Ann Math Artif Intell
DOI 10.1007/s10472-009-9159-0

Disjunctive databases for representing repairs

Cristian Molinaro · Jan Chomicki · Jerzy Marcinkowski

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 subset-based repairs and cardinality-based repairs.

Keywords Inconsistent databases · Incomplete databases · Repairs · Disjunctive databases

Mathematics Subject Classifications (2000) 68P15 · 68T37

C. Molinaro (✉)
DEIS, Universitá della Calabria, Via P. Bucci 41C, 87036 Rende (CS), Italy
e-mail: cmolinaro@deis.unical.it

J. 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

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

Published online: 29 September 2009

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* [3]. 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 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 [11], whereas surveys on this topic are [7, 9]. Recent work in this area has, however, been tackling also the problem of *database repairing* under a variety of repair semantics [1, 5, 8].

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. Consistent query answers might be obtained by querying the disjunctive database under cautious reasoning. Disjunctive databases have been studied for a long time [13, 15, 16, 20]. More recently, they have again attracted attention in the database research community because of potential applications in

data integration, extraction and cleaning [6]. 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 [4, 10, 14].

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 [18].

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:

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

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

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

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

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

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

2.2 Inconsistent databases

As 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 [3]. The symmetric difference is used to capture the distance between two databases. For *subset-based* repairs, that we will call simply repairs, the symmetric difference has to be minimal under set inclusion. Because we consider denial constraints, repairs are *maximal consistent subsets* of the original database. 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 DB w.r.t. a set F of denial constraints is denoted by $repairs(DB, F)$.

Given a database DB and a set F of denial constraints, the *conflict hypergraph* [12] for DB and F, denoted by $\mathcal{G}_{DB,F}$, is a hypergraph whose set of vertices is the set of facts of DB, whereas the set of edges consists of all the subset-minimal set of facts of DB violating together a denial constraint in F. Thus, $e \subseteq DB$ is an edge of $\mathcal{G}_{DB,F}$ if and only if (1) e violates a denial constraint in F, i.e., there exist a denial constraint

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

in F and a substitution θ s.t. $p_i(\theta(\overline{X}_i)) \in e$ for $i = 1..n$ and $\varphi(\theta(\overline{X}_1), \ldots, \theta(\overline{X}_n))$ is true (recall that φ is a conjunction of atoms referring to built-in, arithmetic or comparison, predicates); and (2) there is no $e' \subsetneq e$ which violates a denial constraint in F.

A fact t of DB is said to be *conflicting* (w.r.t. F) if there exists an edge $\{t, t_1, \ldots, t_m\}$ ($m \geq 0$) in $\mathcal{G}_{DB,F}$. For a fact t of DB, we denote by $edges_{DB,F}(t)$ the set of edges of $\mathcal{G}_{DB,F}$ containing t, i.e., $edges_{DB,F}(t) = \{e \mid e \text{ is an edge of } \mathcal{G}_{DB,F} \text{ and } t \in e\}$.

2.3 Disjunctive databases

A disjunction is a finite set of facts. A disjunction containing exactly one fact is called a *singleton* disjunction. A disjunctive database $\mathcal{D}$ is a finite set of non-empty disjunctions. A database DB is a model of $\mathcal{D}$ if for every $d \in \mathcal{D}$, $d \cap DB \neq \emptyset$ (we will also say that DB *satisfies* $\mathcal{D}$); DB is minimal if there is no $DB' \subsetneq DB$ s.t. DB' is a model of $\mathcal{D}$. We denote by $\mathcal{MM}(\mathcal{D})$ the set of minimal models of $\mathcal{D}$. Given

two distinct disjunctions d_1 and d_2 in $\mathcal{D}$, we say that d_1 *subsumes* d_2 iff $d_1 \subsetneq d_2$. 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 refer to *data complexity* [21], i.e., 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 DB w.r.t. a set of integrity constraints F as a function of the number of facts in DB.

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).

Note that a disjunctive database representing the repairs of a database DB w.r.t. a set F of denial constraints may be obtained by rewriting the following DNF formula in CNF:

$$\bigvee_{R \in repairs(DB,F)} \bigwedge_{t \in R} t$$

A drawback of this approach is that the construction of the DNF formula requires the computation of all the repairs, which are, in general, exponentially many. In some cases, e.g. in the presence of one key constraint, even if the number of repairs is exponential, the disjunctive database can be computed in polynomial time (see next section).

Example 2 Consider the relation below where A is a key.

	A	B
t'_1	a_1	b_1
t''_1	a_1	b_2
$\vdots$	$\vdots$	$\vdots$
t'_n	a_n	b_1
t''_n	a_n	b_2

There are 2^n repairs, namely $\{\{t_1, \ldots, t_n\} \mid t_i \in \{t'_i, t''_i\}$ for $i = 1..n\}$. Thus, the corresponding DNF formula consists of 2^n disjuncts, where each disjunct is the conjunction of n facts. The DNF formula needs to be converted in CNF. As we will show in the next section, there exists a disjunctive database of linear size; we propose an algorithm which computes it in polynomial time.

The following proposition identifies necessary conditions that a disjunctive database has to satisfy in order to its minimal models be the repairs for a database and a set of denial constraints. The algorithm that we propose to compute a disjunctive database representing a set of repairs draws on these conditions.

Proposition 1 *Let DB be a database and F a set of denial constraints. Given a disjunctive database $\mathcal{D}$ whose minimal models are the repairs of DB w.r.t. F, then:*

1. *for each fact $t \in DB$,*
   (a) *if $\{t\}$ is an edge of $\mathcal{G}_{DB,F}$, then $\{t\} \notin \mathcal{D}$;*
   (b) *otherwise, let $edges_{DB,F}(t) = \{e_1, \ldots, e_m\}$ and $\mathcal{D}' = \{\{t\} \cup \{t_1\} \cup \cdots \cup \{t_m\} \mid t_i \in e_i - \{t\}$ for $i = 1..m\}$. Then, for each disjunction $d' \in \mathcal{D}'$ there is a disjunction $d \in \mathcal{D}$ s.t. $d \subseteq d'$.*
2. *for each edge $\{t_1, \ldots, t_k\}$ of $\mathcal{G}_{DB,F}$ with $k \geq 2$, if there are k disjunctions $d_1, \ldots, d_k \in \mathcal{D}$ s.t. $d_j \cap \{t_1, \ldots, t_k\} = \{t_j\}$ for $j = 1..k$, then there is a disjunction d in $\mathcal{D}$ s.t. $d \subseteq (d_1 \cup \cdots \cup d_k) - \{t_1, \ldots, t_k\}$.*

Proof

1.*a* Straightforward.

1.*b* Suppose by contradiction that the condition does not hold and let d' be a disjunction in $\mathcal{D}'$ s.t. there is no disjunction $d \in \mathcal{D}$ s.t. $d \subseteq d'$. Let S be the set of facts appearing in $\mathcal{D}$ and $M = S - d'$. It is easy to see that M is a model of $\mathcal{D}$ (the only disjunctions that M could not satisfy are those ones that contain only facts in d') and thus there exists $M' \subseteq M$ which is a minimal model of $\mathcal{D}$. Since $M' \cup \{t\}$ is consistent, then M' is not a repair, which is a contradiction.

2. Suppose by contradiction that there exist an edge $\{t_1, \ldots, t_k\}$ of $\mathcal{G}_{DB,F}$ and k disjunctions $d_1, \ldots, d_k \in \mathcal{D}$ s.t. $d_j \cap \{t_1, \ldots, t_k\} = \{t_j\}$ for $j = 1..k$, and there is no disjunction d in $\mathcal{D}$ s.t. $d \subseteq (d_1 \cup \cdots \cup d_k) - \{t_1, \ldots, t_k\}$. Let S be the set of facts appearing in $\mathcal{D}$, $d' = (d_1 \cup \cdots \cup d_k) - \{t_1, \ldots, t_k\}$ and $M = S - d'$. It is easy to see that M is a model of $\mathcal{D}$ (the only disjunctions that M could not satisfy are those ones that contain only facts in d') and thus there exists $M' \subseteq M$ which is a minimal model of $\mathcal{D}$. Note that $\{t_1, \ldots, t_k\} \subseteq M'$, otherwise there would be a disjunction d_i $(1 \leq i \leq k)$ not satisfied by M'. Hence, M' is inconsistent w.r.t. F, which is a contradiction. □

Intuitively, a disjunctive database representing a set of repairs has to satisfy conditions 1.*a* and 2 of the previous proposition in order to its minimal models be consistent, whereas condition 1.*b* has to be satisfied in order to minimal models be maximal (consistent) subsets of the original database.

The following algorithm computes a disjunctive database representing the repairs of a database w.r.t. a set of denial constraints.

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

1. $DB' = DB - \{t \mid \{t\}$ is an edge of $\mathcal{G}_{DB,F}\}$.
2. $\widehat{\mathcal{D}} = \{\{t\} \cup \{t_1\} \cup \cdots \cup \{t_m\} \quad | \quad t \in DB' \wedge edges_{DB',F}(t) = \{e_1, \ldots, e_m\} \wedge t_i \in e_i - \{t\}$ *for* $i = 1..m\}$.
3. Construct a maximal sequence

$$\widehat{\mathcal{D}} = \widehat{\mathcal{D}}_0 \subsetneq \widehat{\mathcal{D}}_1 \subsetneq \cdots \subsetneq \widehat{\mathcal{D}}_n$$

such that for each $i \in \{1, \ldots, n\}$, for some $k \geq 2$, there exist facts $t_1, \ldots, t_k \in DB'$ and disjunctions $d_1, \ldots, d_k \in \widehat{\mathcal{D}}_{i-1}$ such that:

(i) $\{t_1, \ldots, t_k\}$ is an edge of $\mathcal{G}_{DB',F}$;
(ii) for each $j \in [1..k]$, $|d_j| \geq 2$ and $d_j \cap \{t_1, \ldots, t_k\} = \{t_j\}$; and
(iii) $\widehat{\mathcal{D}}_i = \widehat{\mathcal{D}}_{i-1} \cup \{d\}$ where $d = (d_1 \cup \cdots \cup d_k) - \{t_1, \ldots, t_k\}$.

4. Return *reduction*$(\widehat{\mathcal{D}}_n)$.

We denote by $\mathcal{D}(DB, F)$ the disjunctive database returned by Algorithm 1 with the input consisting of a database DB and a set F of denial constraints.

Example 3 Consider a database $DB = \{t_1, t_2, t_3, t_4, t_5\}$ and a set F of denial constraints s.t. the edges of $\mathcal{G}_{DB,F}$ are $\{t_1, t_2, t_3\}, \{t_3, t_4\}, \{t_4, t_5\}$. Then, $\widehat{\mathcal{D}}_0$ contains the following disjunctions:

$$\{t_1, t_2\} \tag{1}$$
$$\{t_1, t_3\} \tag{2}$$
$$\{t_2, t_3\} \tag{3}$$
$$\{t_1, t_3, t_4\} \tag{4}$$
$$\{t_2, t_3, t_4\} \tag{5}$$
$$\{t_3, t_4, t_5\} \tag{6}$$
$$\{t_4, t_5\} \tag{7}$$

By considering the edge $\{t_3, t_4\}$ and the disjunctions (2) and (7), we can obtain $\widehat{\mathcal{D}}_1 = \widehat{\mathcal{D}}_0 \cup \{d\}$, where d is

$$\{t_1, t_5\} \tag{8}$$

Let us consider again the edge $\{t_3, t_4\}$ and the disjunctions (3) and (7); we have $\widehat{\mathcal{D}}_2 = \widehat{\mathcal{D}}_1 \cup \{d'\}$, where d' is

$$\{t_2, t_5\} \tag{9}$$

Consider now the edge $\{t_4, t_5\}$ and the disjunctions (5) and (8); we have $\widehat{\mathcal{D}}_3 = \widehat{\mathcal{D}}_2 \cup \{d''\}$, where d'' is

$$\{t_1, t_2, t_3\} \tag{10}$$

Consider now the edge $\{t_3, t_4\}$ and the disjunctions (7) and (10); we have $\widehat{\mathcal{D}}_4 = \widehat{\mathcal{D}}_3 \cup \{d'''\}$, where d''' is

$$\{t_1, t_2, t_5\} \tag{11}$$

It can be verified that the sequence is maximal. Thus, the disjunctive database returned by Algorithm 1 is $reduction(\widehat{\mathcal{D}}_4)$, namely the disjunctive database containing the disjunctions (1), (2), (3), (7), (8), (9).

The first, second and third step of Algorithm 1 ensure that condition 1.a, 1.b and 2 of Proposition 1 hold, respectively. The third step can be viewed as an instance of "resolution":

$$\frac{\{t_1\} \cup d_1 \quad \ldots \quad \{t_k\} \cup d_k \quad \neg(t_1 \wedge t_2 \wedge \cdots \wedge t_k)}{d_1 \cup \cdots \cup d_k}$$

where $k \geq 2$, $|d_i| \geq 1 \wedge d_i \cap \{t_1, \ldots, t_k\} = \emptyset$ $(1 \leq i \leq k)$, and $\{t_1, \ldots, t_k\}$ is an edge of the conflict hypergraph. Moreover, observe that, since $\widehat{\mathcal{D}}_{i-1} \subsetneq \widehat{\mathcal{D}}_i$ $(1 \leq i \leq n)$, and the number of disjunctions is bounded (if the original database has h facts, there cannot be more than $2^h - 1$ disjunctions), the sequence if finite. 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 DB and a set F of denial constraints, the set of minimal models of $\mathcal{D}(DB, F)$ is equal to the set of repairs of DB w.r.t. F.*

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

(1) Consider a repair R in $repairs(DB, F)$. First we show that (a) R is a model of $\widehat{\mathcal{D}}_n$ and next (b) that it is a minimal model.

(a) We show by induction on increasing i that for each $0 \leq i \leq n$, for each $d \in \widehat{\mathcal{D}}_i$, $R \cap d \neq \emptyset$.
Basis $i = 0$. Suppose by contradiction that there exists a disjunction $\{t\} \cup \{t_1\} \cup \cdots \cup \{t_m\}$ in $\widehat{\mathcal{D}}_0$, where $edges_{DB',F}(t) = \{e_1, \ldots, e_m\}$ and $t_i \in e_i - \{t\}$ for $i = 1..m$, such that $R \cap (\{t\} \cup \{t_1\} \cup \cdots \cup \{t_m\}) = \emptyset$. Observe that $edges_{DB',F}(t) = edges_{DB,F}(t)$. Since in each edge in $edges_{DB,F}(t)$ there is a fact (different from t) which is not in R, then $R \cup \{t\}$ is consistent, which violates the maximality of R.
Step $i - 1 \rightarrow i$. Consider a disjunction $d \in \widehat{\mathcal{D}}_i$. If $d \in \widehat{\mathcal{D}}_{i-1}$, then $R \cap d \neq \emptyset$ by the induction hypothesis. If $d \notin \widehat{\mathcal{D}}_{i-1}$, then there exist $k \geq 2$ facts $t_1, \ldots, t_k \in DB'$ and disjunctions $d_1, \ldots, d_k \in \widehat{\mathcal{D}}_{i-1}$ such that:

(i) $\{t_1, \ldots, t_k\}$ is an edge of $\mathcal{G}_{DB',F}$;
(ii) for each $j \in [1..k]$, $|d_j| \geq 2$ and $d_j \cap \{t_1, \ldots, t_k\} = \{t_j\}$; and
(iii) $d = (d_1 \cup \cdots \cup d_k) - \{t_1, \ldots, t_k\}$.

By the induction hypothesis, $R \cap d_j \neq \emptyset$ for $j = 1..k$. Hence, $R \cap d \neq \emptyset$, otherwise it would be the case that $\{t_1, \ldots, t_k\} \subseteq R$, which is a contradiction.

(b) We now show that R is a minimal model, reasoning by contradiction. Assume that there exists a model $M' \subsetneq R$ 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 $\widehat{\mathcal{D}}_n$ which does not contain a non-conflicting fact because $\widehat{\mathcal{D}}_n$ contains a singleton disjunction $\{t'\}$ for each non-conflicting fact t'). Moreover, as R is a repair, t is s.t. $\{t\}$ is not an edge of $\mathcal{G}_{DB,F}$ and then t is in DB'. For each edge e_i in $edges_{DB',F}(t) = \{e_1, \ldots, e_m\}$ there is a fact $t_i \in e_i - \{t\}$ which is not in R since R is consistent and $edges_{DB',F}(t) = edges_{DB,F}(t)$. The same holds for M' as it is a subset of R. Then, the disjunction $\{t\} \cup \{t_1\} \cup \cdots \cup \{t_m\}$, which is in $\widehat{\mathcal{D}}_0$ and thus in $\widehat{\mathcal{D}}_n$, is not satisfied by M', which contradicts that M' is a model. Hence R is a minimal model of $\widehat{\mathcal{D}}_n$.

(2) Consider a minimal model M in $\mathcal{MM}(\widehat{\mathcal{D}}_n)$. 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 $\widehat{\mathcal{D}}_n$ doesn't contain a singleton disjunction $\{t\}$ s.t. t is a conflicting fact of DB. This can be shown as follows. Two cases may occur: either $\{t\}$ is an edge of $\mathcal{G}_{DB,F}$ or it is not. As for the first case, since we have proved above that each repair of DB and F is a model of $\widehat{\mathcal{D}}_n$ and no repair contains t, it cannot be the case that $\{t\}$ is a singleton disjunction of $\widehat{\mathcal{D}}_n$. Let us consider the second case. For any conflicting fact t in DB s.t. $\{t\}$ is not an edge of $\mathcal{G}_{DB,F}$, 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 $\widehat{\mathcal{D}}_n$ corresponding to R_1 and R_2, then it cannot be the case that $\{t\} \in \widehat{\mathcal{D}}_n$. 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 $\mathcal{G}_{DB,F}$. Let $S_{t_i} = \{d - \{t_i\} \mid d \in \widehat{\mathcal{D}}_n \text{ and } d \cap \{t_1, \ldots, t_k\} = \{t_i\}\}$ for $i = 1..k$. Two cases may occur: either (i) for some t_i, the set S_{t_i} is empty, or (ii) all the sets S_{t_i} are not empty. (i) 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. (ii) For each $d_1 \in S_{t_1}, \ldots, d_k \in S_{t_k}$, it holds that $d_1 \cup \cdots \cup d_k \in \widehat{\mathcal{D}}_n$ (note that this follows from the the definition of the algorithm and the fact that each d_i is not empty, the latter being true since for any conflicting fact t of DB there does not exist a singleton disjunction $\{t\}$ in $\widehat{\mathcal{D}}_n$). Thus, 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 $d_1 \cup \cdots \cup d_k$ in $\widehat{\mathcal{D}}_n$, 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 DB reasoning by contradiction, thus assuming that there exists $M' \supsetneq M$ which is consistent. Let t be a fact in M' but not in M. Once again, note that $edges_{DB',F}(t) = edges_{DB,F}(t)$. Since M' is consistent, for each edge e_i in $edges_{DB',F}(t) = \{e_1, \ldots, e_m\}$ there is a fact $t_i \in e_i - \{t\}$ which is not in M'. The same holds for M as it is a (proper) subset of M'. This implies that M does not satisfy the disjunction $\{t\} \cup \{t_1\} \cup \cdots \cup \{t_m\}$ in $\widehat{\mathcal{D}}_0$ and then in $\widehat{\mathcal{D}}_n$, thus contradicting the fact the M is a model. Hence, M is a maximal consistent subset of DB, that is a repair. □

Given a database DB containing n facts, a rough bound on the size of $\mathcal{D}(DB, F)$ 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 $\mathcal{D}(DB, F)$ for special classes of denial constraints, namely functional dependencies and key constraints).

The following theorem allows us to say when two disjunctive databases have the same minimal models.

Theorem 2 *Two disjunctive databases $\mathcal{D}_1$ and $\mathcal{D}_2$ have the same minimal models if and only if $reduction(\mathcal{D}_1) = reduction(\mathcal{D}_2)$.*

Proof

($\Leftarrow$) Trivial.

($\Rightarrow$) Suppose by contradiction that $\mathcal{D}_1$ and $\mathcal{D}_2$ have the same minimal models and $reduction(\mathcal{D}_1) \neq reduction(\mathcal{D}_2)$. Thus, $reduction(\mathcal{D}_1) - reduction(\mathcal{D}_2) \neq \emptyset$ or $reduction(\mathcal{D}_2) - reduction(\mathcal{D}_1) \neq \emptyset$ (or both). Suppose that the first case holds and let d_1 be a disjunction in $reduction(\mathcal{D}_1) - reduction(\mathcal{D}_2)$ (the reasoning below can be applied analogously to the second case). Two cases may occur: either (a) there exists $d_2 \in reduction(\mathcal{D}_2)$ which subsumes d_1, or (b) the previous condition does not hold.

(a) Let I be the interpretation $S - d_2$ where S is the set of facts appearing in $reduction(\mathcal{D}_1)$. It is easy to see that I is a model of $reduction(\mathcal{D}_1)$ (the only disjunctions that I could not satisfy are those ones that contain only facts in d_2; such disjunctions are not in $reduction(\mathcal{D}_1)$ as they subsume d_1 and $reduction(\mathcal{D}_1)$ does not contain two disjunctions s.t. one subsumes the other). Thus, there exists $M \subseteq I$ which is a minimal model of $reduction(\mathcal{D}_1)$. As $d_2 \in reduction(\mathcal{D}_2)$, each model of $reduction(\mathcal{D}_2)$ contains a fact in d_2, then M is not a minimal model of $reduction(\mathcal{D}_2)$. It follows that M is a minimal model of $\mathcal{D}_1$ and is not a minimal model of $\mathcal{D}_2$, which is a contradiction.

(b) The reasoning is similar to case (a). Specifically, let I be the interpretation $S - d_1$ where S is the set of facts appearing in $reduction(\mathcal{D}_2)$. It is easy to see that I is a model of $reduction(\mathcal{D}_2)$ (the only disjunctions that I could not satisfy are those ones which contain only facts in d_1; such disjunctions are not in $reduction(\mathcal{D}_2)$ as $reduction(\mathcal{D}_2)$ contains neither d_1 nor a disjunction which subsumes d_1). Thus, there exists $M \subseteq I$ which is a minimal model of $reduction(\mathcal{D}_2)$. As $d_1 \in reduction(\mathcal{D}_1)$, each model of $reduction(\mathcal{D}_1)$ contains a fact in d_1, then M is not a minimal model of $reduction(\mathcal{D}_1)$. It follows that M is a minimal model of $\mathcal{D}_2$ and is not a minimal model of $\mathcal{D}_1$, which is a contradiction. □

Corollary 1 *Given a database DB and a set F of denial constraints, there exists a unique disjunctive database, henceforth called* canonical *and denoted by $\mathcal{D}_{min}(DB, F)$, s.t. (i) the minimal models of $\mathcal{D}_{min}(DB, F)$ are the repairs for DB and F, and (ii) $\mathcal{D}_{min}(DB, F)$ is contained in any other disjunctive database with the same set of minimal models.*

Proof Straightforward from Theorem 2. □

Whenever DB and F are clear from the context, we simply write $\mathcal{D}_{min}$ instead of $\mathcal{D}_{min}(DB, F)$.

Example 4 Consider the relation r below

emp

Name	*Dept*
john	*cs*
john	*math*
john	*physics*

and the following denial constraint stating that each employee may have at most two different departments:

$$\forall X, Y_1, Y_2, Y_3 \neg \big[emp(X, Y_1) \wedge emp(X, Y_2) \wedge emp(X, Y_3) \\ \wedge Y_1 \neq Y_2 \wedge Y_2 \neq Y_3 \wedge Y_1 \neq Y_3\big]$$

Clearly, the relation above is inconsistent. There are three repairs which are obtained from the original relation by deleting exactly one tuple. Let t_1, t_2, t_3 be the facts corresponding to the tuples in r. In this case, $\mathcal{D}_{min}$ is as follows:

$$\{t_1, t_2\}$$
$$\{t_2, t_3\}$$
$$\{t_1, t_3\}$$

It can be easily verified that the minimal models of $\mathcal{D}_{min}$ are the repairs of r w.r.t. the denial constraint above. Moreover, note that $\mathcal{D}_{min}$ is equal to its reduction and then Theorem 2 entails that any other disjunctive database $\mathcal{D}$ with the same set of minimal models is s.t. $reduction(\mathcal{D}) = \mathcal{D}_{min}$, that is $\mathcal{D}$ is a superset of $\mathcal{D}_{min}$ containing, in addition, disjunctions which are subsumed by disjunctions in $\mathcal{D}_{min}$.

As stated by the following corollary, Algorithm 1 computes the canonical disjunctive database.

Corollary 2 *Given a database DB and a set F of denial constraints, then $\mathcal{D}(DB, F) = \mathcal{D}_{min}(DB, F)$.*

Proof Straightforward from Theorems 1 and 2. □

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 (in the sense that it follows a pattern) 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 has an irregular structure and it is harder to identify a pattern for $\mathcal{D}_{min}$.

The size of a disjunction d, denoted by $||d||$, is equal to $|d|$. The size of a disjunctive database $\mathcal{D}$, denoted as $||\mathcal{D}||$, is the sum of the size of the disjunctions 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 $cliques(r, k)$ 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 2 *Given a relation r and a key constraint k, then $\mathcal{D}_{min}$ is equal to*

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

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. □

Example 5 Consider the relation of Example 2. There are n cliques $C_i = \{t'_i, t''_i\}$, $1 \leq i \leq n$. Then, $\mathcal{D}_{min} = \{\{t'_i, t''_i\} \mid 1 \leq i \leq n\}$.

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

Proposition 3 *Given a relation and a key constraint, $\mathcal{D}_{min}$ is computed in polynomial time by Algorithm* 1.

Proof It can be easily verified that $\mathcal{D}_{min} = \widehat{\mathcal{D}}$ and that $\widehat{\mathcal{D}}$ is computed in polynomial time. □

Two keys. We now show that, in the presence of two key constraints, $\mathcal{D}_{min}$ may have exponential size. Let DB_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

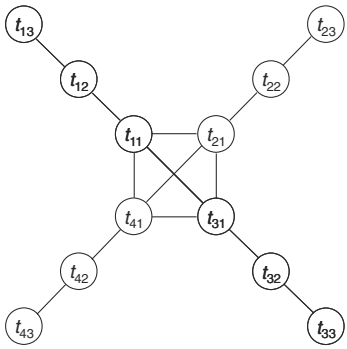

Fig. 1 Conflict hypergraph for a database in DB_4 w.r.t. A, B key constraints

Let $DB \in DB_n$ and A, B be two keys. The conflict hypergraph for DB 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 6 The conflict hypergraph for a database in DB_4, assuming that A and B are two keys, is reported in Fig. 1.

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

Proposition 4 *Consider a database DB in DB_n and a set of constraints F consisting of two keys, A and B. Then $\mathcal{D}_{min}$ is equal to $\mathcal{D}$ where*

$$\mathcal{D} = \{\{t_{i2}, t_{i3}\} \mid 1 \leq i \leq n\} \ \cup \{\{t_{i1}, t_{i2}\} \cup \bigcup_{j=1..n \ \wedge \ j \neq i} \{t_{jz_j}\} \mid 1 \leq i \leq n \ \wedge z_j \in \{1, 3\}\}$$

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

(1) Consider a minimal model $M \in \mathcal{MM}(DB)$. 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 $\mathcal{G}_{DB,F}$. First we show that for each $e = \{t', t''\}$ in E and pair of disjunctions d', d'' in $\mathcal{D}$ s.t. $d' \cap \{t', t''\} = \{t'\}$ and $d'' \cap \{t', t''\} = \{t''\}$, there is a disjunction in $\mathcal{D}$ which is equal to or subsumes $(d' \cup d'') - \{t', t''\}$; next we use this property to show 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

$$d'_1 \ : \ \{t_{i1}, t_{i2}, t_{j3}\} \cup \bigcup_{z=1..n \ \wedge \ z \neq i,\, j} \{t'_{zk_z}\}$$

where $k_z \in \{1, 3\}$, or of the form

$$d'_2 \ : \ \{t_{h1}, t_{h2}, t_{i1}, t_{j3}\} \cup \bigcup_{z=1..n \ \wedge \ z \neq h,i,\, j} \{t'_{zk_z}\}$$

where $1 \leq h \leq n \ \wedge \ h \neq i, j$ and $k_z \in \{1, 3\}$. Likewise, a disjunction in $\mathcal{D}$ that contains t_{j1} but not t_{i1} is of the form

$$d''_1 \ : \ \{t_{j1}, t_{j2}, t_{i3}\} \cup \bigcup_{z=1..n \ \wedge \ z \neq i,\, j} \{t''_{zk_z}\}$$

where $k_z \in \{1, 3\}$, or of the form

$$d''_2 \ : \ \{t_{k1}, t_{k2}, t_{j1}, t_{i3}\} \cup \bigcup_{z=1..n \ \wedge \ z \neq k,i,\, j} \{t''_{zk_z}\}$$

where $1 \leq k \leq n \ \wedge \ k \neq i, j$ and $k_z \in \{1, 3\}$. In all the four possible cases, there is disjunction in $\mathcal{D}$ which subsumes $(d' \cup d'') - \{t', t''\}$:

- if $d' = d'_1$ and $d'' = d''_1$, then there are both $\{t_{j2}, t_{j3}\}$ and $\{t_{i2}, t_{i3}\}$ in $\mathcal{D}$;
- if $d' = d'_1$ and $d'' = d''_2$, then we have $\{t_{i2}, t_{i3}\}$ in $\mathcal{D}$;
- if $d' = d'_2$ and $d'' = d''_1$, then we have $\{t_{j2}, t_{j3}\}$ in $\mathcal{D}$;
- if $d' = d'_2$ and $d'' = d''_2$, there are both $\{t_{h1}, t_{h2}, t_{i3}, t_{j3}\} \cup \bigcup_{z=1..n \ \wedge \ z \neq h,i,\, j} \{t'_{zk_z}\}$ and $\{t_{k1}, t_{k2}, t_{i3}, t_{j3}\} \cup \bigcup_{z=1..n \ \wedge \ z \neq k,i,\, j} \{t''_{zk_z}\}$ in $\mathcal{D}$.

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

$$\{t_{k1}, t_{k2}, t_{i1}\} \cup \bigcup_{z=1..n \ \wedge \ z \neq i,k} \{t_{zk_z}\}$$

where $1 \leq k \leq n \ \wedge \ k \neq i$ and $k_z \in \{1, 3\}$, whereas a disjunction containing t_{i2} but not t_{i1} is of the form $\{t_{i2}, t_{i3}\}$. Thus, $(d' \cup d'') - \{t', t''\}$, which is equal to

$$\{t_{k1}, t_{k2}, t_{i3}\} \cup \bigcup_{z=1..n \ \wedge \ z \neq i,k} \{t_{zk_z}\}$$

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

$$\{t_{i1}, t_{i2}\} \cup \bigcup_{z=1..n \ \wedge \ z \neq i} \{t'_{zk_z}\}$$

where $k_z \in \{1, 3\}$, whereas a disjunction containing t_{i3} but not t_{i2} is of the form

$$\{t_{h1}, t_{h2}, t_{i3}\} \cup \bigcup_{z=1..n \wedge z \neq h,i} \{t''_{zk_z}\}$$

where $1 \leq h \leq n \wedge h \neq i$ and $k_z \in \{1, 3\}$; $(d' \cup d'') - \{t', t''\}$ is subsumed or equal to the disjunction

$$\{t_{h1}, t_{h2}, t_{i1}\} \cup \bigcup_{z=1..n \wedge z \neq h,i} \{t''_{zk_z}\}$$

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 $S_{t_a} = \{d - \{t_a\} \mid d \in \mathcal{D} \text{ and } d \cap \{t_a, t_b\} = \{t_a\}\}$ and $S_{t_b} = \{d - \{t_b\} \mid d \in \mathcal{D} \text{ and } d \cap \{t_a, t_b\} = \{t_b\}\}$. 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 $d_a \cup d_b$. 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 $d_a \cup d_b$ 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 DB. First of all, we note that for each fact $t \in DB$ there is a disjunction $\{t, t_1, \ldots, t_n\}$ $(n \geq 1)$ 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 $\{t_{i2}, t_{i3}\}$;
- for the facts t_{i1} $(i = 1..n)$ such disjunctions are $\{t_{i1}, t_{i2}\} \cup \bigcup_{z=1..n \wedge z \neq i}\{t_{z1}\}$.

Assume by contradiction that M is not a maximal (consistent) subset of DB. Then there exists $M' \supsetneq M$ 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 $\{t, t_1, \ldots, t_n\}$ containing t and some fact conflicting with it: the fact that M is a model is contradicted.

(2) Consider a repair R for DB 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 $\{t_{i2}, t_{i3}\}$ $(1 \leq i \leq n)$ or $\{t_{i1}, t_{i2}\} \cup \bigcup_{z=1..n \wedge z \neq i}\{t_{zk_z}\}$, $1 \leq i \leq n$ and $k_z \in \{1, 3\}$. 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 $T_3 = \{t_{j3} \mid t_{j3} \in d\}$. 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 $t_{j3} \in T_3$, R does not contain t_{j1} since it contains t_{j2} and 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 $M' \subsetneq R$ 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 $t' \in DB$ there is a disjunction in $\mathcal{D}$ containing t' and only facts conflicting with t'; then there is a disjunction $d = \{t, t_1, \ldots, t_n\}$ 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 DB w.r.t. F. Since $\mathcal{D} = reduction(\mathcal{D})$, from Theorem 2 we have that $\mathcal{D}$ is the canonical disjunctive database whose minimal models are the repairs of DB w.r.t. F. □

Corollary 3 *Consider a database DB in DB_n and let A and B be two keys; $||\mathcal{D}_{min}|| = 2n + (n+1) \cdot n2^{n-1}$.*

Proof From Proposition 4, it is easy to see that $\mathcal{D}_{min}$ contains n disjunctions of 2 facts and $n2^{n-1}$ disjunctions of $n+1$ facts. □

The following lemma identifies the repairs of a database in DB_n w.r.t. a set of integrity constraints consisting of two keys, A and B. Such a lemma will be used in the next section (see Corollary 4).

Lemma 1 *Consider a database DB in DB_n and a set of integrity constraints F consisting of two keys, A and B. Then, the set of repairs is equal to $\mathcal{R}$ where*

$$\mathcal{R} = \{\{t_{12}, \ldots, t_{n2}\}\} \cup \{\{t_{i1}, t_{i3}\} \cup \bigcup_{j=1..n \ \wedge \ j \neq i} \{t_{jz_j}\} \mid 1 \leq i \leq n \ \wedge \ z_j \in \{2, 3\}\}$$

Proof It is easy to see that each database in $\mathcal{R}$ is a repair.

Consider a repair R of DB w.r.t. F. We show that R is in $\mathcal{R}$ 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 $R = \{t_{11}, t_{13}, t_{2z_2}, \ldots, t_{nz_n}\}$ where $z_j \in \{2, 3\}$, $j = 2..n$. It is easy to see that $R \in \mathcal{R}$.
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 $R = \{t_{13}, t_{k1}, t_{k3}\} \cup \bigcup_{j=2..n \ \wedge \ j \neq k} \{t_{jz_j}\}$ where $z_j \in \{2, 3\}$. Clearly, $R \in \mathcal{R}$.

– 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 $R = \{t_{12}, t_{k1}, t_{k3}\} \cup \bigcup_{j=2..n \ \wedge \ j \neq k} \{t_{jz_j}\}$ where $z_j \in \{2, 3\}$. It is easy to see that $R \in \mathcal{R}$.
  2. $R = \{t_{12}, \ldots, t_{n2}\}$ which is in $\mathcal{R}$. □

One functional dependency. Given a relation r and a functional dependency $f : X \rightarrow Y$, we denote by $cliques(r, f)$ 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 $\{G_1 \cup \cdots \cup G_n \mid G_i \in clusters(C_i) \text{ for } i = 1..n\}$.

Proposition 5 *Given a relation r and a functional dependency f, then $\mathcal{D}_{min}$ is equal to $\mathcal{D}$ where*

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

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 $\{t_1, \ldots, t_k\}$ 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 $\overline{G}_1 \cup \cdots \cup \overline{G}_n \subseteq M$ and $\overline{G}_1 \cup \cdots \cup \overline{G}_n$ is a model of $\mathcal{D}$, then $M = \overline{G}_1 \cup \cdots \cup \overline{G}_n$, which is, as we have observed before, a repair.

(2) Consider a repair R in $repairs(r, f)$. As R consists of one cluster for each clique, it is easy to see that R is a model of $\mathcal{D}$. We show that R is minimal by contradiction assuming that there exists $R' \subsetneq R$ which is a model of $\mathcal{D}$. Let t be a fact in R which is not in R'. 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 $\{t, t_1, \ldots, t_k\}$, where $t_i \in G_i, \ i = 1..k$, which is in $\mathcal{D}$, is not satisfied by R' as R' 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 R' is a model. So R 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 7 Consider the relation r, consisting of $2n$ facts, reported below and the non-key functional dependency $A \rightarrow B$.

	A	B	C
t'_1	a	b_1	c_1
t''_1	a	b_1	c_2
$\vdots$	$\vdots$	$\vdots$	$\vdots$
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 $\mathcal{D}_{min} = \{\{t_1, \ldots, t_n\} \mid t_i \in G_i \text{ for } i = 1..n\}$ 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 subset-based repairs, i.e., consistent databases for which the symmetric difference is minimal under set inclusion).

It is worth noting that also when cardinality-based repairs are considered, a canonical disjunctive database exists. In fact, a disjunctive database $\mathcal{D}$ representing the cardinality-based repairs of a database DB w.r.t. a set F of denial constraints might be naively computed by rewriting the following DNF formula in CNF:

$$\bigvee_{R \in repairs^c(DB,F)} \bigwedge_{t \in R} t$$

where $repairs^c(DB, F)$ is the set of cardinality-based repairs for DB and F. Theorem 2 allows us to say that $reduction(\mathcal{D})$ is contained in any other disjunctive database with the same set of minimal models. We will denote by $\mathcal{D}^c_{min}$ the canonical disjunctive database representing the cardinality-based repairs.

We show that, likewise to what has been presented in Section 4, the size of $\mathcal{D}^c_{min}$ 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 subset-based 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 subset-based repairs. This is the case for the database of Example 7, 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.

Corollary 4 *Consider a database DB in DB_n and a set of integrity constraints F consisting of two keys, A and B. Then the set of cardinality-based repairs is*

$$\{\{t_{i1}, t_{i3}\} \cup \bigcup_{j=1..n \wedge j \neq i} \{t_{jz_j}\} \mid 1 \leq i \leq n \wedge z_j \in \{2, 3\}\}$$

Proof Since cardinality-based repairs are subset-based repairs with maximum cardinality, the claim is straightforwardly entailed by Lemma 1. □

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

Proposition 6 *Consider a database DB in DB_n and a set of integrity constraints F consisting of two keys, A and B. Then, $\mathcal{D}^c_{min}$ is equal to $\mathcal{D}$ where*

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

Proof We first show that (1) each cardinality-based repair of DB 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 DB 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 4, it is easy to see that R satisfies each disjunction $\{t_{i2}, t_{i3}\}$ in $\mathcal{D}$, $1 \leq i \leq n$. Since Corollary 4 entails that there exists $1 \leq j \leq n$ s.t. $\{t_{j1}, t_{j3}\} \subseteq R$, then R satisfies each disjunction $\{t_1, \ldots, t_n\}$ 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 $t \in DB$ there is a disjunction $\{t, t_1, \ldots, t_n\}$ ($n \geq 1$) 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 $\{t_{i2}, t_{i3}\}$; for the facts t_{i1} ($i = 1..n$) there is the disjunction $\{t_{11}, \ldots, t_{n1}\}$. In the same way as in Proposition 4, 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 subset-based repair of DB w.r.t. F can be shown in the same way as in Proposition 4.
   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 4, M is a cardinality-based repair of DB w.r.t. F.

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

Corollary 5 *Consider a database DB in DB_n and let A and B be two keys; $||\mathcal{D}^c_{min}|| = 2n + n2^n$.*

Proof From Proposition 6, it is easy to see that $\mathcal{D}^c_{min}$ 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.

A disjunctive database representing a set of repairs might be exploited for computing consistent query answers using existing efficient disjunctive logic programming systems, such as DLV [17]. Indeed, every Relational Algebra query can be expressed by means of stratified Datalog program, which can be combined with a disjunctive database representing the repairs, directly providing the consistent answers under cautious reasoning. Moreover, since a disjunction is true in every minimal model of a disjunctive database $\mathcal{D}$ (or more generally, of a disjunctive logic program) iff it is true in every model of $\mathcal{D}$ [19], then a propositional theorem prover might be used as well to compute the consistent answers to negation-free queries.

One could potentially *restrict* inconsistent databases in such a way that the resulting repairs can be succinctly represented by relational databases with *OR-objects* [15]. Patterns of OR-objects leading to tractable conjunctive queries were characterized in [16].

It could be advantageous to precompute a disjunctive specification of all repairs and use it multiple times, perhaps even for different tasks. For instance, consider the information coming from a set of sensors. It is often inconsistent, so it needs to be repaired. But we may be interested in not throwing away any repairs and keeping them all as a disjunctive database, so further processing on it (diagnosis etc.) can be done using a single DLP system like DLV.

Approaches that rely on stable model semantics of *Disjunctive Logic Programs with negation* (DLP) to represent repairs of inconsistent databases have been proposed in [4, 10, 14]; however, they do not provide a study of the size of such representations. Also, those approaches are based on more complex semantics than minimal-model semantics. As future work, one could do an experimental comparison of CQA computation using a fully-DLP approach with one in which the repairs are represented as disjunctive databases (computed using the proposed algorithm) and only queries are in a DLP format.

Other 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. On the other hand, more general sets of integrity constraints could be considered.

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 [6], leading to a host of classical database research issues like query optimization and evaluation.

Finally, other kinds of representations of sets of possible worlds, e.g., *world-set decompositions* [2], should be considered. For example, the set of repairs of the database in Example 7 can be represented as a world-set decomposition of polynomial size.

Acknowledgements The authors thank the referees for their extensive comments. 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. Afrati, F.N., Kolaitis, P.G.: Repair checking in inconsistent databases: algorithms and complexity. In: International Conference on Database Theory (ICDT), pp. 31–41 (2009)
2. Antova, L., Koch, C., Olteanu, D.: 10^{10^6} worlds and beyond: efficient representation and processing of incomplete information. In: International Conference on Data Engineering (ICDE), pp. 606–615 (2007)
3. Arenas, M., Bertossi, L.E., Chomicki, J.: Consistent query answers in inconsistent databases. In: ACM Symposium on Principles of Database Systems (PODS), pp. 68–79 (1999)
4. Arenas, M., Bertossi, L.E., Chomicki, J.: Answer sets for consistent query answering in inconsistent databases. Theory Pract. Log. Program **3**(4–5), 393–424 (2003)
5. Arieli, O., Denecker, M., Bruynooghe, M.: Distance semantics for database repair. Ann. Math. Artif. Intell. **50**(3–4), 389–415 (2007)
6. Benjelloun, O., Sarma, A.D., Halevy, A.Y., Theobald, M., Widom, J.: Databases with uncertainty and lineage. VLDB J. **17**(2), 243–264 (2008)
7. Bertossi, L.E.: Consistent query answering in databases. SIGMOD Rec. **35**(2), 68–76 (2006)
8. Bertossi, L.E., Bravo, L., Franconi, E., Lopatenko, A.: The complexity and approximation of fixing numerical attributes in databases under integrity constraints. Inf. Syst. **33**(4–5), 407–434 (2008)
9. Bertossi, L.E., Chomicki, J.: Query answering in inconsistent databases. In: Logics for Emerging Applications of Databases, pp. 43–83 (2003)
10. Calì, A., Lembo, D., Rosati, R.: Query rewriting and answering under constraints in data integration systems. In: International Joint Conference on Artificial Intelligence (IJCAI), pp. 16–21 (2003)
11. Chomicki, J.: Consistent query answering: five easy pieces. In: International Conference on Database Theory (ICDT), pp. 1–17 (2007)
12. Chomicki, J., Marcinkowski, J.: Minimal-change integrity maintenance using tuple deletions. Inf. Comput. **197**(1–2), 90–121 (2005)
13. Fernández, J.A., Minker, J.: Semantics of disjunctive deductive databases. In: International Conference on Database Theory (ICDT), pp. 21–50 (1992)
14. Greco, G., Greco, S., Zumpano, E.: A logical framework for querying and repairing inconsistent databases. IEEE Trans. Knowl. Data Eng. **15**(6), 1389–1408 (2003)
15. Imielinski, T., Naqvi, S.A., Vadaparty, K.V.: Incomplete objects—a data model for design and planning applications. In: ACM SIGMOD Conference, pp. 288–297 (1991)
16. Imielinski, T., van der Meyden, R., Vadaparty, K.V.: Complexity tailored design: a new design methodology for databases with incomplete information. J. Comput. Syst. Sci. **51**(3), 405–432 (1995)
17. Leone, N., Pfeifer, G., Faber, W., Eiter, T., Gottlob, G., Perri, S., Scarcello, F.: The dlv system for knowledge representation and reasoning. ACM Trans. Comput. Logic **7**(3), 499–562 (2006)
18. Lopatenko, A., Bertossi, L.E.: Complexity of consistent query answering in databases under cardinality-based and incremental repair semantics. In: International Conference on Database Theory (ICDT), pp. 179–193 (2007)

19. Minker, J.: On indefinite databases and the closed world assumption. In: International Conference on Automated Deduction (CADE), pp. 292–308 (1982)
20. Minker, J., Seipel, D.: Disjunctive logic programming: a survey and assessment. In: Computational Logic: Logic Programming and Beyond, pp. 472–511 (2002)
21. Vardi, M.Y.: The complexity of relational query languages (extended abstract). In: ACM Symposium on Theory of Computing (STOC), pp. 137–146 (1982)