On Estimating the Cardinality of Aggregate Views *

Paolo Ciaccia Matteo Golfarelli Stefano Rizzi

DEIS - University of Bologna
40136 Bologna, Italy
{pciaccia, mgolfarelli, srizzi}@deis.unibo.it

Abstract

Accurately estimating the cardinality of aggregate views is crucial for logical and physical design of data warehouses. While the warehouse is under development and data are not available yet, the approaches based on accessing data cannot be adopted. This paper proposes an approach to estimate the cardinality of views based on a-priori information derived from the application domain. We face the problem by first computing satisfactory bounds for the cardinality, then by capitalizing on these bounds to determine a good probabilistic estimate for it. Bounds are determined by using, besides the functional dependencies expressed by the multidimensional scheme, additional domain-derived information in the form of cardinality constraints which may bound either the cardinality of a given view or the ratio between the cardinalities of two given views. In particular, we propose a bounding strategy which achieves an effective trade-off between the tightness of the bounds produced and the computational complexity.

1 Introduction and Motivation

The multidimensional model is the foundation for data representation and querying in multidimensional databases and data warehouses [AGS97]. It represents facts of interest for the decision process into *cubes* in which each cell contains numerical *measures* which quantify the fact from different points of view, while each axis represents an interesting *dimension* for analysis. For instance, within a 4-dimensional cube modeling the phone calls supported by a telecommunication company, the dimensions might be the calling number, the number called, the date, and the time segment in which the call is placed; each cube cell could be associated to a measure of the total duration of the calls made from a given number to another number on a given time segment and date.

The basic mechanism to extract significant information from the huge quantity of fine-grained data stored in base cubes is aggregation according to hierarchies of attributes rooted in dimensions [GL97]. In most application cases, cubes are significantly sparse (for instance, most couples of telephone numbers are never connected by a call in a given date), and so are the aggregate views.

Accurately estimating the actual cardinality of each view is crucial for logical and physical design as well as for query processing and optimization [Vas00]. As a relevant case, consider the view materialization problem, where the aggregate views which are the most useful in answering the workload queries have to be selected for materialization (see [TB00] for a survey). Since the number of possible views which can be derived by aggregating a cube is exponential in the number of attributes, most approaches assume that a constraint on the total disk space occupied by materialization is posed, and attempt to find the subset of views which contemporarily satisfies this constraint and minimizes the workload cost [GR00, Gup97, HRU96]. Another case where estimation of view cardinalities is relevant is index selection [GHRU97].

If the data warehouse has already been loaded, view cardinalities can be quite accurately estimated by using statistical techniques based, say, on histograms [MD88] or sampling [HO91]. However, such techniques cannot be applied at all if the data warehouse is still under development, and the estimation of view cardinalities is needed for

* This work has been partially supported by the D2I MURST project.

Proceedings of the International Workshop on Design and Management of Data Warehouses (DMDW'2001)
Interlaken, Switzerland, June 4, 2001
(D. Theodoratos, J. Hammer, M. Jeusfeld, M. Staudt, eds.)
http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-39/

design purposes. To obviate this, current approaches are based on estimation models that only exploit the cardinality of the base cube and that of the single attribute domains [RS97, SDNR96], which however leads to significant overestimation.

In this paper we propose a novel approach to estimate the cardinality of views based on a-priori information derived from the application domain. Similarly to what is done when estimating the cardinality of projections in relational databases [CM95], we face the problem by first computing satisfactory bounds for the cardinality, then by capitalizing on these bounds to determine a good probabilistic estimate for it. Besides the functional dependencies expressed by the multidimensional scheme, the bounds we determine also take into account additional domain-derived information expressed in the form of *cardinality constraints*, namely, bounds of the cardinality of some views and bounds (called *k-dependencies*) on the ratio between the cardinalities of two views. The computation of bounds is based on a *bounding strategy*, which is aimed at achieving an effective trade-off between the tightness of the bounds produced and the computational complexity.

The paper is organized as follows. After providing some basic definitions in Section 2, in Section 3 we introduce k-dependencies. In Section 4 we outline our overall approach to estimation and show its benefits with an example. Section 5 introduces the basic properties of bounds, proposes an efficient bounding strategy, and sketches a branch-and-bound approach to determine the upper bound of the cardinality of a given view when the cardinality constraints in input do not contain k-dependencies; besides, it discusses how the strategy introduced can be improved. Section 6 shows how the bounds derived may be used to improve the cardinality estimates. Finally, Section 7 discusses the most interesting open issues.

2 Background and Working Example

In this section we formalize the concept of view, define a partial ordering on the set of views, and present the application domain we will use as an example.

Definition 1 (Dimensional Scheme) *We call* dimensional scheme $\mathcal{D}$ *a couple* $(U, \mathcal{F})$ *where* U *is a set of attributes and* $\mathcal{F} = \{A_i \rightarrow A_j \mid A_i, A_j \in U\}$ *is a set of functional dependencies (FD's) which relate the attributes of* U *into a set of pairwise disjoint directed trees. We call* dimensions *the attributes* $A_k \in U$ *in which the trees are rooted, i.e., such that* $\forall A_i \in U\ (A_i \rightarrow A_k) \notin \mathcal{F}$*; let* $dim(\mathcal{D}) \subseteq U$ *denote the set of dimensions of* $\mathcal{D}$.

Definition 2 (View) *Let* $\mathcal{D} = (U, \mathcal{F})$ *be a dimensional scheme. We call* view *on* $\mathcal{D}$ *any subset of attributes* $V \subseteq U$ *such that* $\forall A_i, A_j \in V\ (A_i \rightarrow A_j) \notin \mathcal{F}^+$*, where* $\mathcal{F}^+$ *denotes the set of all functional dependencies logically implied by* $\mathcal{F}$.

It should be noted that we are using the term *view* to denote the set of grouping attributes used for aggregation, while the "actual" views will typically include also one or more measures. This slight abuse in terminology is possible since we are interested in determining the *cardinality* of views, which only depends on the grouping attributes.

Definition 3 (Roll-up) *Given the set* $\mathcal{V}_\mathcal{D}$ *of all possible views on* $\mathcal{D}$*, we define on* $\mathcal{V}_\mathcal{D}$ *the* roll-up *partial order* $\preceq$ *as follows:* $V \preceq W$ *iff* $\forall A_i \in V\ \exists A_j \in W \mid (A_j \rightarrow A_i) \in \mathcal{F}^+$*, i.e., iff* $W \rightarrow V$. *We call* multidimensional lattice *for* $\mathcal{D}$ *the corresponding lattice, whose top and bottom elements are* $dim(\mathcal{D})$ *and the empty view* $\{\}$*, respectively. We will denote with* $V \oplus W$ *the view that is the least upper bound of* V *and* W *in the lattice; given a set of views* S*, we will briefly denote with* $\oplus(S)$ *the view that is their least upper bound.*

Example 1 Consider an enterprise with branches in different cities. A simple dimensional scheme *Transfers* modeling the transfers of employees between offices might include:

$$U = \{\mathsf{date}, \mathsf{month}, \mathsf{year}, \mathsf{fromOffice}, \mathsf{fromDept}, \mathsf{fromCity}, \mathsf{toOffice}, \mathsf{toDept}, \mathsf{toCity}, \mathsf{employee}\}$$
$$\mathcal{F} = \{\mathsf{date} \rightarrow \mathsf{month}, \mathsf{month} \rightarrow \mathsf{year}, \mathsf{fromOffice} \rightarrow \mathsf{fromDept}, \mathsf{fromOffice} \rightarrow \mathsf{fromCity}, \mathsf{toOffice} \rightarrow \mathsf{toDept}, \mathsf{toOffice} \rightarrow \mathsf{toCity}\}$$

thus $dim(\mathcal{D}) = \{\mathsf{date}, \mathsf{fromOffice}, \mathsf{toOffice}, \mathsf{employee}\}$. Examples of views on the *Transfers* scheme are

$$V = \{\mathsf{month}, \mathsf{fromOffice}, \mathsf{toCity}, \mathsf{employee}\}$$
$$W = \{\mathsf{month}, \mathsf{fromCity}, \mathsf{fromDept}\}$$
$$Z = \{\mathsf{year}, \mathsf{fromOffice}, \mathsf{toCity}\}$$

It is $W \oplus Z = \{\mathsf{month}, \mathsf{fromOffice}, \mathsf{toCity}\}$, with $(W \oplus Z) \preceq V$. □

The following notation is used throughout the rest of the paper. Uppercase letters from the beginning of the alphabet (A, B, ...) denote dimensions. Attributes which are functionally determined by another attribute, i.e. attributes other than dimensions, are denoted by the corresponding primed letters (e.g., $A \rightarrow A'$, $A \rightarrow A''$). Sets of attributes are represented by omitting braces, thus writing ABC for $\{A, B, C\}$. V is the view whose cardinality is to be estimated, while W, X, Y, and Z, possibly with subscripts ($W_1, W_2, \ldots$), denote generic views in $\mathcal{V}_\mathcal{D}$. Finally, lowercase letters are used for the cardinalities of views and attributes (e.g., w is the cardinality of view W, abc is the cardinality of the view with attributes ABC, and so on).

3 The k-dependencies

A k-dependency is a relevant case of cardinality constraint which naturally generalizes a functional dependency. In the

authors' experience, k-dependencies are particularly useful to characterize the knowledge of the business domain held by the experts in the field. For instance, in the transfer domain, we might have some information concerning the number of destination cities for an employee, or on the number of distinct areas moved to from each area. If such information is in the form of bounds, it can be effectively used to improve the bounds of view cardinality.

Definition 4 (k-dependency) *Let X and Y be two views on $\mathcal{D}$. We say that a* k-dependency *(kD) holds between X and Y, and denote it with $X \xrightarrow{k} Y$, when k ($k \geq 1$) is an upper bound of the number of distinct tuples of Y which correspond to each distinct tuple of X within view $X \oplus Y$.*

Example 2 In the *Transfers* scheme, assume the domain expert provides the following information: *The maximum number of inter-department transfers of an employee during one year is 2.* This constraint can be formalized by the following kD: $X \xrightarrow{2} Y$, where $X = \{\mathsf{year}, \mathsf{employee}\}$, $Y = \{\mathsf{toDept}\}$. Intuitively, from this we can derive that the cardinality of the view $\{\mathsf{year}, \mathsf{employee}, \mathsf{toDept}\}$ cannot exceed twice the cardinality of X. □

The kD's have been studied in the context of relational database theory, where they are also known as *numerical dependencies*. Grant and Minker [GM83] proved that kD's are not finitely axiomatizable, thus no fixed set of inference rules can be used to determine whether or not a given kD is logically implied by a set of kD's. Nonetheless, a basic set of rules, which naturally extend those for FD's, was proposed in [GM83]. The rules we use, generalized to the multidimensional lattice, are:

$$\text{R1}: \quad X \xrightarrow{k} Y \ \vdash\ X \oplus Z \xrightarrow{k} Y \oplus Z$$

$$\text{R2}: \quad X \xrightarrow{k} Y \wedge Y \xrightarrow{l} Z \ \vdash\ X \xrightarrow{k \cdot l} Y \oplus Z$$

$$\text{R3}: \quad X \xrightarrow{k} Y \oplus Z \ \vdash\ X \xrightarrow{k} Y$$

$$\text{R4}: \quad X \xrightarrow{k} Y \wedge X \xrightarrow{l} Z \ \vdash\ X \xrightarrow{k \cdot l} Y \oplus Z$$

Note that the "union" rule R4 is not strictly needed, since it can be derived from rules R1 ("extension"), R2 ("transitivity"), and R3 ("decomposition").

4 A Framework for Estimation

The framework for this work is the logical design of multidimensional databases carried out off-line, i.e. assuming that the source data cannot be directly queried to estimate the cardinality of multidimensional views. Without loss of generality, in the following we consider that estimates are needed for the purpose of view materialization, thus reliable information on the size of the candidate views has to be supplied to the materialization algorithm.

As sketched in Figure 1, whenever the materialization algorithm requires information about a candidate view V,

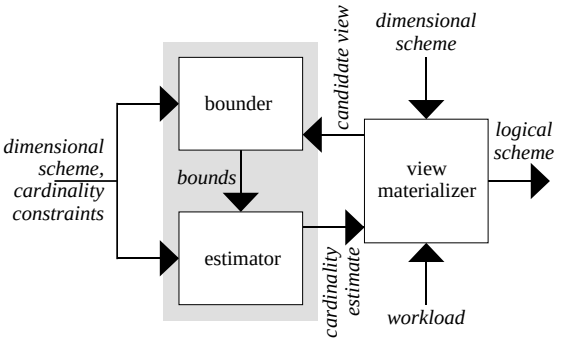

Figure 1: Overall architecture for logical design

our approach works in two steps. First, the *bounder* uses the set $\mathcal{I}$ of *cardinality constraints* supplied by the user to determine effective bounds for the cardinalities of a proper set of views; then, the *estimator* uses these bounds to derive a probabilistic estimate for the cardinality of V. Note that this two-steps approach generalizes well-known *parametric* models for the estimation of the cardinality of relational queries [MCS88], and in particular those for projection size estimation [CM95], for which bounds are typically given as input parameters.

The different forms of cardinality constraints we will consider are:

1. a lower (w^-) and/or an upper (w^+) bound of the cardinality w of a view W;
2. a k-dependency ($X \xrightarrow{k} Y$) expressing an upper bound of the ratio between the cardinalities of two views X and Y.

We will assume that at least the upper bounds of the cardinalities of all the single attributes in the dimensional scheme are known. This assumption, which is perfectly reasonable in all application domains, is necessary in order to guarantee that at least one upper bound can be determined for each view.

The set $\mathcal{I}$, together with the dimensional scheme $\mathcal{D}$, univocally determines two bounds for the cardinality of V, which are called the *greatest lower bound* and the *least upper bound*, denoted as v^- and v^+, respectively.[1] The interpretation of such bounds is as follows:

1. in each instance of $\mathcal{D}$ that does not violate any constraint in $\mathcal{I}$, the cardinality v of V is such that $v \in [v^-, v^+]$; and
2. there exist two instances, both compatible with $\mathcal{I}$, where v equals v^- and v^+, respectively.

We say a constraint $c \in \mathcal{I}$ is *redundant* iff all the greatest lower bounds and the least upper bounds determined by $\mathcal{I}$ are equal to those determined by $\mathcal{I} - \{c\}$.

[1] For simplicity of notation, in denoting bounds we omit the dependence on $\mathcal{D}$ and $\mathcal{I}$.

Definition 5 (Sound and Minimal Input) *Let $\mathcal{I}$ be a set of cardinality constraints on dimensional scheme $\mathcal{D}$. We say $\mathcal{I}$ is* sound *iff there exists at least one non-empty instance of $\mathcal{D}$ which satisfies all the constraints in $\mathcal{I}$. We say $\mathcal{I}$ is* minimal *iff no constraint in $\mathcal{I}$ is redundant.*

In this paper we will assume that the input $\mathcal{I}$ is sound and minimal. It is straightforward to derive that, in this case, all the bounds in $\mathcal{I}$ are either greatest lower bounds or least upper bounds (whereas the opposite is not necessarily true).

Computing the bounds implied by $\mathcal{I}$ turns out to be a challenging combinatorial problem, even for "simple" forms of cardinality constraints. For instance, it is known that the problem is NP-hard for arbitrary patterns of functional dependencies [CM92]. Furthermore, the actual computational effort needed to compute these bounds might limit applicability in real-world cases. For this reason, the bounder is built around the concept of *bounding strategy*. A bounding strategy s is characterized by a couple of bounding functions that, given $\mathcal{I}, \mathcal{D}$, and V, compute bounds v_{s}^- and v_{s}^+ such that $v_{\mathsf{s}}^- \leq v^-$ and $v^+ \leq v_{\mathsf{s}}^+$ both hold. In other terms, a bounding strategy never computes bounds which are more restrictive than the ones logically implied by the input constraints, trading-off accuracy for speed of evaluation. We say that a strategy s is *decoupled* iff computing v_{s}^+ for an arbitrary view V only requires the knowledge of upper bounds w_{s}^+ of other views W, but no knowledge of lower bounds w_{s}^-, and vice versa. Thus, for a decoupled bounding strategy, the two bounding functions can be defined independently of each other.

Turning to the estimator, our framework supports different *probabilistic models*. A probabilistic model is a function that, given $\mathcal{I}, \mathcal{D}$, V, as well as bounds computed by the bounder, provides an estimate, $\overline{v}$, for the cardinality of V. In general, this step can use further information from the application domain that is not suitable to derive bounds. Typically this is the case with information concerning average values (e.g., the number of transfers of each employee on each year is 1.5, on the average).

Example 3 Let 10^4 be the number of employees who have been transfered at least once, and let the enterprise consist of 10^3 offices distributed over 10 cities and belonging to one of 10 departments; let 10^3 days be the observation period. Let $V = \{\mathsf{date}, \mathsf{fromOffice}, \mathsf{toOffice}\}$. Since each office is involved in transfers at most with every other office on each date, the first trivial upper bound of v is $10^3 \cdot 10^3 \cdot 10^3 = 10^9$. If the maximum number of transfers for an employee during one year is 2, and since we consider 3 years, it is derived that the cardinality of the base cube is at most $2 \cdot 3 = 6$ times the number of transferred employees, i.e. $6 \cdot 10^4$. Thus, the upper bound of v can be improved to $6 \cdot 10^4$ as well (the cardinality of a view cannot exceed that of its base cube). On the other hand, if we assume that each office is involved in at least one transfer, it is $v \geq 10^3$. Finally, by using the model in Section 6, the cardinality of V is estimated as $\overline{v} = 3.8 \cdot 10^4$. □

5 The Bounder

The basic observation to determine bounds for view cardinalities using bounds of the cardinalities of other views is that the multidimensional lattice induces an isomorphic structure over such cardinalities. In fact, from Definition 3 it follows that $W \preceq Z$ implies $w \leq z$ in each instance of $\mathcal{D}$, since $Z \rightarrow W$ holds. This inequality also applies to bounds.

Lemma 1 *If $W \preceq Z$, then $w^- \leq z^-$ and $w^+ \leq z^+$.*

Proof: ($w^- \leq z^-$) Assume $w^- > z^-$. Then, there is an instance of $\mathcal{D}$ in which $w \geq w^- > z \geq z^-$, thus $w > z$, which is a contradiction. Similarly for $w^+ \leq z^+$. □

As to k-dependencies, their influence on the determination of bounds is summarized by the following lemma.

Lemma 2 *Let $Z = X \oplus Y$. If $X \stackrel{k}{\rightarrow} Y$, then $x^- \geq z^-/k$ and $z^+ \leq k \cdot x^+$.*

Proof: From Definition 4 it follows immediately that, if $X \stackrel{k}{\rightarrow} Y$, the cardinality z of Z is related to the cardinality x of X by inequality $z \leq k \cdot x$. The inequalities on bounds follow immediately. □

In the rest of this section we first propose a decoupled strategy to compute upper bounds (Section 5.1), then we discuss some issues related to coupled strategies (Section 5.2).

5.1 A Decoupled Upper Bounding Strategy

The bounding strategy we propose in this section, called *cover-based*, relies on the concept of *cover* of a view to compute upper bounds. The following are two preliminary definitions whose aim is to precisely characterize how sets of views and kD's can be sinergically combined together.

Definition 6 (Graph of a set of kD's) *Let $K = \{X_1 \stackrel{k_1}{\rightarrow} Y_1, \ldots, X_p \stackrel{k_p}{\rightarrow} Y_p\}$ be a set of kD's. The (labelled oriented)* graph of K *is $\mathcal{G}(K) = (N, E)$, with set of nodes $N = \bigcup_i \{X_i, Y_i\}$, set of edges $E = \{e_i = (X_i, Y_i), i = 1, \ldots, p\}$, and labeling function λ such that $\lambda(e_i) = k_i$.*[2]

Definition 7 (K-set of views) *Let $S = \{W_1, \ldots, W_m\}$ be a non-empty set of views, and let $K = \{X_1 \stackrel{k_1}{\rightarrow} Y_1, \ldots, X_p \stackrel{k_p}{\rightarrow} Y_p\}$ be a set of kD's. The couple $\mathcal{C} =$*

[2] Technically, $\mathcal{G}(K)$ is a *multi-graph*, since two edges may share the same couple of nodes. This, however, does not influence the following arguments.

(S, K) *is called a* k-set of views *iff* $\forall i = 1, \ldots, p$ *it is* $Y_i \in S$ *and there exists a set of kD's,* $K' = \{W_{j_1} \stackrel{k_1}{\rightarrow} Y_1, \ldots, W_{j_p} \stackrel{k_p}{\rightarrow} Y_p\}$*, such that: 1)* $\forall i = 1, \ldots, p$ *it is* $X_i \preceq W_{j_i}$ *with* $W_{j_i} \in S$*, and 2)* $\mathcal{G}(K') = (N', E')$ *is a* forest*, i.e., a set of disjoint directed trees. We call* S-compliant *a set* K' *with such properties.*

Each kD in an S-compliant set K' is derived from a corresponding kD in K by applying rules R1 and R3 (since, by hypothesis, it is $X_i \oplus W_{j_i} = W_{j_i}$). Note that $N' \subseteq S$ always holds and that, in general, multiple S-compliant K' sets can be derived from the same $\mathcal{C}$, depending on how each $W_{j_i} \in S$ is chosen.

Example 4 $\mathcal{C}_1 = (\{A'B, C, D\}, K)$, with $K = \{A'B \stackrel{k_1}{\rightarrow} C, C \stackrel{k_2}{\rightarrow} D\}$, is a k-set of views, since K is S-compliant. The same is true for $\mathcal{C}_2 = (\{AB, C, D\}, K)$, since $K' = \{AB \stackrel{k_1}{\rightarrow} C, C \stackrel{k_2}{\rightarrow} D\}$ is S-compliant (in fact, $A'B \preceq AB$). On the other hand, $\mathcal{C}_3 = (\{B, C, D\}, K)$ is not a k-set since no S-compliant set of kD's can be found.

It is important to remark that Definition 7 requires K', and not necessarily K, to be a forest. For instance, the couple $(\{A\}, \{A' \stackrel{k}{\rightarrow} A\})$ is not a k-set, though $\mathcal{G}(\{A' \stackrel{k}{\rightarrow} A\})$ is a forest, since $\mathcal{G}(\{A \stackrel{k}{\rightarrow} A\})$ is cyclic. On the other hand, $\mathcal{C}_4 = (\{A, A', B\}, \{A' \stackrel{k_1}{\rightarrow} B, B \stackrel{k_2}{\rightarrow} A'\})$ is a k-set (after deriving $A \stackrel{k_1}{\rightarrow} B$ from $A' \stackrel{k_1}{\rightarrow} B$) even if $\mathcal{G}(\{A' \stackrel{k_1}{\rightarrow} B, B \stackrel{k_2}{\rightarrow} A'\})$ is cyclic.

Finally, for the k-set $\mathcal{C}_5 = (\{A, A'B, A'C\}, \{A' \stackrel{k}{\rightarrow} A\})$, two S-compliant sets, $K'_1 = \{A'B \stackrel{k}{\rightarrow} A\}$ and $K'_2 = \{A'C \stackrel{k}{\rightarrow} A\}$, can be derived. □

Definition 8 (Cover) *Let* $V \in \mathcal{V}_\mathcal{D}$ *be a view on* $\mathcal{D}$ *and* $\mathcal{C} = (S, K)$ *be a k-set of views.* $\mathcal{C}$ *is called a* V-cover *iff* $V \preceq \oplus(S)$.

As the following example suggests, a V-cover can be used to bound from above the cardinality of V by generalizing Lemma 1 to the case of multiple views (since $V \preceq \oplus(S)$ holds). When also kD's are present, Lemma 2 can be exploited to improve the bound. Since a cover must be a k-set, we are guaranteed that the cardinalities of some views in S can be safely "replaced" by the k_i's of the kD's in K'.

Example 5 Let $V = ABC$. Below we consider some notable examples of V-covers and show how each of them can be used to derive an upper bound for v. In order to help the reader, Figure 2 depicts the roll-up relationships between the views involved.

- $\mathcal{C}_1 = (\{ABCD\}, \emptyset)$ is a V-cover since $V \preceq \oplus(S_1) = ABCD$. From Lemma 1 it is derived $abc \leq abcd^+$.
- $\mathcal{C}_2 = (\{AB, BC\}, \emptyset)$ is a V-cover since $V \preceq \oplus(S_2) = ABC$. Since the natural join between two views is a subset of their Cartesian product, it is $abc \leq ab^+ \cdot bc^+$.

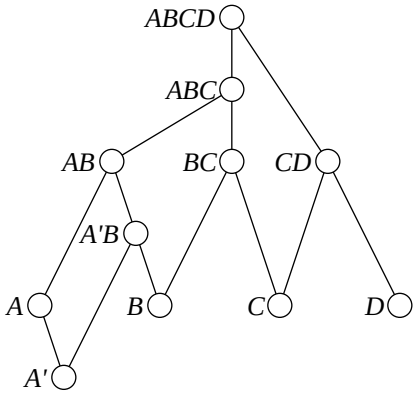

Figure 2: Roll-up relationships of views in Example 5

- $\mathcal{C}_3 = (\{AB, C\}, \{AB \stackrel{k}{\rightarrow} C\})$. From Lemma 2 it immediately follows $abc \leq ab^+ \cdot k$.
- $\mathcal{C}_4 = (\{A, B, C\}, \{A \stackrel{k_1}{\rightarrow} B, B \stackrel{k_2}{\rightarrow} C\})$. By applying rule R2, we derive $A \stackrel{k_1 k_2}{\rightarrow} BC$, thus $abc \leq a^+ \cdot k_1 \cdot k_2$.
- $\mathcal{C}_5 = (\{A, B, C\}, \{A \stackrel{k_1}{\rightarrow} B, A \stackrel{k_2}{\rightarrow} C\})$. Rule R4 is now used to derive $A \stackrel{k_1 k_2}{\rightarrow} BC$, thus $abc \leq a^+ \cdot k_1 \cdot k_2$.
- $\mathcal{C}_6 = (\{A, A'B, C\}, \{A' \stackrel{k}{\rightarrow} A\})$. According to rule R1 it is $A'B \stackrel{k}{\rightarrow} AB$, and from Lemma 2 $ab^+ \leq k \cdot a'b^+$. On the other hand, $abc \leq ab^+ \cdot c^+$, thus $abc \leq k \cdot a'b^+ \cdot c^+$. □

The following theorem precisely characterizes how bounds are related to the graph of K'.

Theorem 1 (Cover-based bounding) *Let* V *be a view and* $\mathcal{C} = (S, K)$ *be a* V*-cover, with* $S = \{W_1, \ldots, W_m\}$ *and* $K = \{X_1 \stackrel{k_1}{\rightarrow} Y_1, \ldots, X_p \stackrel{k_p}{\rightarrow} Y_p\}$*. Let* K' *be an* S*-compliant set,* $R(\mathcal{G}(K'))$ *be the set of root nodes of the forest* $\mathcal{G}(K') = (N', E')$ *associated to* K'*, and let* $S_0 = S - N'$ *stand for the set of views which are not nodes in* $\mathcal{G}(K')$*. Then:*

$$v \leq u(\mathcal{C}, K') \stackrel{\text{def}}{=} \prod_{i=1}^{p} k_i \cdot \prod_{W_j \in R(\mathcal{G}(K')) \cup S_0} w_j^+ \qquad (1)$$

Proof: The intuition behind the proof is that each tree $\mathcal{G}_t = (N'_t, E'_t)$ of $\mathcal{G}(K')$ contributes to $u(\mathcal{C}, K')$ with the upper bound of the cardinality of its root W_t times all the k_i's which label the edges in E'_t.

Since by Definition 8 it is $V \preceq \oplus(S)$, it is sufficient to prove that $u(\mathcal{C}, K')$ is an upper bound of $\oplus(S)$. Since the size of the natural join of a set of views can never exceed

that of their Cartesian product, it is

$$\begin{aligned} card(\oplus(S)) &\leq card((\oplus(S_0))\oplus(\oplus(N'))) \\ &\leq card(\oplus(S_0)) \cdot card(\oplus(N')) \\ &\leq \prod_{W_j \in S_0} w_j^+ \cdot \prod_t card(\oplus(N'_t)) \end{aligned}$$

Since the k_i's are partitioned over the trees, it is enough to prove that $w_t^+ \cdot \prod_{i \in \Lambda'_t} k_i$, where W_t is the root of $\mathcal{G}_t$ and Λ'_t is the set of labels in $\mathcal{G}_t$, is an upper bound of $card(\oplus(N'_t))$. This is proved by induction on the number L of levels in $\mathcal{G}_t$.

Base step (L=2). [3] In this case $\mathcal{G}_t$ corresponds to the set of kD's $\{W_t \stackrel{k_{2,1}}{\rightarrow} W_{2,1}, \ldots, W_t \stackrel{k_{2,q_2}}{\rightarrow} W_{2,q_2}\}$. From the union rule R4 it immediately follows that $card(\oplus(N'_t)) \leq w_t^+ \cdot \prod_{i=1}^{q_2} k_{2,i}$.

Inductive step ($L-1 \Rightarrow L$). Let $N'_t(L-1)$ be the set of nodes in the first $L-1$ levels. By inductive hypothesis it is $card(\oplus(N'_t(L-1))) \leq w_t^+ \cdot \prod_{l=2}^{L-1} \prod_{i=1}^{q_l} k_{l,i}$. Adding the L-th level introduces new edges with labels $k_{L,1}, \ldots, k_{L,q_L}$ and corresponding terminal nodes $W_{L,1}, \ldots, W_{L,q_L}$. From the i-th of the corresponding kD's we can derive (using rules R1 and R3) the kD $\oplus(N'_t(L-1)) \stackrel{k_{L,i}}{\rightarrow} W_{L,i}$. From the union rule R4 it is derived:

$$\oplus(N'_t(L-1)) \stackrel{\prod_{i=1}^{q_L} k_{L,i}}{\rightarrow} \oplus(\{W_{L,1}, \ldots, W_{L,q_L}\})$$

which, due to Lemma 2, leads to:

$$\begin{aligned} &card((\oplus(N'_t(L-1)))\oplus(\oplus(\{W_{L,1}, \ldots, W_{L,q_L}\}))) = \\ &\quad = card(\oplus(N'_t(L))) \\ &\quad \leq \prod_{i=1}^{q_L} k_{L,i} \cdot w_t^+ \cdot \prod_{l=2}^{L-1} \prod_{i=1}^{q_l} k_{l,i} = w_t^+ \cdot \prod_{l=2}^{L} \prod_{i=1}^{q_l} k_{l,i} \quad \square \end{aligned}$$

It is possible to prove that (1) is valid even if $\mathcal{G}(K')$ is not a forest, provided that $R(\mathcal{G}(K'))$ contains (at least) a set of nodes from which *every* other node in $\mathcal{G}(K')$ can be reached through a directed path. On the other hand, the bounds determined by such "non-forest" V-covers are always redundant, meaning that a proper V-cover yielding a better bound for v can always be found.

Example 6 Let $V = ABC$, and consider the couple $(\{A, B, C\}, K)$ with $K = \{A \stackrel{k_1}{\rightarrow} C, B \stackrel{k_2}{\rightarrow} C\}$, which is not a k-set since the graph of $K' = K$ has two roots (A and B). The bound returned by (1) is $v \leq k_1 \cdot k_2 \cdot a^+ \cdot b^+$ which is redundant, since a better bound is obviously obtained through the V-cover $(\{A, B, C\}, \{A \stackrel{k_1}{\rightarrow} C\})$. $\square$

The following lemma shows that, when multiple S-compliant sets exist for a given cover, the bound returned by (1) is actually independent of the one chosen. For instance, the reader may immediately verify that, in Example 4, it is $u(\mathcal{C}_5, K'_1) = u(\mathcal{C}_5, K'_2) = k \cdot a'b^+ \cdot a'c^+$.

Lemma 3 *Let $\mathcal{C} = (S, K)$ be a V-cover, and let K'_1 and K'_2 be two arbitrary S-compliant sets. It is $u(\mathcal{C}, K'_1) = u(\mathcal{C}, K'_2) \stackrel{\text{def}}{=} u(\mathcal{C})$.*

Coherently with Theorem 1 and Lemma 3, the cover-based bounding strategy cb computes v^+_{cb} as:

$$v^+_{\mathsf{cb}} = \begin{cases} v^+ & \text{if } v^+ \in \mathcal{I}, \\ \min\{u_{\mathsf{cb}}(\mathcal{C}) \mid \mathcal{C} \text{ is a } V\text{-cover}\} & \text{if } v^+ \notin \mathcal{I}. \end{cases} \quad (2)$$

where $u_{\mathsf{cb}}(\mathcal{C})$ is obtained by replacing w_j^+ with $w^+_{j,\mathsf{cb}}$ in $u(\mathcal{C})$. In general, evaluating the cover-based bound leads to a recursive computational flow; note that the "case-0" of recursion, $v^+_{\mathsf{cb}} = v^+$, is correctly defined since we assumed the input $\mathcal{I}$ to be minimal.

The space of the V-covers to be analyzed in order to determine v^+_{cb} has exponential size. On the other hand, the following theorem shows that, under some circumstances, a V-cover $\mathcal{C}_2$ can be discarded from the search space without even computing $u_{\mathsf{cb}}(\mathcal{C}_2)$.

Theorem 2 *Let $\mathcal{C}_1 = (S_1, K_1)$ and $\mathcal{C}_2 = (S_2, K_2)$ be two V-covers. If $S_1 \subseteq S_2$ and $K_1 = K_2$ or $S_1 = S_2$ and $K_2 \subseteq K_1$, then $u_{\mathsf{cb}}(\mathcal{C}_1) \leq u_{\mathsf{cb}}(\mathcal{C}_2)$.*

5.1.1 Reasoning without k-dependencies

When no k-dependencies are included among the input constraints $\mathcal{I}$, covers degenerate into sets of views, which allows us to precisely characterize the set of V-covers that can provide useful (non redundant) bounds. To see how such covers are determined, two orthogonal aspects are considered: a *domination* relationship between sets of views and the input information, $\mathcal{I}$. While the former induces a partial order on the bounds obtainable from V-covers, regardless of the specific input $\mathcal{I}$, the latter can be used to restrict the set of useful V-covers to those including only views in $\mathcal{I}$.

In this section, since we assume $K = \emptyset$, we will work only with the S part of V-covers. Consequently, in (2), $u_{\mathsf{cb}}(\mathcal{C})$ can be replaced by

$$u_{\mathsf{cb}}(S) = \prod_{W_i \in S} w_i^+ \ . \quad (3)$$

Definition 9 (Domination between sets of views) *Let $S_1 = \{W_{1,1}, \ldots, W_{1,i}, \ldots, W_{1,m}\}$ and $S_2 = \{W_{2,1}, \ldots, W_{2,j}, \ldots, W_{2,n}\}$ be two sets of views. We say that S_1* dominates *S_2, written $S_1 \sqsubseteq S_2$, iff S_2 can be partitioned into m subsets $S_{2,1}, \ldots, S_{2,m}$ such that $W_{1,i} \preceq \oplus(S_{2,i})$ $\forall i = 1, \ldots, m$.*

[3] The case $L = 1$ cannot arise, since each $\mathcal{G}_t$ has at least one edge.

For instance, $\{A'B, C\} \sqsubseteq \{AB, CD, E\}$. Note that if $S_i \sqsubseteq S_j$ then $\oplus(S_i) \preceq \oplus(S_j)$ necessarily holds, whereas the opposite is not always true (e.g., $\{AB, BC\} \not\sqsubseteq \{ABCD\}$ though $ABC \preceq ABCD$).

Lemma 4 *Let S_1 and S_2 be two sets of views. If $S_1 \sqsubseteq S_2$ then $u_{\mathsf{cb}}(S_1) \leq u_{\mathsf{cb}}(S_2)$.*

Definition 10 (Ground Views and Covers) *We say that a view W is* ground *iff w^+ is in $\mathcal{I}$. A V-cover is said to be* ground *when all the views it includes are ground.*

Lemma 5 *Let S be a non-ground V-cover. Then there exists a ground V-cover S_1 such that $u_{\mathsf{cb}}(S_1) \leq u_{\mathsf{cb}}(S)$.*

Proof (sketch): Since S is not ground, at least one view in S is not ground. By recursively applying (3), $u_{\mathsf{cb}}(S)$ will be eventually expressed as a product of bounds in $\mathcal{I}$. The case of strict inequality ($u_{\mathsf{cb}}(S_1) < u_{\mathsf{cb}}(S)$) can arise since in this recursive process there is no guarantee that a given ground view will be generated just once, thus its least upper bound might appear more than once in $u_{\mathsf{cb}}(S)$. □

Definition 11 (Minimal Cover) *A ground V-cover S is* minimal *iff there is no other ground V-cover S_1 such that $S_1 \sqsubseteq S$ holds.*

The following theorem immediately derives from Lemmas 4 and 5.

Theorem 3 (Sufficiency of Minimal Covers) *It is:*

$$\min\{u_{\mathsf{cb}}(S) \mid S \text{ is a } V\text{-cover}\} = \\ = \min\{u_{\mathsf{cb}}(S) \mid S \text{ is a minimal } V\text{-cover}\}. \quad (4)$$

For instance, let $\mathcal{I} = \{ab'^+, cd^+, a'de^+, a^+, a'^+, b^+, b'^+, c^+, d^+, e^+\}$ and $V = A'B'CD$. The minimal V-covers are $\{AB', CD\}$, $\{A', B', CD\}$, and $\{A'DE, B', C\}$.

From the above results, several facts can be easily derived, which can be exploited to efficiently generate minimal V-covers by means, say, of a branch-and-bound algorithm:

1. A ground view W such that $V \preceq W$ is a ground V-cover (from Definition 8).
2. A ground view W such that $arity(W) = 1$ and $W \cap V = \emptyset$ does not belong to any minimal V-cover[4] (from Definitions 9 and 11).
3. A ground view W such that $arity(W) > 1$ and $\forall W'$ for which $W' \preceq W$ it is $arity(W' \cap V) < 2$ does not belong to any minimal V-cover (since $\mathcal{C}$ includes the cardinalities of all the attributes).

[4] $arity(W)$ denotes the number of attributes in W.

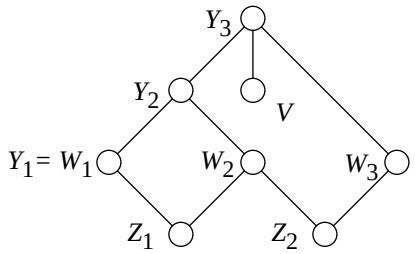

Figure 3: Roll-up relationships between views in Lemma 6, in the case $n = 3$

4. If S is a ground V-cover, no set S' such that $S \subset S'$ is a minimal V-cover (from Definitions 9 and 11).
5. If a minimal V-cover S contains a ground view W, it cannot contain any other ground view W' such that $W \preceq W'$ (from Definitions 9 and 11).

5.2 Towards a Coupled Bounding Strategy

The bounds we derive through the strategy described in Section 5.1 are not necessarily the tightest possible ones. In fact, more complex and effective bounding strategies can be defined to the detriment of computational speed. Basically, in these strategies the concept of cover may be extended by considering more complex patterns of views, where upper and lower bounds are used jointly. In this section we present some preliminary considerations on coupled strategies; for simplicity, we will assume that the input does not contain k-dependencies.

As to upper bounding, the cover-based strategy can be improved by exploiting results from *majorization theory*, which state that the size of the natural join between two relations is majorized when the distributions of the join attribute(s) in the two relations are maximally skewed [IC91]. The extension of this argument to the multidimensional lattice is as follows. Given two views W_1 and W_2 such that $W_1 \not\preceq W_2$ and $W_2 \not\preceq W_1$, let $Y = W_1 \oplus W_2$ and let $Z = W_1 \otimes W_2$, where $\otimes$ is the greatest lower bound operator on the lattice; it can be proved that

$$y \leq w_1^+ \cdot w_2^+ - (z^- - 1)(w_1^+ + w_2^+ - z^-) \quad (5)$$

It should be noted that, when $W_1 \otimes W_2 = \{\}$, since the empty view $\{\}$ has cardinality 1, (5) correctly reduces to (3).

This result can be extended to a V-cover whose views are connected by a *linear join graph*.

Lemma 6 *Let $S = \{W_1, \ldots, W_n\}$ be a V-cover; let $Z_i = W_i \otimes W_{i+1}$, $i = 1, \ldots, n-1$, $Y_1 = W_1$, and $Y_{i+1} = Y_i \oplus W_{i+1}$, $i = 1, \ldots, n-1$. Then:*

$$y_{i+1}^+ \leq y_i^+ \cdot w_{i+1}^+ - (z_i^- - 1)(y_i^+ + w_{i+1}^+ - z_i^-) \\ \text{for } i = 1, \ldots, n-1\ ; \quad (6)$$
$$v \leq y_n^+\ .$$

The pattern consisting of views Y_i, W_{i+1}, Z_i, and Y_{i+1}, depicted in Figure 3, can in principle be extended to take into account also size information on $Y_i - Z_i$ and $W_{i+1} - Z_i$, that is, on non-join attributes; this will further strengthen the upper bound. At present, we guess that the exact computation of v^+ might involve taking into account patterns that can extend over the whole lattice. However, besides the theoretical interest, it is important to trade-off the increased complexity with the actual gain that could be obtained by having more accurate bounds, considering also how bounds can be used by the estimator.

A coupled strategy requires also lower bounds to be computed, which is radically different from computing upper bounds. In fact, while computing an upper bound corresponds to bounding the size of a join, computing a lower bound corresponds to bounding the size of a projection, where the relevant difference is that projection is a unary operator. This leads to a much simpler situation to deal with, in which Lemma 1 is exploited and the lower bound of v is computed as $\max\{w^- \mid w^- \in \mathcal{I}, W \preceq V\}$. Differently from upper bounds, no combinatorial issues arise in computing lower bounds through this strategy; thus, complexity is linear in the cardinality of $\mathcal{I}$.

A better bound can be obtained by using information associated to "sibling" views. Let W be a view such that $V \cap W = \emptyset$, and $Z = V \oplus W$; then:

$$v^- \geq \frac{z^-}{w^+} \tag{7}$$

In fact, if $v < z^-/w^+$, then the size of the Cartesian product of V and W would be less than z^-, which is impossible.

6 The Estimator

Assuming that effective bounds have been derived, cardinality estimation must be based on a probabilistic model to derive an estimate, $\overline{v}$, of the cardinality of view V. The model we adopt here is based on the Cardenas' formula [Car75], which states that, when throwing N distinct objects into B buckets, the expected number of buckets in which at least one object will fall can be estimated as:

$$\Phi(B,N) \stackrel{\text{def}}{=} B \cdot \left(1 - \left(1 - \frac{1}{B}\right)^N\right) \leq \min\{B,N\} \tag{8}$$

Within the approach proposed in [SDNR96], (8) is used to estimate v by relying on the maximum cardinality of V, defined as the Cartesian product of the cardinalities of the attributes in V, $v_{max} = \prod_{A_i \in V} a_i$, and on the cardinality of the base cube, $d = card(dim(\mathcal{D}))$, that is:

$$\overline{v}_{\mathsf{sdnr}} = \Phi(v_{max}, d) \leq \min\{v_{max}, d\} \tag{9}$$

This formula turns out to significantly overestimate the cardinalities and can easily lead to violate the constraint $\overline{v}_{\mathsf{sdnr}} \leq v^+$.

In our approach, denoted se ("safe-estimate"), the above estimate is improved in two ways: by replacing v_{max} with the upper bound computed for v, for instance v^+_{cb}, as a measure of the maximum cardinality of V, and by replacing the cardinality of the base cube d with an estimate, $\overline{w}_{\mathsf{se}}$, of the cardinality of a view W such that $V \preceq W$. This leads to:

$$\overline{v}_{\mathsf{se}} = \Phi(v^+_{\mathsf{cb}}, \overline{w}_{\mathsf{se}}) \leq \min\{v^+_{\mathsf{cb}}, \overline{w}_{\mathsf{se}}\} \tag{10}$$

Since both v^+_{cb} and $\overline{w}_{\mathsf{se}}$ can be considerably lower than v_{max} and d, respectively, it is usually the case that $\overline{v}_{\mathsf{se}} \ll \overline{v}_{\mathsf{sdnr}}$. The rationale for (10) is that we can view the problem of estimating v as the one of distributing the tuples of view W, which are estimated to be $\overline{w}_{\mathsf{se}}$, over a number of v^+_{cb} "buckets".

Due to the need to know $\overline{w}_{\mathsf{se}}$, it is obvious that our estimation process must move downward from the top of the lattice (whose cardinality d is typically known) following a path leading to V. Clearly, this represents a simplification of the correct estimation procedure, which would require to determine $\overline{v}$ by following *all* the paths from $dim(\mathcal{D})$ to V. On the other hand, this would lead to combinatorial explosion and necessitate of highly complex probabilistic models that are well beyond the current state-of-the-art knowledge.

From a more practical (numerical) point of view, it should be noted that moving from upper bounds to estimates leads to significant differences under specific conditions only. Two relevant cases should be considered, which arise from the limit behavior of Cardenas' formula:

1. When $\overline{w}_{\mathsf{se}} \leq 0.1 \cdot v^+_{\mathsf{cb}}$ it is $\overline{v}_{\mathsf{se}} \approx \overline{w}_{\mathsf{se}}$
2. When $\overline{w}_{\mathsf{se}} \geq 3 \cdot v^+_{\mathsf{cb}}$ it is $\overline{v}_{\mathsf{se}} \approx v^+_{f+}$

The values 0.1 and 3 can thus be used to predict whether the estimator will deliver results which substantially differ from those directly obtainable from the bounder.

Example 7 In the *Transfers* scheme, we consider three input situations:

$$\begin{aligned}
\mathcal{I}_1 = &\{\{\mathsf{date}\}^+ = 10^3, \{\mathsf{year}\}^+ = 3, \{\mathsf{employee}\}^+ = 10^4,\\
&\{\mathsf{fromOffice}\}^+ = \{\mathsf{toOffice}\}^+ = 10^3,\\
&\{\mathsf{fromCity}\}^+ = \{\mathsf{toCity}\}^+ = 10,\\
&\{\mathsf{fromDept}\}^+ = \{\mathsf{toDept}\}^+ = 10\}\\
\mathcal{I}_2 = &\mathcal{I}_1 \cup \{\{\mathsf{employee}, \mathsf{year}\} \stackrel{4}{\rightarrow} \{\mathsf{fromOffice}, \mathsf{toOffice}, \mathsf{date}\}\}\\
\mathcal{I}_3 = &\mathcal{I}_2 \cup \{\{\mathsf{fromCity}, \mathsf{fromDept}\}^+ = 40,\\
&\{\mathsf{toCity}, \mathsf{toDept}\}^+ = 40,\\
&\{\mathsf{fromCity}, \mathsf{fromDept}\} \stackrel{2}{\rightarrow} \{\mathsf{toCity}, \mathsf{toDept}\},\\
&\{\mathsf{fromCity}, \mathsf{fromDept}\} \stackrel{30}{\rightarrow} \{\mathsf{fromOffice}\},\\
&\{\mathsf{toCity}, \mathsf{toDept}\} \stackrel{30}{\rightarrow} \{\mathsf{toOffice}\}\}
\end{aligned}$$

Table 1: Improving upper bounds and estimates for increasing domain-derived information

$input$	w^+_{cb}	v^+_{cb}	$\overline{w}_{\mathsf{se}}$	$\overline{v}_{\mathsf{se}}$
$\mathcal{I}_1$	10^{13}	10^6	10^{13}	10^6
$\mathcal{I}_2$	$1.2 \cdot 10^5$	$1.2 \cdot 10^5$	$1.2 \cdot 10^5$	$7.6 \cdot 10^4$
$\mathcal{I}_3$	$1.2 \cdot 10^5$	$7.2 \cdot 10^4$	$1.2 \cdot 10^5$	$5.8 \cdot 10^4$

Let $W = dim(\mathcal{D}) = \{\mathsf{date}, \mathsf{employee}, \mathsf{fromOffice}, \mathsf{toOffice}\}$ be the base cube and $V = \{\mathsf{fromOffice}, \mathsf{toOffice}\}$ be the view whose cardinality is to be estimated. Table 1 shows how the upper bound w^+_{cb} of W, the upper bound v^+_{cb} of V, and the estimate $\overline{v}_{\mathsf{se}}$ improve as new cardinality constraints are progressively supplied. The estimate $\overline{v}_{\mathsf{se}}$ is based on the estimate of w, $\overline{w}_{\mathsf{se}}$, which is assumed to be equal to its upper bound w^+_{cb}. □

7 Conclusions and Open Issues

In this paper we have shown how cardinality constraints derived from the application domain may be employed to determine effective bounds on the cardinality of aggregate views and how, in turn, such bounds can be used to estimate the cardinality of the views. In order to improve the approach effectiveness, some issues still need to be investigated. In the following we briefly discuss those we believe to be crucial:

- *Domination*. A characterization of domination between k-sets of views, similar to that reported in Definition 9 for sets of views, needs to be developed in order to reduce the complexity of computing upper bounds in presence of k-dependencies.
- *Minimality*. Throughout this paper we assumed that the cardinality constraints supplied by the domain expert are sound and non redundant. Of course, this gives rise to the problem of determining, given an input $\mathcal{I}$, if $\mathcal{I}$ is sound and minimal, which we argue can be dealt with as done for, say, functional dependencies (whose inference rules can be used both for schema normalization as well as for input minimization).
- *Cardinality constraints*. The input knowledge may be further extended by considering other forms of cardinality constraints which are typically known to the experts of the application domain. For instance, while in this paper we have defined k-dependencies to express *bounds* on the ratio between the cardinalities of two views, they may also be used to denote the *average* of such ratio; while this kind of knowledge cannot be used by the bounder, it allows the cardinality estimations to be improved. For instance, knowing that the average number of transfers for each employee on each year is 2, would allow the cardinality of the base cube to be estimated as twice the cardinality of view $\{\mathsf{employee}, \mathsf{year}\}$.
- *Probabilistic estimates*. Estimates based on Cardenas' formula can be improved in several ways. In particular, information on lower bounds could be considered by exploiting the results in [CM95], as well as information concerning the distribution of attribute values over their domains. For this, the challenge is to derive new models that can be applied when the data warehouse has not been loaded yet.

References

[AGS97] R. Agrawal, A. Gupta, and S. Sarawagi. Modeling Multidimensional Databases. In *Proc. ICDE'97*, pages 232–243, Birmingham, UK, 1997.

[Car75] A.F. Cardenas. Analysis and Performance of Inverted Database Structures. *Communications of the ACM*, 18(5):253–263, 1975.

[CM92] P. Ciaccia and D. Maio. On the Complexity of Finding Bounds for Projection Cardinalities in Relational Databases. *Information Systems*, 17(6):511–515, 1992.

[CM95] P. Ciaccia and D. Maio. Domains and Active Domains: What This Distinction Implies for the Estimation of Projection Sizes in Relational Databases. *IEEE Transactions on Knowledge and Data Engineering*, 7(4):641–655, 1995.

[GHRU97] H. Gupta, V. Harinarayan, A. Rajaraman, and J. Ullman. Index Selection for OLAP. In *Proc. ICDE'97*, pages 208–219, Birmingham, UK, 1997.

[GL97] M. Gyssens and L.V .S. Lakshmanan. A Foundation for Multi-Dimensional Databases. In *Proc. 23rd VLDB*, pages 106–115, Athens, Greece, 1997.

[GM83] J. Grant and J. Minker. Numerical Dependencies. In H. Gallaire, J. Minker, and J.-M. Nicolas, editors, *Advances in Database Theory*, volume II. Plenum Publ. Co., 1983.

[GR00] M. Golfarelli and S. Rizzi. View Materialization for Nested GPSJ Queries. In *Proc. DMDW'2000*, Stockholm, Sweden, 2000.

[Gup97] H. Gupta. Selection of Views to Materialize in a Data Warehouse. In *Proc. ICDT'97*, pages 98–112, Delphi, Greece, 1997.

[HO91] W. Hou and G. Özsoyoglu. Statistical Estimators for Aggregate Relational Algebra Queries. *ACM Transactions on Database Systems*, 16(4):600–654, 1991.

[HRU96] V. Harinarayan, A. Rajaraman, and J. Ullman. Implementing Data Cubes Efficiently. In *Proc. ACM Sigmod Conf.*, pages 205–216, Montreal, Canada, 1996.

[IC91] Y.E. Ioannidis and S. Christodoulakis. On the Propagation of Errors in the Size of Join Results. In *Proc. ACM Sigmod Conf.*, pages 268–277, Denver, CO, 1991.

[MCS88] M. V. Mannino, P. Chu, and T. Sager. Statistical Profile Estimation in Database Systems. *ACM Computing Surveys*, 20(3):191–221, 1988.

[MD88] M. Muralikrishna and D.J. DeWitt. Equi-depth Histograms for Estimating Selectivity Factors for Multi-Dimensional Queries. In *Proc. ACM Sigmod Conf.*, pages 28–36, Chicago, IL, 1988.

[RS97] K. Ross and D. Srivastava. Fast Computation of Sparse Datacubes. In *Proc. 23rd VLDB*, pages 116–125, Athens, Greece, 1997.

[SDNR96] A. Shukla, P. Deshpande, J. Naughton, and K. Ramasamy. Storage Estimation for Multidimensional Aggregates in the Presence of Hierarchies. In *Proc. 22nd VLDB*, pages 522–531, Mumbai, India, 1996.

[TB00] D. Theodoratos and M. Bouzeghoub. A General Framework for the View Selection Problem for Data Warehouse Design and Evolution. In *Proc. DOLAP 2000*, pages 1–8, Washington, DC, 2000.

[Vas00] P. Vassiliadis. Gulliver in the land of data warehousing: practical experiences and observations of a researcher. In *Proc. DMDW'2000*, pages 12/1–12/16, Stockholm, Sweden, 2000.