Approximate inference of functional dependencies from relations," Theoretical Computer Science, vol. 149, 1995, pp. 129–149.

[6] G. L. Mark Levene, "Axiomatisation of functional dependencies in incomplete relations," Theoretical Computer Science, vol. 206, 1998.

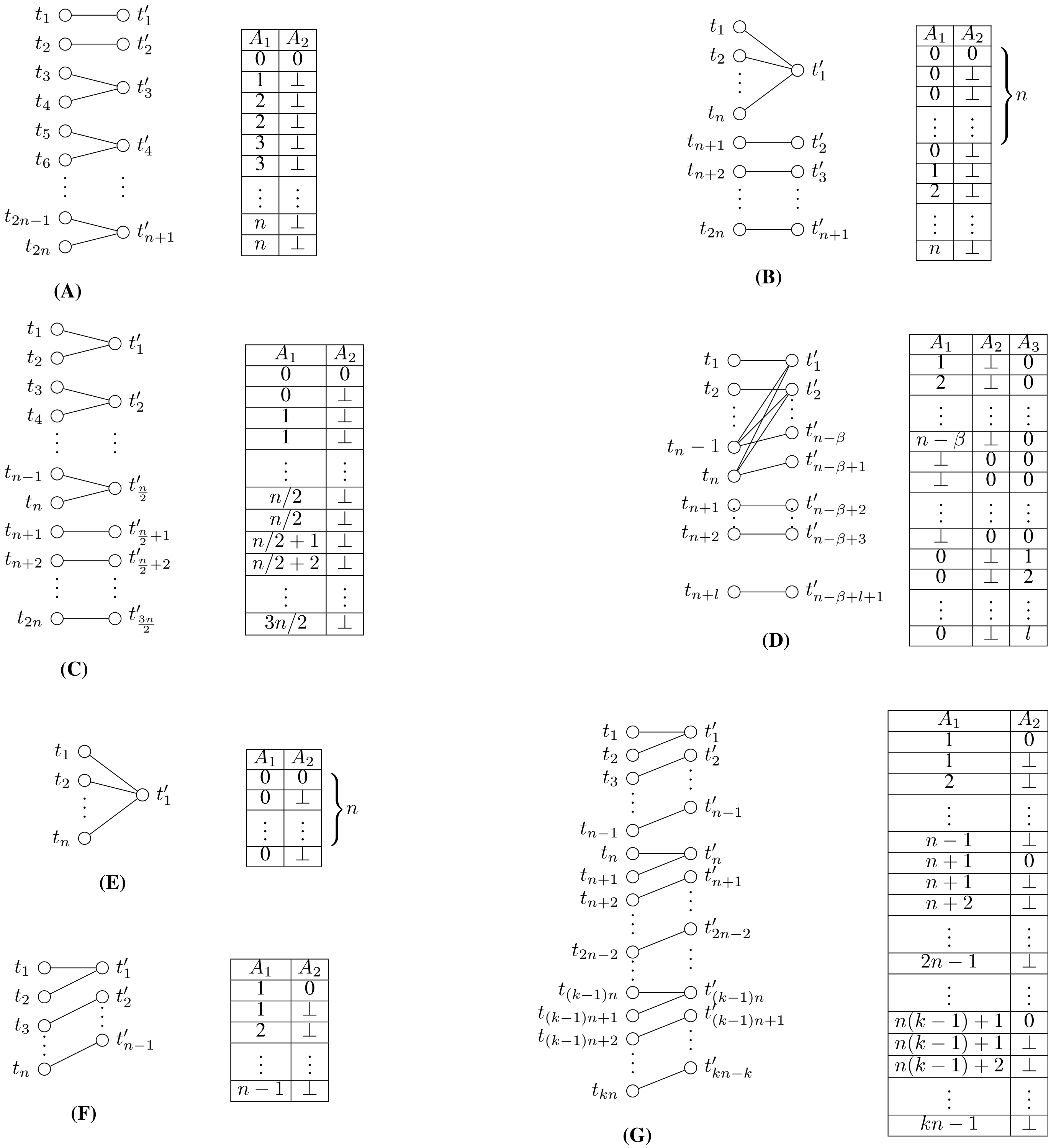

Figure 2. Sample Tables for Comparison Results

TABLE IV. MAIN COMPARISON RESULTS

	A	B	C	D	E	F	G
g_3	$\frac{n-1}{2n}$	$\frac{n-1}{2n}$	$\frac{1}{4}$	$\frac{\beta-1}{n+l}$	$\frac{n-1}{n}$	$\frac{1}{n}$	$\frac{1}{n}$
g_3^c	$\frac{n-1}{2n+2}$	$\frac{n-1}{3n}$	$\frac{1}{6}$	$\frac{\beta-1}{n+2l}$	$\frac{n-1}{n}$	$\frac{1}{2n-2}$	$\frac{1}{2n-2}$