Noname manuscript No.
(will be inserted by the editor)

Disjunctive Databases for Representing Repairs

Cristian Molinaro · Jan Chomicki · Jerzy Marcinkowski

Received: date / Accepted: date

Abstract This paper addresses the problem of representing the set of repairs of a possibly inconsistent database by means of a disjunctive database. Specifically, the class of denial constraints is considered. We show that, given a database and a set of denial constraints, there exists a (unique) disjunctive database, called *canonical*, which represents the repairs of the database w.r.t. the constraints and is contained in any other disjunctive database with the same set of minimal models. We propose an algorithm for computing the canonical disjunctive database. Finally, we study the size of the canonical disjunctive database in the presence of functional dependencies for both repairs and cardinality-based repairs.

Keywords Inconsistent databases · Incomplete databases · Repairs · Disjunctive databases

1 Introduction

The problem of managing inconsistent data nowadays arises in several scenarios. How to extract reliable information from *inconsistent databases*, i.e. databases violating integrity constraints, has been extensively studied in the past several years. Most of the works in the literature rely on the notions of *repair* and *consistent query answer* [2]. Intuitively, a repair for a database w.r.t. a set of integrity constraints is a consistent database which "minimally" differs from the (possibly inconsistent) original database. The consistent answers to a query over an inconsistent database are those tuples which

Cristian Molinaro
DEIS, Universitá della Calabria, 87036 Rende, Italy
E-mail: cmolinaro@deis.unical.it

Jan Chomicki
Department of Computer Science and Engineering, 201 Bell Hall, The State University of New York at Buffalo, Buffalo, NY 14260, USA
E-mail: chomicki@cse.buffalo.edu

Jerzy Marcinkowski
Institute of Informatics, Wroclaw University, Przesmyckiego 20, 51-151 Wroclaw, Poland
E-mail: jma@cs.uni.wroc.pl

can be obtained by evaluating the query in every repair of the database. Let us illustrate the notions of repair and consistent query answer by means of an example.

Example 1 Consider the following relation r

employee

$Name$	$Salary$	$Dept$
$john$	50	cs
$john$	100	cs

and the functional dependency $f: \ Name \rightarrow Salary\ Dept$ stating that each employee has a unique salary and a unique department. Clearly, r is inconsistent w.r.t. f as it stores two different salaries for the same employee *john*. Assuming that the database is viewed as a set of facts and the symmetric difference is used to capture the distance between two databases, there exist two repairs for r w.r.t. f, namely $\{employee(john, 50, cs)\}$ and $\{employee(john, 100, cs)\}$. The consistent answer to the query asking for the department of *john* is *cs* (as this is the answer of the query in both repairs), whereas the query asking for the salary of *john* has no consistent answer (as the two repairs do not agree on the answer).

An introduction to the central concepts of consistent query answering is [8], whereas surveys on this topic are [6,5].

Inconsistency leads to *uncertainty* as to the actual values of tuple attributes. Thus, it is natural to study the possible use of incomplete database frameworks in this context. The set of repairs for a possibly inconsistent database could be represented by means of an incomplete database whose possible worlds are exactly the repairs of the inconsistent database.

In this paper, we consider a specific incomplete database framework: *disjunctive databases*. A disjunctive database is a finite set of disjunctions of facts. Its semantics is given by the set of minimal models. There is a clear intuitive connection between inconsistent and disjunctive databases. For instance, the repairs of the relation r of Example 1 could be represented by the disjunctive database $\mathcal{D} = \{employee(john, 50, cs) \vee employee(john, 100, cs)\}$, as the minimal models of $\mathcal{D}$ are exactly the repairs of r w.r.t. f. Disjunctive databases have been studied for a long time [12,13,15,10]. More recently, they have again attracted attention in the database research community because of potential applications in data integration, extraction and cleaning [4]. Our approach should be distinguished from the approaches that rely on stable model semantics of *disjunctive logic programs with negation* to represent repairs of inconsistent databases [3,7,11].

In this paper we address the problem of *representing* the set of repairs of a database w.r.t. a set of denial constraints by means of a disjunctive database (in other words, a disjunctive database whose minimal models are the repairs).

We show that, given a database and a set of denial constraints, there exists a unique, *canonical* disjunctive database which (a) represents the repairs of the database w.r.t. the constraints, and (b) is contained in any other disjunctive database having the same set of minimal models. We propose an algorithm for computing the canonical disjunctive database which in general can be of exponential size. Next, we study the size of the canonical disjunctive database in the presence of restricted functional dependencies. We show that the canonical disjunctive database is of linear size when only one key in considered, but it may be of exponential size in the presence of two keys or one non-key functional dependency. Finally, we demonstrate that these results hold also for a different, cardinality-based semantics of repairs [14].

The paper is organized as follows. In Section 2, we introduce some basic notions in inconsistent and disjunctive databases. In Section 3, we present an algorithm to compute the canonical disjunctive database and show that this database is contained in any other disjunctive database with the same minimal models. In Section 4, we study the size of the canonical disjunctive databases in the presence of functional dependencies. In Section 5, we investigate the size of the canonical disjunctive databases under the cardinality-based semantics of repairs. Finally, in Section 6 we draw the conclusions and outline some possible future research topics.

2 Preliminaries

In this section we introduce some basic notions of relational, inconsistent, and disjunctive databases.

2.1 Relational databases

We assume the standard concepts of the relational data model. A database is a collection of relations. Each relation is a finite set of tuples and has a finite set of attributes. The values of each attribute are integers, rationals or uninterpreted constants. Each tuple $\bar{t}$ in a relation p can be viewed as a fact $p(\bar{t})$; then a database can be viewed as a finite set of facts.

We say that a database is *consistent* w.r.t. a set of integrity constraints if it satisfies the integrity constraints, otherwise it is *inconsistent*. In this paper we consider the class of *denial constraints*. A denial constraint is a first-order logic sentence of the following form:

$$\forall \overline{X}_1 \ldots \overline{X}_n \ \neg[p_1(\overline{X}_1) \wedge \ldots \wedge p_n(\overline{X}_n) \wedge \varphi(\overline{X}_1, \ldots, \overline{X}_n)]$$

where the $\overline{X}_i$'s are sequences of variables, the p_i's are relational symbols and φ is a conjunction of atoms referring to built-in, arithmetic or comparison, predicates. Special cases of denial constraints are functional dependencies and key constraints. A functional dependency is of the form

$$\forall \overline{X}_1 \overline{X}_2 \overline{X}_3 \overline{X}_4 \overline{X}_5 \ \neg[p(\overline{X}_1, \overline{X}_2, \overline{X}_4) \wedge p(\overline{X}_1, \overline{X}_3, \overline{X}_5) \wedge \overline{X}_2 \neq \overline{X}_3]$$

The previous functional dependency can be also stated as $X \rightarrow Y$, where X is the set of attributes of p corresponding to $\overline{X}_1$ whereas Y is the set of attributes of p corresponding to $\overline{X}_2$ (and $\overline{X}_3$). A key constraint is of the form

$$\forall \overline{X}_1 \overline{X}_2 \overline{X}_3 \ \neg[p(\overline{X}_1, \overline{X}_2) \wedge p(\overline{X}_1, \overline{X}_3) \wedge \overline{X}_2 \neq \overline{X}_3]$$

We say that the set of attributes corresponding to $\overline{X}_1$ is a key. We assume that the given set of integrity constraints is satisfiable.

2.2 Inconsistent databases

As it has been already said in the introduction, a repair of a database w.r.t. a set of integrity constraints is a consistent database which "minimally" differs from the (possibly inconsistent) original database [2]. The symmetric difference is used to capture the distance between two databases. Because we consider denial constraints and assume that the symmetric difference has to be minimal under set inclusion, repairs are *maximal consistent subsets* of the original database (although in Section 5 we will consider cardinality-based repairs, where the *cardinality* of the symmetric difference is minimized). The set of repairs of a database D w.r.t. a set F of denial constraints is denoted by $repairs(D, F)$.

Given a database D and a set F of denial constraints, the *conflict hypergraph* [9] for D and F, denoted by $\mathcal{G}_{D,F}$, is a hypergraph whose set of vertices is the set of facts of D, whereas the set of edges consists of all the sets $\{p_1(\overline{c}_1), \dots, p_n(\overline{c}_n)\}$ s.t. $p_1(\overline{c}_1), \dots, p_n(\overline{c}_n)$ are facts of D which violate together a denial constraint in F, i.e. there exist a denial constraint

$$\forall \overline{X}_1 \dots \overline{X}_n \ \neg[p_1(\overline{X}_1) \wedge \dots \wedge p_n(\overline{X}_n) \wedge \varphi(\overline{X}_1, \dots, \overline{X}_n)]$$

in F and a substitution ρ s.t. $\rho(\overline{X}_i) = \overline{c}_i$ for $i = 1..n$ and $\varphi(\overline{c}_1, \dots, \overline{c}_n)$ is true. A fact t of D is said to be *conflicting* (w.r.t. F) if it is involved in some constraint violations, that is there exists an edge $\{t, t_1, \dots, t_m\}$ ($m \geq 0$) in $\mathcal{G}_{D,F}$. For a fact t of D, we denote by $edges_{D,F}(t)$ the set of edges of $\mathcal{G}_{D,F}$ containing t, i.e. $edges_{D,F}(t) = \{e = \{t, t_1, \dots, t_k\} \mid e \in E\}$.

2.3 Disjunctive databases

A disjunctive database $\mathcal{D}$ is a finite set of non-empty disjunctions of distinct facts. A disjunction containing exactly one fact is called a *singleton* disjunction. A set M of facts is a model of $\mathcal{D}$ if $M \models \mathcal{D}$; M is minimal if there is no $M' \subset M$ s.t. $M' \models \mathcal{D}$. We denote by $\mathcal{MM}(\mathcal{D})$ the set of minimal models of $\mathcal{D}$. For a disjunction $d \in \mathcal{D}$, S_d denotes the set of facts appearing in d. Given two distinct disjunctions d_1 and d_2 in $\mathcal{D}$, we say that d_1 *subsumes* d_2 if the set of facts appearing in d_1 is a (proper) subset of the set of facts appearing in d_2, i.e. $S_{d_1} \subset S_{d_2}$. Moreover, the reduction of $\mathcal{D}$, denoted by $reduction(\mathcal{D})$, is the disjunctive database obtained from $\mathcal{D}$ by discarding all the subsumed disjunctions, that is

$$reduction(\mathcal{D}) = \{d \mid d \in \mathcal{D} \ \wedge \nexists d' \in \mathcal{D} \text{ s.t. } d' \text{ subsumes } d\}.$$

Observe that for any disjunctive database $\mathcal{D}$, $\mathcal{MM}(\mathcal{D}) = \mathcal{MM}(reduction(\mathcal{D}))$.

2.4 Computational complexity

We adopt here the *data complexity* assumption [16], under which the complexity is a function of the number of facts in the database. The set of integrity constraints is considered fixed. In this setting, the conflict hypergraph is of polynomial size and can be computed in polynomial time. We study the size of a disjunctive database representing the set of repairs of a relational database D w.r.t. a set of integrity constraints F as a function of the number of facts in D.

3 Disjunctive databases for representing repairs

In this section we propose an algorithm to compute a disjunctive database whose minimal models are the repairs of a given database w.r.t. a set of denial constraints. We show that the so computed disjunctive database is the canonical one, that is any other disjunctive database whose minimal models coincide with the repairs of the original database is a superset of the canonical one (containing, in addition, only disjunctions which are subsumed by disjunctions in the canonical disjunctive database).

Algorithm 1
Input: a database D and a set F of denial constraints
Output: a disjunctive database whose minimal models are the repairs for D and F

```
 1 : $\widehat{\mathcal{D}} := \emptyset$
 2 : $D' := D - \{t \mid \{t\}$ is an edge of $\mathcal{G}_{D,F}\}$
 3 : for each $t \in D'$
 4 :      Let $edges_{D',F}(t) = \{e_1, \ldots, e_n\}$
 5 :      $\widehat{\mathcal{D}} := \widehat{\mathcal{D}} \cup \{t \vee t_1 \vee \ldots \vee t_n \mid t_i \in e_i$ and $t_i \neq t$ for $i = 1..n\}$
 6 : repeat until $\widehat{\mathcal{D}}$ does not change
 7 :    for each edge $e = \{t_1, \ldots, t_k\}$ in $\mathcal{G}_{D',F}$
 8 :        for each $t_1 \vee D_1, \ldots, t_k \vee D_k \in \widehat{\mathcal{D}}$ s.t. $D_i$ is not an empty disjunction and
           $D_i$ does not contain any fact $t' \neq t_i$ in $e$, $i = 1..\,k$
 9 :            $\widehat{\mathcal{D}} := \widehat{\mathcal{D}} \cup \{D_1 \vee \ldots \vee D_k\}$
10 : return $reduction(\widehat{\mathcal{D}})$
```

We denote by $\mathcal{D}(D, F)$ the disjunctive database returned by Algorithm 1 with the input consisting of a database D and a set F of denial constraints. In the second step of the algorithm, every fact t s.t. $\{t\}$ is an edge of the conflict hypergraph is discarded.

The disjunctions introduced in the step 5 allow us to guarantee that the minimal models are maximal (consistent) subsets of D. Intuitively, a disjunction of the form $t \vee t_1 \vee \ldots \vee t_n$ (which contains one fact from each edge containing t) prevents from having a model m of $\widehat{\mathcal{D}}$ which contains neither t nor the t_i's as in this case m would not be maximal.

The disjunctions introduced in the step 9 allow us to guarantee that the minimal models of $\mathcal{D}(D, F)$ are consistent w.r.t. F. Specifically, the loop in lines 6–9 is performed until $\widehat{\mathcal{D}}$ satisfies the following property: for every edge $e = \{t_1, \ldots, t_k\}$ of the conflict hypergraph ($k > 1$), if there are $t_1 \vee D_1, \ldots, t_k \vee D_k \in \widehat{\mathcal{D}}$ s.t. each D_i is not an empty disjunction, then $\{D_1 \vee \ldots \vee D_k\}$ is also in $\widehat{\mathcal{D}}$. As it is shown in the proof of Theorem 1, this property entails that every minimal model of $\widehat{\mathcal{D}}$ does not contain $\{t_1, \ldots, t_k\}$. Observe that the loop ends when $\widehat{\mathcal{D}}$ does not change anymore; at each iteration new disjunctions are added to $\widehat{\mathcal{D}}$. Since the number of disjunctions is bounded (if the original database has h facts, there cannot be more than $2^h - 1$ disjunctions) the algorithm always terminates. In the last step of the algorithm, subsumed disjunctions are deleted. The following theorem states the correctness of Algorithm 1.

Theorem 1 *Given a database D and a set F of denial constraints, the set of minimal models of $\mathcal{D}(D, F)$ coincides with the set of repairs of D w.r.t. F.*

Proof. Since the the disjunctive database $\mathcal{D}(D,F)$ returned by Algorithm 1 is equal to $reduction(\widehat{\mathcal{D}})$ (step 10), then $\mathcal{MM}(\mathcal{D}(D,F)) = \mathcal{MM}(\widehat{\mathcal{D}})$. First we prove (1) $repairs(D,F) \subseteq \mathcal{MM}(\widehat{\mathcal{D}})$ and next (2) $repairs(D,F) \supseteq \mathcal{MM}(\widehat{\mathcal{D}})$.

(1) Consider a repair r in $repairs(D,F)$. First we show that (a) r is a model of $\widehat{\mathcal{D}}$ and next (b) that it is a minimal model.
(a) We prove that r satisfies each disjunction in $\widehat{\mathcal{D}}$ by induction. Specifically, as base case we consider the disjunctions introduced in the step 5 of the algorithm, whereas the inductive step refers to the disjunctions introduced in the step 9. Suppose by contradiction that r does not satisfy a disjunction $t \vee t_1 \vee \ldots \vee t_n$ introduced in the step 5. Observe that $edges_{D',F}(t) \subseteq edges_{D,F}(t)$ and each edge e' in $edges_{D,F}(t) - edges_{D',F}(t)$ is s.t. there is a fact $t' \in e'$ s.t. $\{t'\}$ is an edge of $\mathcal{G}_{D,F}$ (clearly, $t' \notin r$). Since in each edge in $edges_{D,F}(t)$ there is a fact (different from t) which is not in r, then $r \cup \{t\}$ is consistent, which violates the maximality of r. The inductive step consists in showing that r satisfies any disjunction added to $\widehat{\mathcal{D}}$ in the step 9 assuming that r satisfies $\widehat{\mathcal{D}}$. A disjunction $D_1 \vee \ldots \vee D_k$, where the D_i's are not empty disjunctions, is added to $\widehat{\mathcal{D}}$ whenever there exist $t_1 \vee D_1, \ldots, t_k \vee D_k$ in $\widehat{\mathcal{D}}$ s.t. $e = \{t_1, \ldots, t_k\}$ is an edge of $\mathcal{G}_{D',F}$, and D_i does not contain any fact $t' \neq t_i$ in e, for $i = 1..k$. Since r satisfies all the disjunctions $t_1 \vee D_1, \ldots, t_k \vee D_k$ and does not contain some fact t_j in e (as e is an edge of $\mathcal{G}_{D,F}$ too), it satisfies the disjunction D_j and then $D_1 \vee \ldots \vee D_k$ as well. Hence r is a model of $\widehat{\mathcal{D}}$.
(b) We now show that r is a minimal model, reasoning by contradiction. Assume that there exists a model $m' \subset r$ and let t be a fact in r but not in m'. Observe that t is a conflicting fact (it cannot be the case that there is a model of $\widehat{\mathcal{D}}$ which does not contain a non-conflicting fact because the algorithm introduces, in the step 5, a singleton disjunction d for each non-conflicting fact d). Moreover, as r is a repair, t is s.t. $\{t\}$ is not an edge of $\mathcal{G}_{D,F}$ and then t is in D'. For each edge e_i in $edges_{D',F}(t) = \{e_1, \ldots, e_n\}$ there is a fact $t_i \neq t$ which is not in r as it is consistent and $edges_{D',F}(t) \subseteq edges_{D,F}(t)$. The same holds for m' as it is a subset of r. Then, the disjunction $t \vee t_1 \vee \ldots \vee t_n$ in $\widehat{\mathcal{D}}$ (added in the step 5) is not satisfied by m', which contradicts that m' is a model. Hence r is a minimal model of $\widehat{\mathcal{D}}$.

(2) Consider a minimal model m in $\mathcal{MM}(\widehat{\mathcal{D}})$. We show first (a) that it is consistent w.r.t. F and then (b) that it is maximal.
(a) First of all, it is worth noting that $\widehat{\mathcal{D}}$ doesn't contain a singleton disjunction t s.t. t is a conflicting fact of D. This can be shown as follows. Two cases may occur: either $\{t\}$ is an edge of $\mathcal{G}_{D,F}$ or it is not. As for the first case, since we have proved above that each repair of D and F is a model of $\widehat{\mathcal{D}}$ and no repair contains t, it cannot be the case that t is a singleton disjunction of $\widehat{\mathcal{D}}$. Let us consider the second case. For any conflicting fact t in D s.t. $\{t\}$ is not an edge of $\mathcal{G}_{D,F}$, there exist a repair r_1 s.t. $t \in r_1$ and a repair r_2 s.t. $t \notin r_2$. As we have proved above, there are two minimal models of $\widehat{\mathcal{D}}$ corresponding to r_1 and r_2, then it cannot be the case that $t \in \widehat{\mathcal{D}}$. We prove that m is consistent w.r.t. F by contradiction, assuming that m contains a set of facts $t_1, \ldots, t_k$ s.t. $e = \{t_1, \ldots, t_k\}$ is in $\mathcal{G}_{D,F}$. Let $S_{t_i} = \{D \mid t_i \vee D \in \widehat{\mathcal{D}}$ and $D \neq \emptyset$ does not contain any fact $t' \neq t_i$ in $e\}$ for $i = 1..k$. Two cases may occur: either (a) there is a set S_{t_i} which is empty or (b) all the sets S_{t_i} are not empty. (a) Let t_j be a fact in e s.t. S_{t_j} is empty. It is easy to see that $m - \{t_j\}$ is a model, which contradicts the minimality of m. (b) For each $D_1 \in S_{t_1}, \ldots, D_k \in S_{t_k}$, it holds that $D_1 \vee \ldots \vee D_k \in \widehat{\mathcal{D}}$. Then there is a set S_{t_j} s.t. m satisfies each D in S_{t_j}, otherwise it would be the case that some $D_1 \vee \ldots \vee D_k$ in

$\widehat{\mathcal{D}}$, where D_i is in S_{t_i} for $i = 1..k$, is not satisfied. It is easy to see that $m - \{t_j\}$ is a model, which contradicts the minimality of m. Hence m is consistent w.r.t. F.
(b) Now we prove that m is a maximal (consistent) subset of D reasoning by contradiction, thus assuming that there exists $m' \supset m$ which is consistent. Let t be a fact in m' but not in m. Since m' is consistent, for each edge e_i in $edges_{D',F}(t) = \{e_1, \ldots, e_n\}$ there is a fact $t_i \neq t$ which is not in m'. The same holds for m as it is a (proper) subset of m'. This implies that m doesn't satisfy the disjunction $t \vee t_1 \vee \ldots \vee t_n$ in $\widehat{\mathcal{D}}$ (added in the step 5), thus contradicting the fact the m is a model. Hence m is a maximal consistent subset of D, that is a repair. □

Given a database D with n facts, a rough bound on the size of $\mathcal{D}(D, F)$ is that it cannot have more than $2^n - 1$ disjunctions and each disjunction contains at most n facts, for any set F of denial constraints (in the next section we will study more precisely the size of $\mathcal{D}(D, F)$ for special classes of denial constraints, namely functional dependencies and key constraints).

The following theorem allows us to identify all the disjunctive databases which have the same minimal models of a given disjunctive database. Specifically, it states that given a disjunctive database $\mathcal{D}$, any other disjunctive database with the same minimal models is a superset of *reduction*($\mathcal{D}$) containing in addition only disjunctions subsumed by disjunctions in *reduction*($\mathcal{D}$). This result allows us to state that there is a (unique) disjunctive database representing the repairs for a given database and a set of denial constraints which is contained in any other disjunctive database with the same set of minimal models. We call such a disjunctive database *canonical*. Algorithm 1 computes the canonical disjunctive database (see Corollary 1).

Theorem 2 *Given a disjunctive database $\mathcal{D}$, the set $\mathcal{R}$ of all disjunctive databases having the same minimal models as $\mathcal{D}$ is equal to:*

$$\begin{aligned}\mathcal{R} = \{\mathcal{D}' \mid\ & reduction(\mathcal{D}) \subseteq \mathcal{D}' \wedge \\ & \forall d' \in \mathcal{D}' - reduction(\mathcal{D})\ \exists d \in reduction(\mathcal{D})\ \textit{which subsumes } d'\}\end{aligned}$$

Proof. We denote by $\mathcal{S}(\mathcal{D})$ the set of all the disjunctive databases whose minimal models are $\mathcal{MM}(\mathcal{D})$. In order to prove that $\mathcal{R} = \mathcal{S}(\mathcal{D})$, first we show that (1) each disjunctive database in $\mathcal{R}$ is also in $\mathcal{S}(\mathcal{D})$ and next that (2) each disjunctive database in $\mathcal{S}(\mathcal{D})$ is in $\mathcal{R}$ too.
(1) Consider a disjunctive database $\mathcal{D}'$ in $\mathcal{R}$. It is easy to see that $reduction(\mathcal{D}') = reduction(\mathcal{D})$. As a disjunctive database and its reduction have the same minimal models, $\mathcal{MM}(\mathcal{D}') = \mathcal{MM}(\mathcal{D})$ and hence $\mathcal{D}'$ is in $\mathcal{S}(\mathcal{D})$.

(2) We show that any disjunctive database not belonging to $\mathcal{R}$ is not in $\mathcal{S}(\mathcal{D})$. We recall that for a disjunction d, S_d denotes the set of facts appearing in d. Consider a disjunctive database $\mathcal{D}_{out}$ which is not in $\mathcal{R}$. Two cases may occur: (a) $reduction(\mathcal{D}) \not\subseteq \mathcal{D}_{out}$ or (b) $reduction(\mathcal{D}) \subseteq \mathcal{D}_{out}$ and $\exists d' \in \mathcal{D}_{out} - reduction(\mathcal{D})$ s.t. there is no $d \in reduction(\mathcal{D})$ which subsumes d'.
(a) As $reduction(\mathcal{D}) \not\subseteq \mathcal{D}_{out}$, there is a disjunction a in $reduction(\mathcal{D})$ which is not in $\mathcal{D}_{out}$. Two cases may occur:

- there exists $a_1 \in \mathcal{D}_{out}$ which subsumes a;
- the previous condition does not hold.

Let us consider the first case and let I be the interpretation $S - S_{a_1}$ where S is the set of facts appearing in $reduction(\mathcal{D})$. It is easy to see that I is a model of $reduction(\mathcal{D})$ (the only disjunctions that I could not satisfy are those ones that contain only facts in S_{a_1}; such disjunctions are not in $reduction(\mathcal{D})$ as they subsume a and $reduction(\mathcal{D})$ does not contain two disjunctions s.t. one subsumes the other). Then, there exists $M \subseteq I$ which is a minimal model of $reduction(\mathcal{D})$. As $a_1 \in \mathcal{D}_{out}$, each model of $\mathcal{D}_{out}$ contains a fact in S_{a_1}, then M is not a minimal model of $\mathcal{D}_{out}$ and so $\mathcal{MM}(reduction(\mathcal{D})) \neq \mathcal{MM}(\mathcal{D}_{out})$. Hence $\mathcal{D}_{out} \notin \mathcal{S}(\mathcal{D})$.
We consider now the second case. We show that $\mathcal{D}_{out} \notin \mathcal{S}(\mathcal{D})$ in a similar way to the previous case. Let I be the interpretation $S - S_a$ where S is the set of facts appearing in $\mathcal{D}_{out}$. It is easy to see that I is a model of $\mathcal{D}_{out}$ (the only disjunctions that I could not satisfy are those ones which contain only facts in S_a; such disjunctions are not in $\mathcal{D}_{out}$ as $\mathcal{D}_{out}$ contains neither a nor a disjunction which subsumes a). Then, there exists $M \subseteq I$ which is a minimal model of $\mathcal{D}_{out}$. As $a \in reduction(\mathcal{D})$, each model of $reduction(\mathcal{D})$ contains a fact in S_a, then M is not a minimal model of $reduction(\mathcal{D})$; hence $\mathcal{D}_{out} \notin \mathcal{S}(\mathcal{D})$.
(b) Let I be the interpretation $S - S_{d'}$ where S is the set of facts appearing in $reduction(\mathcal{D})$. It is easy to see that I is a model of $reduction(\mathcal{D})$ (the only disjunctions that I could not satisfy are d' and those ones which subsume d'). Then, there exists $M \subseteq I$ which is a minimal model of $reduction(\mathcal{D})$. As $d' \in \mathcal{D}_{out}$, each model of $\mathcal{D}_{out}$ contains a fact in $S_{d'}$, then M is not a minimal model of $\mathcal{D}_{out}$; hence $\mathcal{D}_{out} \notin \mathcal{S}(\mathcal{D})$. □

Corollary 1 *Given a database D and a set F of denial constraints, then $\mathcal{D}(D, F)$ is the canonical disjunctive database whose minimal models are the repairs for D and F.*

Proof. Straightforward from Theorem 1 and 2. □

From now on, we will denote by $\mathcal{D}_{min}(D, F)$ the canonical disjunctive database whose minimal models are the repairs for a database D and a set F of denial constraints. Whenever D and F are clear from the context, we simply write $\mathcal{D}_{min}$ instead of $\mathcal{D}_{min}(D, F)$.

4 Functional dependencies

In this section we study the size of the canonical disjunctive database representing the repairs of a database in the presence of functional dependencies. Specifically, we show that when the constraints consist of only one key, the canonical disjunctive database is of linear size, whereas for one non-key functional dependency or two keys the size of the canonical database may be exponential.
We observe that in the presence of only one functional dependency, the conflict hypergraph has a regular structure that "induces" a regular disjunctive database which can be identified without performing Algorithm 1. When two key constraints are considered, we are not able to provide such a characterization; this is because the conflict hypergraph can have an irregular structure and it is harder to identify a pattern for $\mathcal{D}_{min}$.

Given a disjunction d, we denote by $||d||$ the number of facts occurring in d. The size of a disjunctive database $\mathcal{D}$, denoted as $||\mathcal{D}||$, is the number of facts occurring in it, that is $||\mathcal{D}|| = \sum_{d \in \mathcal{D}} ||d||$. We study the size $||\mathcal{D}_{min}||$ of $\mathcal{D}_{min}$ as a function of the size of the given database.

One key. Given a relation r and a key constraint k stating that the set X of attributes is a key of r, we denote by $\mathit{cliques}(r, k)$ the partition of r into $n = |\pi_X(r)|$ sets $C_1, \ldots, C_n$, called *cliques*, s.t. each C_i does not contain two facts with different values on X. Observe that (i) facts in the same clique are pairwise conflicting with each other, (ii) the set of repairs of r w.r.t. k is $\{\{t_1, \ldots, t_n\} \mid t_i \in C_i \text{ for } i = 1..n\}$.

Proposition 1 *Given a relation r and a key constraint k, then $\mathcal{D}_{min}$ is equal to*

$$\{t_1 \vee \ldots \vee t_m \mid \exists C = \{t_1, \ldots, t_m\} \in \mathit{cliques}(r, k)\}$$

Proof. It is straightforward to see that the minimal models of the disjunctive database reported above are the repairs of r w.r.t. k; since it coincides with its reduction, Theorem 2 implies that it is the canonical one. □

It is easy to see that when one key constraint is considered, $||\mathcal{D}_{min}|| = |r|$.

Proposition 2 *Given a relation and a key constraint, $\mathcal{D}_{min}$ is computed in polynomial time by Algorithm 1.*

Proof. It is easy to see that after the first loop (steps 3-5) Algorithm 1 produces $\mathcal{D}_{min}$ and, after that, step 9 is never performed. □

Two keys. We now show that, in the presence of two key constraints, $\mathcal{D}_{min}$ may have exponential size. Let D_n ($n > 0$) be the family of databases, containing $3n$ facts, of the following form:

	A	B
t_{11}	a	b_1
$\vdots$	$\vdots$	$\vdots$
t_{n1}	a	b_n
t_{12}	a_1	b_1
t_{13}	a_1	b'_1
$\vdots$	$\vdots$	$\vdots$
t_{n2}	a_n	b_n
t_{n3}	a_n	b'_n

Let $D \in D_n$ and A, B be two keys. The conflict hypergraph for D w.r.t. the two key constraints consists of the following edges:

$$\{\{t_{i1}, t_{j1}\} \mid 1 \leq i, j \leq n \ \wedge \ i \neq j\} \ \cup \ \{\{t_{i1}, t_{i2}\} \mid 1 \leq i \leq n\} \ \cup \ \{\{t_{i2}, t_{i3}\} \mid 1 \leq i \leq n\}$$

Thus, the conflict hypergraph contains a clique $\{t_{11}, \ldots, t_{n1}\}$ of size n and, moreover, t_{i1} is connected to t_{i2} which is in turn connected to t_{i3} ($i = 1..n$).

Example 2 The conflict hypergraph for a database in D_4, assuming that A and B are two keys, is reported in Figure 1.

The following proposition identifies the canonical disjunctive database for a database in D_n for which A and B are keys; such a disjunctive database has exponential size.

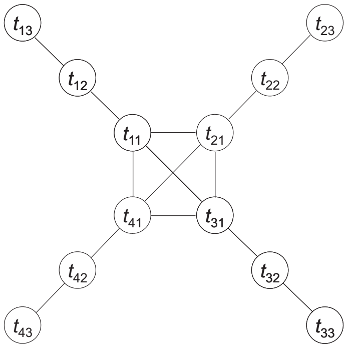

Fig. 1 Conflict hypergraph for a database in D_4 w.r.t. A, B key constraints

Proposition 3 *Consider a database D in D_n and a set of constraints F consisting of two keys, A and B. Then $\mathcal{D}_{min}$ is equal to $\mathcal{D}$ where*

$$\mathcal{D} = \{t_{i2} \vee t_{i3} \mid 1 \leq i \leq n\} \ \cup \{t_{i1} \vee t_{i2} \vee \bigvee_{z=1..n \ \wedge \ z \neq i} t_z \mid 1 \leq i \leq n \ \wedge t_z \in \{t_{z1}, t_{z3}\}\}$$

Proof. First of all, we show that the minimal models of $\mathcal{D}$ are the repairs of D w.r.t. F; in particular we prove that (1) $\mathcal{MM}(\mathcal{D}) \subseteq repairs(D, F)$ and (2) $\mathcal{MM}(\mathcal{D}) \supseteq repairs(D, F)$.

(1) Consider a minimal model $m \in \mathcal{MM}(D)$. First we show that (a) m is consistent w.r.t. F and next (b) that it is maximal.

(a) Let E be the set of edges of $\mathcal{G}_{D,F}$. First we show that for each $e = \{t', t''\}$ in E and pair of disjunctions $d' = t' \vee D'$, $d'' = t'' \vee D''$ in $\mathcal{D}$ s.t. D' (resp. D'') does not contain t'' (resp. t'), there is a disjunction in $\mathcal{D}$ which is equal to or subsumes $D' \vee D''$; next we show that this property implies that m is consistent w.r.t. F. We recall that E is the union of the following three sets:

$$E_1 = \{\{t_{i1}, t_{j1}\} \mid 1 \leq i, j \leq n \ \wedge i \neq j\}$$

$$E_2 = \{\{t_{i1}, t_{i2}\} \mid 1 \leq i \leq n\}$$

$$E_3 = \{\{t_{i2}, t_{i3}\} \mid 1 \leq i \leq n\}$$

Let us consider the case where $e \in E_1$, that is $e = \{t_{i1}, t_{j1}\}$ $(1 \leq i, j \leq n \ \wedge \ i \neq j)$. Then a disjunction in $\mathcal{D}$ containing t_{i1} but not t_{j1} is of the form

$$d'_1 \ : \ t_{i1} \vee t_{i2} \vee t_{j3} \vee \bigvee_{z=1..n \ \wedge \ z \neq i,j} t'_z$$

where $t'_z \in \{t_{z1}, t_{z3}\}$, or of the form

$$d'_2 \ : \ t_{h1} \vee t_{h2} \vee t_{i1} \vee t_{j3} \vee \bigvee_{z=1..n \ \wedge \ z \neq h,i,j} t'_z$$

where $1 \leq h \leq n \ \wedge \ h \neq i, j$ and $t'_z \in \{t_{z1}, t_{z3}\}$. Likewise, a disjunction in $\mathcal{D}$ that contains t_{j1} but not t_{i1} is of the form

$$d''_1 \ : \ t_{j1} \vee t_{j2} \vee t_{i3} \vee \bigvee_{z=1..n \ \wedge \ z \neq i,j} t''_z$$

where $t''_z \in \{t_{z1}, t_{z3}\}$, or of the form

$$d''_2 \; : \; t_{k1} \vee t_{k2} \vee t_{j1} \vee t_{i3} \vee \bigvee_{z=1..n \;\wedge\; z \neq k,i,j} t''_z$$

where $1 \leq k \leq n \;\wedge\; k \neq i, j$ and $t''_z \in \{t_{z1}, t_{z3}\}$. In all the four possible cases, there is disjunction in $\mathcal{D}$ which subsumes $D' \vee D''$:

- if $d' = d'_1$ and $d'' = d''_1$, then there exist both $t_{j2} \vee t_{j3}$ and $t_{i2} \vee t_{i3}$ in $\mathcal{D}$ which subsume $D' \vee D''$;
- if $d' = d'_1$ and $d'' = d''_2$, then there exists $t_{i2} \vee t_{i3}$ in $\mathcal{D}$ which subsumes $D' \vee D''$;
- if $d' = d'_2$ and $d'' = d''_1$, then there exists $t_{j2} \vee t_{j3}$ in $\mathcal{D}$ which subsumes $D' \vee D''$;
- if $d' = d'_2$ and $d'' = d''_2$, then both $t_{h1} \vee t_{h2} \vee t_{i3} \vee t_{j_3} \vee \bigvee_{z=1..n \;\wedge\; z \neq h,i,j} t'_z$ and $t_{k1} \vee t_{k2} \vee t_{i3} \vee t_{j_3} \vee \bigvee_{z=1..n \;\wedge\; z \neq k,i,j} t''_z$, which are in $\mathcal{D}$, subsume $D' \vee D''$.

Let us consider the case where $e \in E_2$, namely $e = \{t_{i1}, t_{i2}\}$ $(1 \leq i \leq n)$. A disjunction containing t_{i1} but not t_{i2} is of the form

$$t_{k1} \vee t_{k2} \vee t_{i1} \vee \bigvee_{z=1..n \;\wedge\; z \neq i,k} t_z$$

where $1 \leq k \leq n \;\wedge\; k \neq i$ and $t_z \in \{t_{z1}, t_{z3}\}$, whereas a disjunction containing t_{i2} but not t_{i1} is of the form $t_{i2} \vee t_{i3}$. Thus, $D' \vee D''$, which is equal to

$$t_{k1} \vee t_{k2} \vee t_{i3} \vee \bigvee_{z=1..n \;\wedge\; z \neq i,k} t_z$$

is in $\mathcal{D}$. Finally, consider the last case where $e \in E_3$, that is $e = \{t_{i2}, t_{i3}\}$ $(1 \leq i \leq n)$. A disjunction containing t_{i2} but not t_{i3} is of the form

$$t_{i1} \vee t_{i2} \vee \bigvee_{z=1..n \;\wedge\; z \neq i} t'_z$$

where $t'_z \in \{t_{z1}, t_{z3}\}$, whereas a disjunction containing t_{i3} but not t_{i2} is of the form

$$t_{h1} \vee t_{h2} \vee t_{i3} \vee \bigvee_{z=1..n \;\wedge\; z \neq h,i} t''_z$$

where $1 \leq h \leq n \;\wedge\; h \neq i$ and $t''_z \in \{t_{z1}, t_{z3}\}$; $D' \vee D''$ is subsumed or equal to the disjunction

$$t_{h1} \vee t_{h2} \vee t_{i1} \vee \bigvee_{z=1..n \;\wedge\; z \neq h,i} t''_z$$

which is in $\mathcal{D}$.

Assume by contradiction that m is not consistent. Then there are two facts $t_a, t_b \in m$ s.t. $\{t_a, t_b\} \in E$. Let $S_{t_a} = \{D \mid t_a \vee D \in \mathcal{D}$ and D does not contain $t_b\}$ and $S_{t_b} = \{D \mid t_b \vee D \in \mathcal{D}$ and D does not contain $t_a\}$. As we have seen before, both these sets are not empty. We have previously proved that for each $D_a \in S_{t_a}$ and $D_b \in S_{t_b}$ there is a disjunction in $\mathcal{D}$ which equals or subsumes $D_a \vee D_b$. Then, there is a set S_{t_x} among S_{t_a} and S_{t_b} s.t. m satisfies each D in S_{t_x}, otherwise there would be $D_a \in S_{t_a}, D_b \in S_{t_b}$ and a disjunction in $\mathcal{D}$ which is equal to or subsumes $D_a \vee D_b$ which is not satisfied by m. Consider the interpretation $m' = m - \{t_x\}$ and let t_y be the fact among t_a and t_b which is not t_x. We now show that m' is a model, that contradicts the minimality of m. Clearly, m' satisfies every disjunction in $\mathcal{D}$ which does

not contain t_x. As for the disjunctions in $\mathcal{D}$ containing t_x, it is easy to see that they are satisfied by m': disjunctions containing t_y are satisfied since $t_y \in m'$, disjunctions not containing t_y are satisfied as well since m' satisfies every disjunction in S_{t_x}. Hence m is consistent w.r.t. F.
(b) Now we prove that m is a maximal (consistent) subset of D. First of all, we note that for each fact $t \in D$ there is a disjunction $t \vee t_1 \vee \ldots \vee t_n$ in $\mathcal{D}$ s.t. $t_1, \ldots, t_n$ are facts conflicting with t:

- for the facts t_{i2} and t_{i3} $(i = 1..n)$ such disjunctions are $t_{i2} \vee t_{i3}$;
- for the facts t_{i1} $(i = 1..n)$ such disjunctions are $t_{i1} \vee t_{i2} \vee \bigvee_{z=1..n\ \wedge\ z \neq i} t_{z1}$.

Assume by contradiction that m is not a maximal (consistent) subset of D. Then there exists $m' \supset m$ which is consistent. Let t be a fact in m' but not in m. Since m' is consistent, each fact conflicting with t is not in m' and, thus, neither in m. This implies that m doesn't satisfy the disjunction $t \vee t_1 \vee \ldots \vee t_n$ containing t and some fact conflicting with it: the fact that m is a model is contradicted.
(2) Consider a repair r for D and F. We show first (a) that r is a model of $\mathcal{D}$ and next (b) that it is a minimal model.
(a) Suppose by contradiction that r is not a model of $\mathcal{D}$, then there is a disjunction $d \in \mathcal{D}$ which is not satisfied by r. Specifically, d is either of the form $t_{i2} \vee t_{i3}$ $(1 \leq i \leq n)$ or $t_{i1} \vee t_{i2} \vee \bigvee_{z=1..n\ \wedge\ z \neq i} t_z$, $1 \leq i \leq n$ and $t_z \in \{t_{z1}, t_{z3}\}$. In the former case, $r \cup \{t_{i3}\}$ is consistent, since the only fact conflicting with t_{i3}, namely t_{i2}, is not in r. This contradicts the maximality of r. As for the latter case, let $T_3 = \{t_{j3} \mid t_{j3} \text{ appears in } d\}$. For each $t_{j3} \in T_3$ we have that $t_{j2} \in r$, because r does not contain t_{j3} and t_{j3} is conflicting only with t_{j2} (if t_{j2} was not in r, then r would not be maximal). Then for each $t_{j3} \in T_3$, since r contains t_{j2}, it does not contain t_{j1} otherwise it would not be consistent. Thus r does not contain any fact t_{k1} with $1 \leq k \leq n\ \wedge\ k \neq i$. Since r contains neither the facts t_{k1}'s nor t_{i2}, which are all the facts conflicting with t_{i1}, then $r \cup \{t_{i1}\}$ is consistent (observe that $t_{i1} \notin r$). This contradicts the maximality of r. Hence r is a model of $\mathcal{D}$.
(b) We now show that r is a minimal model of $\mathcal{D}$ reasoning by contradiction. Assume that there exists a model $m' \subset r$ of $\mathcal{D}$ and let t be a fact in r but not in m'. All the facts conflicting with t are not in r as r is consistent. The same holds for m' since it is a (proper) subset of r. We recall that for each fact $t' \in D$ there is a disjunction in $\mathcal{D}$ containing t' and only facts conflicting with t'; then there is a disjunction $d : t \vee t_1 \vee \ldots \vee t_n$ in $\mathcal{D}$ s.t. $t_1, \ldots, t_n$ are facts conflicting with t. Since m' does not satisfy d, it is not a model, thus we get a contradiction. Hence r is a minimal model of $\mathcal{D}$.

We have shown that the minimal models of $\mathcal{D}$ are the repairs of D w.r.t. F. Since $\mathcal{D} = \mathit{reduction}(\mathcal{D})$, from Theorem 2 we have that $\mathcal{D}$ is the canonical disjunctive database whose minimal models are the repairs of D w.r.t. F. □

Corollary 2 *Consider a database D in D_n and let A and B be two keys;* $||\mathcal{D}_{min}|| = 2n + (n+1) \cdot n2^{n-1}$.

Proof. From Proposition 3, it is easy to see that $\mathcal{D}_{min}$ contains n disjunctions of 2 facts and $n2^{n-1}$ disjunctions of $n+1$ facts. □

One functional dependency. Given a relation r and a functional dependency $f : X \rightarrow Y$, we denote by $\mathit{cliques}(r, f)$ the partition of r into $n = |\pi_X(r)|$ sets $C_1, \ldots, C_n$, called *cliques*, s.t. each C_i does not contain two facts with different values on X. For each clique C_i in *cliques*(r, f) we denote by *clusters*(C_i) the partition of C_i into $m_i = |\pi_Y(C_i)|$ sets $G_1, \ldots, G_{m_i}$, called *clusters*, s.t. each cluster doesn't contain two facts with different values on Y. It is worth noting that (i) facts in the same cluster are not conflicting each other, (ii) given two different clusters G_1, G_2 of the same clique, each fact in G_1 (resp. G_2) is conflicting with every fact in G_2 (resp. G_1), (iii) the set of repairs of r w.r.t. f is $\{G_1 \cup \ldots \cup G_n \mid G_i \in clusters(C_i) \text{ for } i = 1..n\}$.

Proposition 4 *Given a relation r and a functional dependency f, then $\mathcal{D}_{min}$ is equal to $\mathcal{D}$ where*

$$\begin{aligned}\mathcal{D} = \{t_1 \vee \ldots \vee t_k \mid \exists C \in cliques(r, f) \ s.t.\ clusters(C) = \{G_1, \ldots, G_k\}\\ and\ t_1 \in G_1, \ldots, t_k \in G_k\}\end{aligned}$$

Proof. We show first (1) that each minimal model of $\mathcal{D}$ is a repair for r and f and next (2) that each repair of r w.r.t. f is a minimal model of $\mathcal{D}$.
(1) Consider a minimal model m of $\mathcal{D}$. Let *cliques*$(r, f) = \{C_1, \ldots, C_n\}$ be the cliques for r and f. For each clique C_i in *cliques*(r, f) there is a cluster G_j in *clusters*$(C_i) = \{G_1, \ldots, G_k\}$ s.t. $G_j \subseteq m$ (otherwise m would not satisfy the disjunction $t_1 \vee \ldots \vee t_k$ in $\mathcal{D}$ where $t_h \in G_h$ and $t_h \notin m$, $h = 1..k$). Let $\overline{G}_1, \ldots, \overline{G}_n$ be such clusters, where each $\overline{G}_l$ is a cluster of C_l for $l = 1..n$. Since $\overline{G}_1 \cup \ldots \cup \overline{G}_n \subseteq m$ and $\overline{G}_1 \cup \ldots \cup \overline{G}_n \models \mathcal{D}$, then $m = \overline{G}_1 \cup \ldots \cup \overline{G}_n$, which is, as we have observed before, a repair.
(2) Consider a repair s in *repairs*(r, f). As s consists of one cluster for each clique, it is easy to see that s is a model of $\mathcal{D}$. We show that s is minimal by contradiction assuming that there exists $s' \subset s$ which is a model of $\mathcal{D}$. Let t be a fact in s which is not in s'. Let C_t and G_t be the clique and the cluster, respectively, containing t; moreover let *clusters*$(C_t) = \{G_t, G_1, \ldots, G_k\}$. The disjunction $t \vee t_1 \vee \ldots \vee t_k$, where $t_i \in G_i,\ i = 1..k$, which is in $\mathcal{D}$, is not satisfied by s' as s' contains exactly one cluster per clique (thus it does not contain any fact in $G_i,\ i = 1..k$) and does not contain t. This contradicts the fact that s' is a model. So s is a minimal model of $\mathcal{D}$.
Hence the minimal models of $\mathcal{D}$ are exactly the repairs for r and f; as $\mathcal{D}$ is equal to its reduction, Theorem 2 entails that $\mathcal{D} = \mathcal{D}_{min}$. □

Clearly, the size of $\mathcal{D}_{min}$ may be exponential if the functional dependency is a non-key dependency, as shown in the following example.

Example 3 Consider the relation r, consisting of $2n$ facts, reported below and the non-key functional dependency $A \to B$.

	A	B	C
t'_1	a	b_1	c_1
t''_1	a	b_1	c_2
⋮	⋮	⋮	⋮
t'_n	a	b_n	c_1
t''_n	a	b_n	c_2

There is a unique clique consisting of n clusters $G_i = \{t'_i, t''_i\}$, $i = 1..n$. Then $\mathcal{D}_{min} = \{t_1 \vee \ldots \vee t_n \mid t_i \in G_i \text{ for } i = 1..n\}$ and $||\mathcal{D}_{min}|| = n2^n$.

5 Cardinality-based repairs

In this section we consider *cardinality-based repairs*, that is consistent databases which minimally differ from the original database in terms of the number of facts in the symmetric difference (in the previous sections we have considered consistent databases for which the symmetric difference is minimal under set inclusion, we will refer to them as *S-repairs*).

We show that, likewise to what has been presented in Section 4, the size of the canonical disjunctive database (representing the cardinality-based repairs) is linear when only one key constraint is considered, whereas it may be exponential when two keys or one non-key functional dependency are considered.

It is easy to see that in the presence of only one key constraint the cardinality-based repairs coincide with the S-repairs, so the canonical disjunctive database is of linear size.

When the constraints consists of one functional dependency, it is easy to see that if for every clique its clusters have the same cardinality, then the cardinality-based repairs coincide with the S-repairs. This is the case for the database of Example 3, where the size of the canonical disjunctive database is exponential.

Finally, we consider the case where two key constraints are considered. We directly show that the size of the canonical disjunctive database is also exponential.

Lemma 1 *Consider a database D in D_n and a set of integrity constraints F consisting of two keys, A and B. Then the set of S-repairs is is equal to R where*

$$R = \{\{t_{12}, \ldots, t_{n2}\}\} \cup \{\ \{t_{i1}, t_{i3}\} \cup \bigcup_{j=1..n\ \wedge\ j \neq i} \{t_j\} \mid 1 \leq i \leq n\ \wedge\ t_j \in \{t_{j2}, t_{j3}\}\}$$

Proof. It is easy to see that each database in R is a S-repair.
Consider a S-repair r of D w.r.t. F. We show that r is in R using reasoning by cases:

- Suppose that $t_{13} \in r$. Then $t_{12} \notin r$ and either (1) $t_{11} \in r$ or (2) $t_{11} \notin r$.
  1. Since $t_{11} \in r$, for $j = 2..n$ $t_{j1} \notin r$ and either t_{j2} or t_{j3} is in r, that is $r = \{t_{11}, t_{13}, t_2, \ldots, t_n\}$ where $t_j \in \{t_{j2}, t_{j3}\},\ j = 2..n$. It is easy to see that $r \in R$.
  2. Since $t_{11} \notin r$, there exists $t_{k1} \in r$ with $2 \leq k \leq n$. Then $t_{k2} \notin r$ and $t_{k3} \in r$. For $j = 2..n\ \wedge\ j \neq k$, $t_{j1} \notin r$ and either t_{j2} or t_{j3} is in r, that is $r = \{t_{13}, t_{k1}, t_{k3}\} \cup \bigcup_{j=2..n\ \wedge\ j \neq k} \{t_j\}$ where $t_j \in \{t_{j2}, t_{j3}\}$. Clearly, $r \in R$.
- Suppose that $t_{13} \notin r$. Then $t_{12} \in r$ and $t_{11} \notin r$. Two cases may occur: either (1) there exists $t_{k1} \in r$ with $2 \leq k \leq n$ or (2) $t_{j1} \notin r$ for $j = 1..n$.
  1. Since $t_{k1} \in r$ then $t_{k2} \notin r$ and $t_{k3} \in r$. For $j = 2..n\ \wedge\ j \neq k$ $t_{j1} \notin r$ and either t_{j2} or t_{j3} is in r, that is $r = \{t_{12}, t_{k1}, t_{k3}\} \cup \bigcup_{j=2..n\ \wedge\ j \neq k} \{t_j\}$ where $t_j \in \{t_{j2}, t_{j3}\}$. It is easy to see that $r \in R$.
  2. $r = \{t_{12}, \ldots, t_{n2}\}$ which is in R. □

Corollary 3 *Consider a database D in D_n and a set of integrity constraints F consisting of two keys, A and B. Then the set of cardinality-based repairs is*

$$\{\ \{t_{i1}, t_{i3}\} \cup \bigcup_{j=1..n\ \wedge\ j \neq i} \{t_j\} \mid 1 \leq i \leq n\ \wedge\ t_j \in \{t_{j2}, t_{j3}\}\}$$

Proof. Straightforward from Lemma 1. □

The following proposition identifies the canonical disjunctive database for a database in D_n for which A and B are keys; such a disjunctive database is of exponential size. In the following proposition and corollary, $\mathcal{D}_{min}$ denotes the canonical disjunctive database representing the set of cardinality-based repairs.

Proposition 5 *Consider a database D in D_n and a set of integrity constraints F consisting of two keys, A and B. Then the canonical disjunctive database $\mathcal{D}_{min}$ is equal to $\mathcal{D}$ where*

$$\mathcal{D} = \{t_{i2} \vee t_{i3} \mid 1 \leq i \leq n\} \;\cup \{t_1 \vee \ldots \vee t_n | \; t_i \in \{t_{i1}, t_{i3}\}, \; i = 1..n\}$$

Proof. We first show that (1) each cardinality-based repair of D w.r.t. F is a minimal model of $\mathcal{D}$ and next that (2) each minimal model of $\mathcal{D}$ is a cardinality-based repair.
(1) Consider a cardinality-based repair r of D w.r.t. F. We show first that (a) r is a model of $\mathcal{D}$ and next that (b) it is a minimal model.
(a) From Corollary 3, it is easy to see that r satisfies each disjunction $t_{i2} \vee t_{i3}$ in $\mathcal{D}$, $1 \leq i \leq n$. Since Corollary 3 entails that there exists $1 \leq j \leq n$ s.t. $\{t_{j1}, t_{j3}\} \subseteq r$, then r satisfies each disjunction $t_1 \vee \ldots \vee t_n$ in $\mathcal{D}$ (where $t_i \in \{t_{i1}, t_{i3}\}, \; i = 1..n$). Thus r is a model of $\mathcal{D}$.
(b) We observe that for each fact $t \in D$ there is a disjunction $t \vee t_1 \vee \ldots \vee t_n$ in $\mathcal{D}$ s.t. $t_1, \ldots, t_n$ are facts conflicting with t: for the facts t_{i2} and t_{i3} $(i = 1..n)$ such disjunctions are $t_{i2} \vee t_{i3}$; for the facts t_{i1} $(i = 1..n)$ there is the disjunction $t_{11} \vee \ldots \vee t_{n1}$. In the same way as in Proposition 3, it can be shown that r is a minimal model of $\mathcal{D}$.
(2) Consider a minimal model m of $\mathcal{D}$. The fact that m is a S-repair of D w.r.t. F can be shown in the same way as in Proposition 3.
It is easy to see that $\{t_{12}, \ldots, t_{n2}\}$ is not a model of $\mathcal{D}$ and then, from Lemma 1 and Corollary 3, m is a cardinality-based repair of D w.r.t. F.

We have shown that $\mathcal{D}$ represents the cardinality-based repairs of D w.r.t. F; since $\mathcal{D} = \mathit{reduction}(\mathcal{D})$, from Theorem 2 we have that $\mathcal{D}$ is the canonical one. □

Corollary 4 *Consider a database D in D_n and let A and B be two keys; $||\mathcal{D}_{min}|| = 2n + n2^n$.*

Proof. From Proposition 5, it is easy to see that $\mathcal{D}_{min}$ contains n disjunctions of 2 facts and 2^n disjunctions of n facts. □

6 Conclusions

In this paper we have addressed the problem of representing, by means of a disjunctive database, the set of repairs of a database w.r.t. a set of denial constraints. We have shown that, given a database and a set of denial constraints, there exists a unique canonical disjunctive database representing their repairs: any disjunctive database with the same set of minimal models is a superset of the canonical one, containing in addition disjunctions which are subsumed by the disjunctions in the canonical one. We have proposed an algorithm to compute the canonical disjunctive database. We have shown that the size of the canonical disjunctive database is linear when only one key is considered, but it may be exponential in the presence of two keys or one non-key functional dependency. We have shown that these results hold also when cardinality-based repairs are considered.

Future work in this area could explore different representations for the set of repairs. For instance, one can consider formulas with negation or non-clausal formulas. Such formulas can be more succinct than disjunctive databases, making query evaluation, however, potentially harder. We also observe that in the case of the repairs of a single relation the resulting disjunctive database consists of disjunctions of elements of this relation. It has been recognized that such disjunctions should be supported by database management systems [4]. Moreover, one could consider *restricting* inconsistent databases in such a way that the resulting repairs can be represented by relational databases with *OR-objects* [12]. In this case, one could use the techniques for computing *certain* query answers over databases with OR-objects [13] to compute *consistent* query answers over inconsistent databases. Finally, other kinds of representations of sets of possible worlds, e.g., *world-set decompositions* [1], should be considered. For example, the set of repairs of the database in Example 3 can be represented as a world-set decomposition of polynomial size.

Acknowledgements Jan Chomicki acknowledges the support of National Science Foundation under grant IIS-0119186. Jerzy Marcinkowski was partially supported by Polish Ministry of Science and Higher Education research project N206 022 31/3660, 2006/2009.

References

1. Lyublena Antova, Christoph Koch, and Dan Olteanu. 10^{10^6} worlds and beyond: Efficient representation and processing of incomplete information. In *International Conference on Data Engineering (ICDE)*, pages 606–615, 2007.
2. Marcelo Arenas, Leopoldo E. Bertossi, and Jan Chomicki. Consistent query answers in inconsistent databases. In *ACM Symposium on Principles of Database Systems (PODS)*, pages 68–79, 1999.
3. Marcelo Arenas, Leopoldo E. Bertossi, and Jan Chomicki. Answer sets for consistent query answering in inconsistent databases. *Theory and Practice of Logic Programming*, 3(4-5):393–424, 2003.
4. Omar Benjelloun, Anish Das Sarma, Alon Y. Halevy, Martin Theobald, and Jennifer Widom. Databases with uncertainty and lineage. *VLDB J.*, 17(2):243–264, 2008.
5. Leopoldo E. Bertossi. Consistent query answering in databases. *SIGMOD Record*, 35(2):68–76, 2006.
6. Leopoldo E. Bertossi and Jan Chomicki. Query answering in inconsistent databases. In *Logics for Emerging Applications of Databases*, pages 43–83, 2003.
7. Andrea Calì, Domenico Lembo, and Riccardo Rosati. Query rewriting and answering under constraints in data integration systems. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pages 16–21, 2003.
8. Jan Chomicki. Consistent query answering: Five easy pieces. In *International Conference on Database Theory (ICDT)*, pages 1–17, 2007.
9. Jan Chomicki and Jerzy Marcinkowski. Minimal-change integrity maintenance using tuple deletions. *Information and Computation*, 197(1-2):90–121, 2005.
10. José Alberto Fernández and Jack Minker. Semantics of disjunctive deductive databases. In *International Conference on Database Theory (ICDT)*, pages 21–50, 1992.
11. Gianluigi Greco, Sergio Greco, and Ester Zumpano. A logical framework for querying and repairing inconsistent databases. *IEEE Transactions on Knowledge and Data Engineering*, 15(6):1389–1408, 2003.
12. Tomasz Imielinski, Shamim A. Naqvi, and Kumar V. Vadaparty. Incomplete objects - a data model for design and planning applications. In *ACM SIGMOD Conference*, pages 288–297, 1991.
13. Tomasz Imielinski, Ron van der Meyden, and Kumar V. Vadaparty. Complexity tailored design: A new design methodology for databases with incomplete information. *Journal of Computer and System Sciences*, 51(3):405–432, 1995.

14. Andrei Lopatenko and Leopoldo E. Bertossi. Complexity of consistent query answering in databases under cardinality-based and incremental repair semantics. In *International Conference on Database Theory (ICDT)*, pages 179–193, 2007.
15. Jack Minker and Dietmar Seipel. Disjunctive logic programming: A survey and assessment. In *Computational Logic: Logic Programming and Beyond*, pages 472–511, 2002.
16. Moshe Y. Vardi. The complexity of relational query languages (extended abstract). In *ACM Symposium on Theory of Computing (STOC)*, pages 137–146, 1982.