SIZE BOUNDS AND QUERY PLANS FOR RELATIONAL JOINS[*]

ALBERT ATSERIAS[†], MARTIN GROHE[‡], AND DÁNIEL MARX[§]

Abstract. Relational joins are at the core of relational algebra, which in turn is the core of the standard database query language SQL. As their evaluation is expensive and very often dominated by the output size, it is an important task for database query optimizers to compute estimates on the size of joins and to find good execution plans for sequences of joins. We study these problems from a theoretical perspective, both in the worst-case model and in an average-case model where the database is chosen according to a known probability distribution. In the former case, our first key observation is that the worst-case size of a query is characterized by the fractional edge cover number of its underlying hypergraph, a combinatorial parameter previously known to provide an upper bound. We complete the picture by proving a matching lower bound and by showing that there exist queries for which the join-project plan suggested by the fractional edge cover approach may be substantially better than any join plan that does not use intermediate projections. On the other hand, we show that in the average-case model, every join-project plan can be turned into a plan containing no projections in such a way that the expected time to evaluate the plan increases only by a constant factor independent of the size of the database. Not surprisingly, the key combinatorial parameter in this context is the maximum density of the underlying hypergraph. We show how to make effective use of this parameter to eliminate the projections.

Key words. fractional edge cover, linear programming, join, database query, query plan

AMS subject classifications. 68P15, 68Q25

DOI. 10.1137/110859440

1. Introduction. The join operation is one of the core operations of relational algebra, which in turn is the core of the standard database query language SQL. The two key components of a database system executing SQL queries are the query optimizer and the execution engine. The optimizer translates the query into several possible execution plans, which are basically terms of the relational algebra (also called operator trees) arranging the operations that have to be carried out in a tree-like order. Using statistical information about the data, the optimizer estimates the execution cost of the different plans and passes the best one on to the execution engine, which then executes the plan and computes the result of the query. See [3] for a survey of query optimization techniques.

Among the relational algebra operations, joins are usually the most costly, simply because a join of two relations, just like a Cartesian product of two sets, may be much larger than the relations. Therefore, query optimizers pay particular attention to the execution of joins, especially to the execution order of sequences of joins, and to estimating the size of joins. In this paper, we address the very fundamental questions

of how to estimate the size of a sequence of joins and how to execute the sequence best
from a theoretical point of view. While these questions have been intensely studied in
practice, and numerous heuristics and efficiently solvable special cases are known (see,
e.g., [3, 9, 7]), the very basic theoretical results we present here and their consequences
apparently have not been noticed so far. Our key starting observation is that the size of
a sequence of joins is tightly linked to two combinatorial parameters of the underlying
database schema, the *fractional edge cover number* and the *maximum density*.

To make this precise, we need to get a bit more technical: A *join query* Q is an
expression of the form

$$(1.1) \qquad R_1(a_{11}, \ldots, a_{1r_1}) \bowtie \cdots \bowtie R_m(a_{m1}, \ldots, a_{mr_m}),$$

where the R_i are *relation names* with *attributes* $a_{i1}, \ldots, a_{ir_i}$. Let A be the set of
all attributes occurring in Q and $n = |A|$. A *database instance* D for Q consists of
relations $R_i(D)$ of arity r_i. It is common to think of the relation $R_i(D)$ as a table
whose columns are labeled by the attributes $a_{i1}, \ldots, a_{ir_i}$ and whose rows are the
tuples in the relation. The *answer*, or *set of solutions*, of the query Q in D is the
n-ary relation $Q(D)$ with attributes A consisting of all tuples t whose projection on
the attributes of R_i belongs to the relation $R_i(D)$ for all i. Hence we are considering
natural joins here. (All of our results can easily be transferred to *equijoins*, but not
to general θ-*joins*.) Now the most basic question is how large $Q(D)$ can get in terms
of the size of the database $|D|$, or more generally, in terms of the sizes of the relations
R_i. We address this question both in the worst case and the average case, and also
subject to various constraints imposed on D.

Example. At this point a simple example would probably help readers to under-
stand what we are after. Let $R(a,b)$, $S(b,c)$, and $T(c,a)$ be three relations on the
attributes a, b, and c. Consider the join query

$$Q(a,b,c) := R(a,b) \bowtie S(b,c) \bowtie T(c,a).$$

The answer of Q is precisely the set of triples (u,v,w) such that $(u,v) \in R$, $(v,w) \in S$
and $(w,u) \in T$. How large can the answer size of Q get as a function of $|R|$, $|S|$,
and $|T|$? First note that a trivial upper bound is $|R| \cdot |S| \cdot |T|$. However, one quickly
notices that an improved bound can be derived from the fact that the relations in
Q have overlapping sets of attributes. Indeed, since any solution for any pair of
relations in Q determines the solution for the third, the answer size of Q is bounded
by $\min\{|R| \cdot |S|, |S| \cdot |T|, |T| \cdot |R|\}$. Now, is this the best general upper bound we can
get as a function of $|R|$, $|S|$, and $|T|$? As it turns out, it is not. Although not obvious,
it will follow from the results in this paper that the optimal upper bound in this case
is $\sqrt{|R| \cdot |S| \cdot |T|}$: the answer size of Q is always bounded by this quantity, and for
certain choices of the relations R, S, T, this upper bound is achieved.

In addition to estimating the answer size of join queries, we also study how to
exploit this information to actually compute the query. An *execution plan* for a join
query describes how to carry out the evaluation of the query by simple operations
of the relational algebra such as joins of two relations or projections. The obvious
execution plans for a join query break up the sequence of joins into pairwise joins
and arrange these in a tree-like fashion. We call such execution plans *join plans*. As
described in [3], most practical query engines simply arrange the joins in some linear
(and not even a tree-like) order and then evaluate them in this order. However, it is
also possible to use other operations, in particular projections, in an execution plan
for a join query. We call execution plans that use joins and projections *join-project*

plans. It is one of our main results that, even though projections are not necessary to evaluate join queries, their use may speed up the evaluation of a query superpolynomially.

Fractional covers, worst-case size, and join-project plans. Recall that an *edge cover* of a hypergraph H is a set C of edges of H such that each vertex is contained in at least one edge in C, and the *edge cover number* $\rho(H)$ of H is the minimum size among all edge covers of H. A *fractional edge cover* of H is a feasible solution for the linear programming relaxation of the natural integer linear program describing edge covers, and the *fractional edge cover number* $\rho^*(H)$ of H is the cost of an optimal solution. With a join query Q of the form (1.1) we can associate a hypergraph $H(Q)$ whose vertex set is the set of all attributes of Q and whose edges are the attribute sets of the relations R_i. The *(fractional) edge cover number* of Q is defined by $\rho(Q) = \rho(H(Q))$ and $\rho^*(Q) = \rho^*(H(Q))$. Note that in the example of the previous paragraph, the hypergraph $H(Q)$ is a triangle. Therefore in that case $\rho(Q) = 2$ while it can be seen that $\rho^*(Q) = 3/2$.

An often observed fact about edge covers is that, for every given database D, the size of $Q(D)$ is bounded by $|D|^{\rho(Q)}$, where $|D|$ is the total number of tuples in D. Much less obvious is the fact that the size of $Q(D)$ can actually be bounded by $|D|^{\rho^*(Q)}$, as proved by the second and third author [10] in the context (and the language) of constraint satisfaction problems. This is a consequence to Shearer's lemma [4], which is a combinatorial consequence of the submodularity of the entropy function and is closely related to a result due to Friedgut and Kahn [6] on the number of copies of a hypergraph in another. Our first and most basic observation is that the fractional edge cover number $\rho^*(Q)$ also provides a lower bound to the worst-case answer size: we show that for every Q, there exist arbitrarily large databases D for which the size of $Q(D)$ is at least $(|D|/|Q|)^{\rho^*(Q)}$. The proof is a simple application of linear programming duality. Another result from [10] implies that for every join query there is a join-project plan, which can easily be obtained from the query and certainly be computed in polynomial time, that computes $Q(D)$ in time $O(|Q|^2 \cdot |D|^{\rho^*(Q)+1})$. Our lower bound shows that this is optimal up to a polynomial factor (of $|Q|^{2+\rho^*(Q)} \cdot |D|$, to be precise). In particular, we get the following equivalences giving an exact combinatorial characterization of all classes of join queries that have polynomial size answers and can be evaluated in polynomial time.

THEOREM 1. *Let $\mathcal{Q}$ be a class of join queries. Then the following statements are equivalent:*

1. *Queries in $\mathcal{Q}$ have answers of polynomial size.*
2. *Queries in $\mathcal{Q}$ can be evaluated in polynomial time.*
3. *Queries in $\mathcal{Q}$ can be evaluated in polynomial time by a join-project plan.*
4. *$\mathcal{Q}$ has bounded fractional edge cover number.*

Note that it is not even obvious that the first two statements are equivalent, that is, that for every class of queries with polynomial size answers there is a polynomial time evaluation algorithm. (The converse, of course, is trivial.)

Hence with regard to worst-case complexity, join-project plans are optimal (up to a polynomial factor) for the evaluation of join queries. Our next result is that join plans are not: We prove that there are arbitrarily large join queries Q and database instances D such that our generic join-project plan computes $Q(D)$ in at most cubic time, whereas any join plan requires time $|D|^{\Omega(\log|Q|)}$ to compute $Q(D)$. We also observe that this bound is tight, i.e., the ratio of the exponents between the best join plan and the best join-project plan is at most logarithmic in $|Q|$. Hence incorporating

projections into a query plan may lead to a superpolynomial speed-up even if the projections are completely irrelevant for the query answer.

Maximum density, average-case size, and join plans. Consider the model $\mathcal{D}(N, (p_R))$ of random databases where the tuples in each relation R are chosen randomly and independently with probability $p_R = p_R(N)$ from a domain of size N. This is the analogue of the Erdős–Rényi model of random graphs adapted to our context. It is easy to see that, for D from $\mathcal{D}(N, (p_R))$, the expected size of the query answer $Q(D)$ is $N^n \cdot \prod_R p_R$, where n is the number of attributes and the product ranges over all relation names R in Q. The question is whether $|Q(D)|$ will be concentrated around the expected value. This is governed by the *maximum density* $\overline{\delta}(Q, (p_R))$ of the query, a combinatorial parameter depending on the hypergraph of the query and the probabilities p_R. An application of the second moment method shows that if $\overline{\delta} = \log N - \omega(1)$, then $|Q(D)|$ is concentrated around its expected value, and if $\overline{\delta} = \log N + \omega(1)$, then $|Q(D)| = 0$ almost surely (a.s.). We observe that the maximum density $\overline{\delta}$ can be computed in polynomial time using max-flow min-cut techniques.

In view of the results about the worst case, it is a natural question whether join-project plans are more powerful than join plans in the average case setting as well. It turns out that this is not the case: We show that every join-project plan φ for Q can be turned into a join plan φ' for which the expected execution time increases only by a constant factor independent of the database. This may be viewed as our main technical result. The transformation of φ' into φ depends on a careful balance between delaying certain joins in order to reduce the number of attributes considered in each subquery occurring in the plan and keeping as many joins as possible in order to increase the density of the subquery. The choice of which subqueries to delay and which to keep is governed by a certain submodular function related to the density of the subqueries.

Size and integrity constraints. So far, we considered worst-case bounds which make no assumptions on the database, and average-case bounds which assume a known distribution on the database. However, practical query optimizers usually exploit additional information about the databases when computing their size estimates. We consider the simplest such setting where the sizes of the relations are known (called histograms in the database literature), and we want to get a (worst-case) estimate on the size of $Q(D)$ subject to the constraint that the relations in D have the given sizes.

By suitably modifying the objective function of the linear program for edge covers, we obtain results analogous to those obtained for the unconstrained setting. A notable difference between the two results is that here the gap between upper and lower bound becomes 2^{-n}, where n is the number of attributes, instead of $|Q|^{-\rho^*}$. We give an example showing that the gap between upper and lower bound is essentially tight. However, this is not an inadequacy of our approach through fractional edge covers but is due to the inherent complexity of the problem: by a reduction from the maximum independent-set problem on graphs, we show that, unless NP = ZPP, there is no polynomial time algorithm that approximates the worst-case answer size $|Q(D)|$ for given Q and relation sizes N_R by a better-than-exponential factor.

Besides the actual sizes of the relations, one could consider other pieces of information that are relevant for estimating the answer size of a query, such as functional dependencies or other integrity constraints that the databases may be specified to satisfy. For example, if an attribute or a set of attributes plays the role of a key in a relation, then the size of that relation is bounded by the size of its projection on the key attributes, and therefore it suffices to analyze the contribution of those attributes

to the maximum answer size of the query. In the preliminary version of this paper we announced some partial results in this direction for the case of simple functional dependencies. Since then, the problem of analyzing the answer size subject to general functional dependencies has been addressed in its own right in the more recent works [8] and [15].

Organization. In section 2 we introduce notation and the basic definitions. In section 3 we state and prove the bounds in the worst-case model. Lemmas 2 and 4 state the upper bound and the lower bound, respectively. Theorem 6 states the fact that, for queries of bounded fractional edge cover number, join-project plans can evaluate the query in polynomial time, and Theorem 7 states that, in contrast, join-only plans cannot. In section 4 we incorporate size constraints into the analysis. Theorem 10 states the upper and lower bounds for this case, and Theorem 11 states that approximating the maximum output-size better than what Theorem 10 gives is NP-hard. In section 5 we study the average-case model. In Theorems 15 and 16 we estimate the output-size as a function of the maximum density of the query. In Theorem 20 we show how to exploit the average-case model to remove projections from any join-project plan without affecting the run-time by more than a constant factor, on average.

2. Preliminaries. For integers $m \leq n$, by $[m, n]$ we denote the set $\{m, m + 1, \ldots, n\}$ and by $[n]$ we denote $[1, n]$. All our logarithms are base 2.

Our terminology is similar to that used in [1]: An *attribute* is a symbol a with an associated *domain* dom(a). If not specified otherwise, we assume dom(a) to be an arbitrary countably infinite set, say, $\mathbb{N}$. Sometimes, we will impose restrictions on the size of the domains. A *relation name* is a symbol R with an associated finite set of attributes A. For a set $A = \{a_1, \ldots, a_n\}$ of attributes, we write $R(A)$ or $R(a_1, \ldots, a_n)$ to denote that A is the set of attributes of R. The *arity* of $R(A)$ is $|A|$. A *schema* is a finite set of relation names. If $\sigma = \{R(A_1), \ldots, R(A_m)\}$, we write A_σ for $\bigcup_i A_i$.

For a set A of attributes, an *A-tuple* is a mapping t that associates an element $t(a)$ from dom(a) with each $a \in A$. Occasionally, we denote A-tuples in the form $t = (t_a : a \in A)$, with the obvious meaning that t is the A-tuple with $t(a) = t_a$. The set of all A-tuples is denoted by tup(A). An *A-relation* is a set of A-tuples. The *active domain* of an A-relation R is the set $\{t(a) : t \in R, a \in A\}$. The *projection* of an A-tuple t to a subset $B \subseteq A$ is the restriction $\pi_B(t)$ of t to B, and the projection of an A-relation R is the set $\pi_B(R) = \{\pi_B(t) : t \in R\}$.

A *database instance* D of schema σ, or a *σ-instance*, consists of an A-relation $R(D)$ for every relation name R in σ with set of attributes A. The *active domain* of D is the union of active domains of all its relations. The *size* of a σ-instance D is $|D| := \sum_{R \in \sigma} |R(D)|$.

A *join query* is an expression

$$Q := R_1(A_1) \bowtie \cdots \bowtie R_m(A_m),$$

where R_i is a relation name with attributes A_i. The *schema* of Q is $\{R_1, \ldots, R_m\}$, and the *set of attributes* of Q is $\bigcup_i A_i$. We often denote the set of attributes of a join query Q by A_Q, and we write tup(Q) instead of tup(A_Q). The *size* of Q is $|Q| := \sum_i |A_i|$. We write $H(Q)$ for the (multi)hypergraph that has vertex-set A_Q and edge-(multi)set $\{A_1, \ldots, A_m\}$. If D is an $\{R_1, \ldots, R_m\}$-instance, the *answer* of Q on D is the A_Q-relation

$$Q(D) = \big\{ t \in \text{tup}(A_Q) : \pi_{A_i}(t) \in R_i(D) \text{ for every } i \in [m] \big\}.$$

A *join plan* is a term built from relation names and binary join operators. For example, $(R_1 \bowtie R_2) \bowtie (R_3 \bowtie R_4)$ and $((R_1 \bowtie R_2) \bowtie R_3) \bowtie (R_1 \bowtie R_4)$ are two join plans corresponding to the same join query $R_1 \bowtie R_2 \bowtie R_3 \bowtie R_4$. A *join-project plan* is a term built from relation names, binary join operators, and unary project operators. For example, $(\pi_A(R_1) \bowtie R_2) \bowtie \pi_B(R_1)$ is a join-project plan. Join-project plans have a natural representation as labeled binary trees, where the leaves are labeled by relation names, the unary nodes are labeled by projections π_A, and the binary nodes are labeled by joins. Evaluating a join plan or join-project plan φ in a database instance D means substituting the relation names by the actual relations from D and carrying out the operations in the expression. We denote the resulting relation by $\varphi(D)$. A join(-project) plan φ is a plan *for* a query Q if $\varphi(D) = Q(D)$ for every database D. The *subplans* of a join(-project) plan are defined in the obvious way. For example, the subplans of $(R_1 \bowtie R_2) \bowtie \pi_A(R_3 \bowtie R_4)$ are R_1, R_2, R_3, R_4, $R_1 \bowtie R_2$, $R_3 \bowtie R_4$, $\pi_A(R_3 \bowtie R_4)$, $(R_1 \bowtie R_2) \bowtie \pi_A(R_3 \bowtie R_4)$. If φ is a join-project plan, then we often use A_φ to denote the set of attributes of the query computed by φ (this only includes "free" attributes and not those projected away by some projection in φ), and we write $\mathrm{tup}(\varphi)$ instead of $\mathrm{tup}(A_\varphi)$.

3. Worst-case model. In this section, we study the worst-case model in which we make no assumptions at all on the database. First we discuss the estimates on the answer-size of join queries, and then we address the question of query plans for such queries.

3.1. Size bounds. Let Q be a join query with schema σ. For every $R \in \sigma$, let A_R be the set of attributes of R, so that $A_\sigma = \bigcup_R A_R$. The fractional edge covers are precisely the feasible solutions $(x_R : R \in \sigma)$ for the following linear program L_Q, and the fractional edge cover number $\rho^*(Q)$ is the cost of an optimal solution:

(3.1)
$$L_Q : \quad \begin{array}{ll} \text{minimize} & \sum_R x_R \\ \text{subject to} & \sum_{R \,:\, a \in A_R} x_R \geq 1 \quad \text{for all } a \in A_\sigma, \\ & x_R \geq 0 \qquad\qquad \text{for all } R \in \sigma. \end{array}$$

By standard arguments, there always is an optimal fractional edge cover whose values are rational and of bit-length polynomial in $|Q|$. As observed in [10], fractional edge covers can be used to give an upper bound on the size of a query.

LEMMA 2 (see [10]). *Let Q be a join query with schema σ and let D be a σ-instance. Then for every fractional edge cover $(x_R : R \in \sigma)$ of Q we have*

$$|Q(D)| \leq \prod_{R \in \sigma} |R(D)|^{x_R} = 2^{\sum_{R \in \sigma} x_R \log |R_D|}.$$

Note that the fractional edge cover in the statement of the lemma is not necessarily one of minimum cost. For the reader's convenience, we give a proof of this lemma, which is actually a simplification of the proof in [10].

The proof of Lemma 2 is based on a combinatorial lemma known as Shearer's lemma. The lemma appeared first in [4], where it is attributed to Shearer. The *entropy* of a random variable X with range U is

$$h[X] := -\sum_{x \in U} \Pr[X = x] \log \Pr[X = x].$$

Shearer's lemma gives an upper bound on the entropy of a distribution on a product space in terms of the entropies of its marginal distributions.

LEMMA 3 (Shearer's lemma). *Let $X = (X_i \mid i \in I)$ be a random variable, and let A_j, for $j \in J$, be (not necessarily distinct) subsets of the index set I such that each $i \in I$ appears in at least k of the sets A_j. For every $B \subseteq I$, let $X_B = (X_i \mid i \in B)$. Then*

$$\sum_{j=1}^{m} h[X_{A_j}] \geq k \cdot h[X].$$

A simple proof of the lemma can be found in [14].

Now we are ready to prove Lemma 2.

Proof of Lemma 2. Let A_R be the set of attributes of $R \in \sigma$ so that $A_\sigma = \bigcup_R A_R$. Without loss of generality we may assume that the fractional edge cover x_R only takes rational values, because the rationals are dense in the reals. Let p_R and q be nonnegative integers such that $x_R = p_R/q$. Let $m = \sum_R p_R$, and let $A_1, \ldots, A_m$ be a sequence of subsets of A_σ that contains precisely p_R copies of the set A_R for all $R \in \sigma$. Then every attribute $a \in A_\sigma$ is contained in at least q of the sets A_i, because

$$\left| \{ i \in [m] : a \in A_i \} \right| = \sum_{R:a \in A_R} p_R \;=\; q \cdot \sum_{R:a \in A_R} x_R \;\geq\; q.$$

Let $X = (X_a \mid a \in A_\sigma)$ be uniformly distributed on $Q(D)$, which we assume to be nonempty, as otherwise the claim is obvious. That is, for every tuple $t \in Q(D)$ we have $\Pr[X = t] = 1/|Q(D)|$, and for all other A-tuples we have $\Pr[X = t] = 0$. Then $h[X] = \log |Q(D)|$. We apply Shearer's lemma to the random variable X and the sets A_R for $R \in \sigma$. (Thus we have $I = A_\sigma$ and $J = \sigma$.) Note that for every $R \in \sigma$ the marginal distribution of X on A_R is 0 on all tuples not in $R(D)$. Hence the entropy of X_{A_R} is bounded by the entropy of the uniform distribution on $R(D)$, that is, $h[X_{A_R}] \leq \log |R(D)|$. Thus by Shearer's lemma, we have

$$\sum_{R \in \sigma} p_R \cdot \log |R(D)| \geq \sum_{R \in \sigma} p_R h[X_{A_R}] = \sum_{i=1}^{m} h[X_{A_i}] \geq q \cdot h[X] = q \cdot \log |Q(D)|.$$

It follows that

$$|Q(D)| \leq 2^{\sum_{R \in \sigma} (p_R/q) \cdot \log |R(D)|} = \prod_{R \in \sigma} |R(D)|^{x_R}. \qquad \square$$

The next lemma shows that the upper bound of the previous lemma is tight.

LEMMA 4. *Let Q be a join query with schema σ, and let $(x_R : R \in \sigma)$ be an optimal fractional edge cover of Q. Then for every $N_0 \in \mathbb{N}$ there is a σ-instance D such that $|D| \geq N_0$ and*

$$|Q(D)| \geq \prod_{R \in \sigma} |R(D)|^{x_R}.$$

Furthermore, we can choose D in such a way that $|R(D)| = |R'(D)|$ for all $R, R' \in \sigma$ with $x_R, x_{R'} > 0$.

Proof. Let A_R be the set of attributes of $R \in \sigma$ so that $A_\sigma = \bigcup_R A_R$. Recall that $(x_R : R \in \sigma)$ is an optimal solution for the linear program (3.1). By linear-programming-duality (LP-duality), there is a solution $(y_a : a \in A_\sigma)$ for the dual linear program

(3.2)   maximize $\sum_a y_a$
         subject to $\sum_{a \in A_R} y_a \leq 1$ for all $R \in \sigma$,
                     $y_a \geq 0$ for all $a \in A_\sigma$

such that $\sum_a y_a = \sum_R x_R$. There even exists such a solution with rational values.

We take an optimal solution $(y_a : a \in A_\sigma)$ with $y_a = p_a/q$, where $q \geq 1$ and $p_a \geq 0$ are integers. Let $N_0 \in \mathbb{N}$, and let $N = N_0^q$. We define a σ-instance D by letting

$$R(D) := \left\{ t \in \mathrm{tup}(A_R) : t(a) \in [N^{p_a/q}] \text{ for all } a \in A_R \right\}$$

for all $R \in \sigma$. Here we assume that $\mathrm{dom}(a) = \mathbb{N}$ for all attributes a. As there is at least one a with $y_a > 0$ and hence $p_a \geq 1$, we have $|D| \geq N^{1/q} = N_0$. Observe that

$$|R(D)| = \prod_{a \in A_R} N^{p_a/q} = N^{\sum_{a \in A_R} y_a} \leq N$$

for all $R \in \sigma$. Furthermore, $Q(D)$ is the set of all tuples $t \in \mathrm{tup}(A_\sigma)$ with $t(a) \in [N^{p_a/q}]$ for every $a \in A_\sigma$. Hence

$$|Q(D)| = \prod_{a \in A} N^{p_a/q} = N^{\sum_{a \in A_\sigma} y_a} = N^{\sum_{R \in \sigma} x_R} = \prod_{R \in \sigma} N^{x_R} \geq \prod_{R \in \sigma} |R(D)|^{x_R},$$

as required. To see that $|R(D)|$ is the same for every relation R with $x_R > 0$, we argue as follows. By complementary slackness of linear programming we have

$$\sum_{a \in A_R} y_a = 1 \quad \text{for all } R \in \sigma \text{ with } x_R > 0.$$

Thus $|R(D)| = N$ for all $R \in \sigma$ with $x_R > 0$ and

$$|Q(D)| = \prod_{R \in \sigma} N^{x_R} = \prod_{R \in \sigma} |R(D)|^{x_R}. \qquad \square$$

Now we show how Lemmas 2 and 4 give the equivalence between statements (1) and (4) of Theorem 1. Assume (1) and let $c > 0$ be a constant such that $|Q(D)| \leq |D|^c$ for every $Q \in \mathcal{Q}$ and every instance D. For a fixed join query $Q \in \mathcal{Q}$, if $(x_R^* : R \in \sigma)$ denotes the optimal fractional edge cover of Q, Lemma 4 states that there exist arbitrarily large instances D such that $|R(D)| = |D|/|\sigma|$ for every $R \in \sigma$ and

$$|Q(D)| \geq \prod_{R \in \sigma} |R(D)|^{x_R^*} \geq (|D|/|Q|)^{\sum_{R \in \sigma} x_R^*} = (|D|/|Q|)^{\rho^*(Q)}.$$

In paricular, there exist arbitrarily large instances D for which $(|D|/|Q|)^{\rho^*(Q)} \leq |D|^c$. It follows that $\rho^*(Q) \leq c$ and hence (4) in Theorem 1. The converse is even more direct. Assume (4) and let $c > 0$ be a constant such that $\rho^*(Q) \leq c$ for every $Q \in \mathcal{Q}$. For a fixed join query $Q \in \mathcal{Q}$, if $(x_R^* : R \in \sigma)$ denotes the optimal fractional edge cover of Q, Lemma 2 states that for every instance D we have

$$|Q(D)| \leq \prod_{R \in \sigma} |R(D)|^{x_R^*} \leq |D|^{\sum_{R \in \sigma} x_R^*} = |D|^{\rho^*(Q)}.$$

It follows that $|Q(D)| \leq |D|^c$ for every D and hence (1) in Theorem 1.

3.2. Execution plans. It was proved in [10] that there is an algorithm for evaluating a join query Q in a database D that runs in time $O\left(|Q|^2 \cdot |D|^{\rho^*(Q)+1}\right)$. An analysis of the proof shows that the algorithms can actually be cast as the evaluation of an explicit (and simple) join-project plan. For the reader's convenience, we give a

proof of this fact here. Combined with the bounds obtained in the previous section, this yields Theorem 1.

We define the *size* of a k-ary relation R to be the number $||R|| := |R| \cdot k$. The bounds stated in the following fact depend on the machine model; the statement we give is based on standard random access machines with a uniform cost measure. Other models may require additional logarithmic factors.

FACT 5. *The following hold:*
1. *The join $R \bowtie S$ of two relations R and S can be computed in time $O(||R|| + ||S|| + ||R \bowtie S||)$.*
2. *The projection $\pi_B(R)$ of an A-relation R to a subset $B \subseteq A$ can be computed in time $O(||R||)$.*

For details and a proof of the fact, we refer the reader to [5]. The following theorem gives the promised join-project plan.

THEOREM 6. *For every join query Q, there is a join-project plan for Q that can be evaluated in time $O\big(|Q|^2 \cdot |D|^{\rho^*(Q)+1}\big)$ on every given instance D. Moreover, there is a polynomial-time algorithm that, given Q, computes the join-project plan.*

Proof. Let $Q = R_1(A_1) \bowtie \cdots \bowtie R_m(A_m)$ be a join query and D an instance for Q. Suppose that the attributes of Q are $\{a_1, \ldots, a_n\}$. For $i \in [n]$, let $B_i := \{a_1, \ldots, a_i\}$. Furthermore, let

$$\varphi_1 := \big(\cdots(\pi_{B_1}(R_1) \bowtie \pi_{B_1}(R_2)) \bowtie \cdots \bowtie \pi_{B_1}(R_m)\big),$$
$$\varphi_{i+1} := \Big(\cdots\big((\varphi_i \bowtie \pi_{B_{i+1}}(R_1)) \bowtie \pi_{B_{i+1}}(R_2)\big) \bowtie \cdots \bowtie \pi_{B_{i+1}}(R_m)\Big) \quad \text{for all } i \geq 1.$$

It is easy to see that for every $i \in [n]$ it holds that $\varphi_i(D) = \pi_{B_i}(Q(D))$ and hence $\varphi_n(D) = Q(D)$. Hence to compute $Q(D)$, we can evaluate the join-project plan φ_n.

To estimate the cost of the evaluating the plan, we need to establish the following claim: For every $i \in [n]$ we have $|\varphi_i(D)| \leq |D|^{\rho^*(Q)}$. To see this, we consider the join query

$$Q_i := R_1{}^i \bowtie \cdots \bowtie R_m{}^i,$$

where $R_j{}^i$ is a relation name with attributes $B_i \cap A_j$. The crucial observation is that $\rho^*(Q_i) \leq \rho^*(Q)$, because if $(x_R : R \in \sigma)$ is fractional edge cover of Q, then letting $x_{R^i} = x_R$ for every $R \in \sigma$ we get a fractional edge cover of Q_i of the same cost. If we let D_i be the database instance with $R_j{}^i(D_i) := \pi_{B_i}(R_j)$ for all $j \in [m]$, then we get

$$\varphi_i(D) = Q_i(D_i) \leq |D_i|^{\rho^*(Q_i)} \leq |D|^{\rho^*(Q)}.$$

This proves the claim.

We further observe that all intermediate results in the computation of $\varphi_{i+1}(D)$ from $\varphi_i(D)$ are contained in

$$\varphi_i(D) \times U,$$

where U is the active domain of D. Hence their size is bounded by $|\varphi_i(D)| \cdot |D| \leq |D|^{\rho^*(Q)+1}$, and by Fact 5 they can be computed in time $O(|D|^{\rho^*(Q)+1})$. Overall, we have to compute $n \cdot m$ projections, each requiring time $O(D)$, and $n \cdot m$ joins, each requiring time $O(|D|^{\rho^*(Q)+1})$. This yields the desired running time. $\quad\square$

We shall prove next that join plans perform significantly worse than join-project plans. Note that to evaluate a join plan one has to evaluate all its subplans. Hence

for every subplan ψ of φ and every instance D, the size $|\psi(D)|$ is a lower bound for the time required to evaluate φ in D.

THEOREM 7. *For every $m, N \in \mathbb{N}$ there are a join query Q and an instance D with $|Q| \geq m$ and $|D| \geq N$, and the following hold:*

1. $\rho^*(Q) \leq 2$ *and hence* $|Q(D)| \leq |D|^2$ *(actually, $|Q(D)| \leq |D|$).*
2. *Every join plan φ for Q has a subplan ψ such that* $|\psi(D)| \geq |D|^{\frac{1}{5}\log|Q|}$.

Proof. Let $n = \binom{2m}{m}$. For every $s \subseteq [2m]$ with $|s| = m$, let a_s be an attribute with domain $\mathbb{N}$. For every $i \in [2m]$, let R_i be a relation name having as attributes all a_s such that $i \in s$. Let A_i be the set of attributes of R_i and $A = \bigcup_{i \in [2m]} A_i$. The arity of R_i is

$$|A_i| = \binom{2m-1}{m-1} = \frac{m}{2m} \cdot \binom{2m}{m} = \frac{n}{2}.$$

Let $Q := R_1 \bowtie \cdots \bowtie R_{2m}$. Then $|Q| = 2m \cdot n/2 = m \cdot n$. Furthermore, $\rho^*(Q) \leq 2$. To see this, let $x_{R_i} = 1/m$ for every $i \in [2m]$. This forms a fractional edge cover of Q, because for every $s \subseteq [2m]$ with $|s| = m$, the attribute a_s appears in the m atoms R_i with $i \in s$.

Next, we define an instance D by letting $R_i(D)$ be the set of all A_i-tuples that have an arbitrary value from $[N]$ in one coordinate and 1 in all other coordinates. Formally,

$$R_i(D) := \bigcup_{a \in A_i} \bigcap_{b \in A_i \setminus a} \{t \in \mathrm{tup}(A_i) : t(a) \in [N], t(b) = 1\}.$$

Observe that $|R_i(D)| = (N-1)n/2 + 1$ for all $i \in [2m]$ and thus

$$|D| = (N-1)mn + 2m \geq N.$$

Furthermore, $Q(D)$ is the set of all A-tuples that have an arbitrary value from $[N]$ in one attribute and 1 in all other coordinates. (It is not possible that two attributes have value different from 1, as every two attributes appear together in some relation.) Hence $|Q(D)| = (N-1)n + 1 \leq |D|$. This completes the proof of (1).

To prove (2), we shall use the following simple (and well-known) combinatorial lemma.

LEMMA 8. *Let T be a binary tree whose leaves are colored with $2m$ colors for some $m \geq 1$. Then there exists a node t of T such that at least $(m+2)/2$ and at most $m+1$ of the colors appear at leaves that are descendants of t.*

Proof. For every node t of T, let $c(t)$ be the number of colors that appear at descendants of T. The *height* of a node t is the length of the longest path from t to a leaf.

Let t be a node of minimum height such that $c(t) \geq m + 2$, and let u_1, u_2 be the children of t. (Note that t cannot be a leaf because $c(t) \geq 2$.) Then $c(u_i) \leq m + 1$ for $i = 1, 2$. Furthermore, $c(u_1) + c(u_2) \geq c(t)$, and hence $c(u_i) \geq (m+2)/2$ for at least one i. $\square$

Continuing the proof of the theorem, we let φ be a join plan for Q. We view the term φ as a binary tree T whose leaves are labeled by atoms R_i. We view the atoms as colors. Applying the lemma, we find a node t of T such that at least $(m+2)/2$ and at most $m+1$ of the colors appear at leaves that are descendants of t. Every inner node of the tree corresponds to a subplan of φ. We let ψ be the subplan corresponding to t. Then at least $(m+2)/2$ and at most $m+1$ atoms R_i appear in ψ. By symmetry,

we may assume without loss of generality that the atoms of ψ are $R_1, \ldots, R_\ell$ for some $\ell \in \left[\lceil (m+2)/2 \rceil, m+1 \right]$. Hence ψ is a plan for the join query

$$R_1 \bowtie \cdots \bowtie R_\ell.$$

Let $B := \bigcup_{i=1}^{\ell} A_i$ be the set of all attributes occurring in ψ. For $i \in [m+1]$, let $s_i = \{i\} \cup [m+2, 2m]$. Then for all $i, j \in [\ell]$ we have $a_{s_i} \in A_j$ if and only if $i = j$. Hence all tuples $t \in \text{tup}(B)$ with $t(a_{s_i}) \in [N]$ for all $i \in [\ell]$ and $t(b) = 1$ for all $b \in B \setminus \{a_{s_1}, \ldots, a_{s_\ell}\}$ are contained in $\psi(D)$. As there are N^ℓ such tuples, it follows that

$$|\psi(Q)| \geq N^\ell \geq N^{(m+2)/2}.$$

Statement 2 of the lemma follows, because

$$\log |Q| = \log m + \log n \leq \log m + \log 2^{2m} = \log m + 2m \leq 5 \cdot (m+2)/2,$$

provided m is large enough, which we may assume without loss of generality. □

Statement 2 of the theorem implies that any evaluation algorithm for the query Q based on evaluating join plans, which may even depend on the database instance, has a running time at least $|D|^{\Omega(\log |Q|)}$. This is to be put in contrast with the running time $O(|Q|^2 \cdot |D|^3)$ from Theorem 6. It is a natural question to ask if the difference can be even worse, i.e., more than logarithmic in the exponent.

Using the well-known fact that the integrality gap of the linear program for edge covers is logarithmic in the number of vertices of the hypergraph (that is, attributes of the join query), we prove below that for every query Q there is a join plan φ that can be evaluated in time $O(|Q| \cdot |D|^{2\rho^*(Q) \cdot \log |Q|})$, and hence the lower bound is tight up to a small constant factor.

PROPOSITION 9. *For every join query Q, there is a join plan for Q that can be evaluated in time $O(|Q| \cdot |D|^{2\rho^*(Q) \cdot \log |Q|})$ on every given instance D.*

Proof. Let Q be a join query with schema σ. For every $R \in \sigma$ let A_R be the set of attributes of R so that $A_\sigma = \bigcup_{R \in \sigma} A_R$. An *edge cover* of Q is a subset $\gamma \subseteq \sigma$ such that $A_\sigma \subseteq \bigcup_{R \in \gamma} A_R$. The *edge cover number* $\rho(Q)$ of Q is the minimum size of an edge cover for Q. Observe that edge covers correspond to $\{0, 1\}$-valued fractional edge covers and that the edge cover number is precisely the cost of the optimal integral fractional edge cover. It is well known that the integrality gap for the linear program defining fractional edge covers is H_n, where $n = |A_\sigma|$ and H_n is the nth harmonic number (see, for example, [16, Chapter 13]). It is known that $H_n \leq 2 \log n$. Now the join plan consists in first joining the relations that form an edge cover of size $2\rho^*(Q) \cdot \log |Q|$ in arbitrary order and then joining the result with the rest of relations in arbitrary order. □

Furthermore, the proof of Proposition 9 shows that, for every join query Q, there is a join plan that can be evaluated in time $O(|Q| \cdot |D|^{\rho(Q)})$, where $\rho(Q)$ denotes the edge cover number of Q. However, note not only that $|D|^{\rho(Q)}$ is potentially superpolynomial over $|D|^{\rho^*(Q)}$, but also finding this plan is in general NP-hard. Compare this with the fact that the join-project plan given by [10] can be found efficiently (see Theorem 6).

4. Size constraints. To estimate the size of joins, practical query optimizers use statistical information about the database instance such as the sizes of the relations, the sizes of some of their projections, or histograms. In this section we consider the simplest such setting where the size of the relations is known, and we prove a (worst-case) estimate on the size of $Q(D)$ subject to the constraint that the relations in D have the given sizes.

4.1. Size bounds under size constraints. Let Q be a join query with schema σ. For every $R \in \sigma$, let A_R be the set of attributes of R so that $A_\sigma = \bigcup_R A_R$. For every $R \in \sigma$, let N_R be a natural number, and let $L_Q(N_R : R \in \sigma)$ be the following linear program:

$$
(4.1) \quad
\begin{aligned}
\text{minimize} \quad & \sum_R x_R \cdot \log N_R \\
\text{subject to} \quad & \sum_{R:a \in A_R} x_R \geq 1 \quad \text{for all } a \in A_\sigma, \\
& x_R \geq 0 \qquad\qquad \text{for all } R \in \sigma.
\end{aligned}
$$

Note that the only difference with L_Q as defined in (3.1) is the objective function. This implies that every feasible solution of $L_Q(N_R : R \in \sigma)$ is also a fractional edge cover of Q.

THEOREM 10. *Let Q be a join query with schema σ and let $N_R \in \mathbb{N}$ for all $R \in \sigma$. Let n be the number of attributes of Q, and let $(x_R : R \in \sigma)$ be an optimal solution of the linear program $L_Q(N_R : R \in \sigma)$.*

1. *For every D with $|R(D)| = N_R$ for all R it holds that $|Q(D)| \leq \prod_R N_R^{x_R}$.*
2. *There is a D with $|R(D)| = N_R$ for all $R \in \sigma$ and $|Q(D)| \geq 2^{-n} \prod_R N_R^{x_R}$.*

Proof. Statement 1 is an immediate consequence of Lemma 2. To prove 2, we exploit LP-duality again. The LP-dual of $L_Q(N_R : R \in \sigma)$ is the following linear program $D_Q(N_R : R \in \sigma)$:

$$
\begin{aligned}
\text{maximize} \quad & \sum_a y_a \\
\text{subject to} \quad & \sum_{a \in A_R} y_a \leq \log N_R \quad \text{for all } R \in \sigma, \\
& y_a \geq 0 \qquad\qquad\quad \text{for all } a \in A_\sigma.
\end{aligned}
$$

Let $(y_a : a \in A_\sigma)$ be an optimal solution for the dual. Then $\sum_{a \in A_\sigma} y_a = \sum_{R \in \sigma} x_R \cdot \log N_R$.

For all $a \in A_\sigma$, let $y'_a = \log \lfloor 2^{y_a} \rfloor \leq y_a$. We set

$$
R' := \Big\{ t \in \mathrm{tup}(A_R) : t(a) \in \big[2^{y'_a}\big] \text{ for all } a \in A_R \Big\}.
$$

Then

$$
|R'| = \prod_{a \in A_R} 2^{y'_a} = \prod_{a \in A_R} \lfloor 2^{y_a} \rfloor \leq 2^{\sum_{a \in A_R} y_a} \leq 2^{\log N_R} = N_R.
$$

We arbitrarily add tuples to R' to obtain a relation $R(D)$ of size exactly N_R. In the resulting instance D, we have

$$
|Q(D)| \geq \prod_{a \in A_\sigma} 2^{y'_a} \geq \prod_{a \in A_\sigma} 2^{y_a - 1} = 2^{-n} \cdot 2^{\sum_{a \in A_\sigma} y_a} = 2^{-n} \cdot 2^{\sum_{R \in \sigma} x_R \cdot \log N_R},
$$

and this is precisely $2^{-n} \prod_{R \in \sigma} N_R^{x_R}$. □

Even though usually the query is much smaller than the database instance, and hence we may argue that a constant factor that only depends on the size of the query is negligible, the exponential factor in the lower bound of Theorem 10(2) is unpleasant. In the following, we shall prove that the lower bound cannot be improved substantially. In the next example we show that we cannot replace the lower bound of Theorem 10(2) by $2^{-(1-\epsilon)n} \prod_R N_R^{x_R}$ for any $\epsilon > 0$. This seems to indicate that maybe the approach to estimating the size of joins through fractional edge covers is no longer appropriate in the setting where the size of the relations is fixed. However,

we shall then see that, in some sense, there is no better approach. In Theorem 11, we shall prove that there is no polynomial time algorithm that, given a query Q and relation sizes N_R, for $R \in \sigma$, approximates the worst-case size of the query answer to a factor better than $2^{n^{1-\epsilon}}$.

Example. We give an example where $\prod_{R \in \sigma} N_R^{x_R}$ is roughly 2^n but $|Q(D)|$ is at most $2^{\epsilon n}$, where n is the number of attributes of Q. Thus the factor 2^{-n} in Theorem 10(2) cannot be replaced with anything greater than $2^{-(1-\epsilon)n}$.

Let $n \in \mathbb{N}$ be an integer, $0 < \epsilon < 1$ a fixed constant, and $A = \{a_1, \dots, a_n\}$ a set of attributes with domain $\mathbb{N}$. Let $r := \lfloor \epsilon n / \log n \rfloor$. We assume that n is sufficiently large so that $2^r > n$ holds. For every $B \in \binom{[n]}{r}$, let R_B be an r-ary relation with attributes B. Furthermore, for every $a \in A$, let R_a be a unary relation with the only attribute a. Let Q be the join of all these relations and let σ be the resulting schema.

For every $B \in \binom{[n]}{r}$, let $N_{R_B} = 2^r - 1$, and for every $a \in A$, let $N_{R_a} = 2$. Consider the linear program $L_Q(N_R : R \in \sigma)$. We obtain an optimal solution for this linear program by letting $x_{R_B} := n / \left(r \binom{n}{r} \right)$ and $x_{R_a} := 0$. To see that this is an optimum solution, observe that $y_a := \log(2^r - 1)/r$ is a feasible solution of the dual-LP with the same cost.

Now

$$\prod_{R \in \sigma} N_R^{x_R} = \left((2^r - 1)^{n/(r\binom{n}{r})} \right)^{\binom{n}{r}} \geq (2^r - 1)^{n/r} \geq (2^r (1 - 1/n))^{n/r} = 2^n (1 - o(1)).$$

The second inequality follows from $2^r > n$, and the last equality holds because, as n approaches infinity, $(1 - 1/n)^n$ approaches $1/e$ and r approaches infinity.

To complete the example, we prove that $|Q(D)| \leq 2^{\epsilon n}$ for every instance D respecting the constraints N_R. Let D be a σ-instance with $|R(D)| = N_R$ for every $R \in \sigma$. From $N_{R_a} = 2$ it follows that in $Q(D)$ each attribute has at most two values, and hence we can assume without loss of generality that $Q(D) \subseteq \{0,1\}^n$. Thus each tuple in $t \in Q(D)$ can be viewed as a subset $A_t = \{a \in A : t(a) = 1\}$ of A. For every $B \in \binom{[n]}{r}$, it holds that $\pi_B(Q(D)) \leq N_{R_B} = 2^r - 1$, and hence the Vapnik–Chervonenkis dimension of $Q(D)$ is less than r. Thus by Sauer's lemma, we have

$$|Q(D)| \leq n^r \leq n^{\epsilon n / \log n} = 2^{\epsilon n},$$

as claimed.

4.2. Hardness of better approximation. There is a gap of 2^n between the upper and lower bounds of Theorem 10, which means that both bounds approximate the maximum size of $|Q(D)|$ within a factor of 2^n. However, if $|Q(D)|$ is $2^{O(n)}$, then such an approximation is useless. We show that it is not possible to find a better approximation in polynomial time: the gap between an upper and a lower bound cannot be reduced to $2^{O(n^{1-\epsilon})}$ (under standard complexity-theoretic assumptions).

For the following statement, recall that ZPP is the class of decision problems that can be solved by a probabilistic polynomial-time algorithm with *zero-error*. What this means is that, on any input, the algorithm outputs the correct answer or "don't know," but the probability over the random choices of the algorithm that the answer is "don't know" is bounded by $1/2$. Obviously $\mathrm{P} \subseteq \mathrm{ZPP} \subseteq \mathrm{NP}$, and the assumption that $\mathrm{ZPP} \neq \mathrm{NP}$ is almost as believable as $\mathrm{P} \neq \mathrm{NP}$ (see [13]).

THEOREM 11. *For a given query Q with schema σ and a given set of size constraints $(N_R : R \in \sigma)$, denote by M the maximum of $|Q(D)|$ over databases satisfying $|R(D)| = N_R$ for every $R \in \sigma$. If for some $\epsilon > 0$ there is a polynomial-time algorithm*

that, given a query Q with n attributes and size constraints N_R, computes two values M_L and M_U with $M_L \leq M \leq M_U$ and $M_U \leq M_L 2^{n^{1-\epsilon}}$, then ZPP = NP.

For the proof of Theorem 11, we establish a connection between the query size and the maximum independent set problem (Lemma 13). Then we get our inapproximability result by reduction from the following result by Håstad.

THEOREM 12 (see [11]). *If for some $\epsilon_0 > 0$ there is a polynomial-time algorithm that, given an n-vertex graph G, can distinguish between the cases $\alpha(G) \leq n^{\epsilon_0}$ and $\alpha(G) \geq n^{1-\epsilon_0}$, then* ZPP = NP.

Following is the announced connection between worst-case query-size subject to relation-size constraints and maximum independent sets.

LEMMA 13. *Let Q be a join query with schema σ and let $N_R := 2$ for all $R \in \sigma$. Let G be the primal graph of Q and let $\alpha(G)$ be the size of the maximum independent set in G. The maximum of $|Q(D)|$, taken over database instances satisfying $|R(D)| = N_R$ for every $R \in \sigma$, is exactly $2^{\alpha(G)}$.*

Proof. Let A_R be the attributes of $R \in \sigma$. For this proof we write A instead of A_σ. First we give a database D with $|Q(D)| \geq 2^{\alpha(G)}$. Let $I \subseteq A$ be an independent set of size $\alpha(G)$. Since I is independent, $|A_R \cap I|$ is either 0 or 1 for every $R \in \sigma$. If $|A_R \cap I| = 0$, then we define $R(D)$ to contain a tuple that is 0 on every attribute. If $A_R \cap I = \{a\}$, then we define $R(D)$ to contain a tuple that is 0 on every attribute and a tuple that is 1 on a and 0 on every attribute in $A_R \setminus \{a\}$. We claim that

$$Q(D) = \{t \in \mathrm{tup}(A) : t(a) \in \{0,1\} \text{ for all } a \in I, t(a) = 0 \text{ for all } a \in A \setminus I\}.$$

Clearly, the value of an attribute in I is either 0 or 1, and every attribute in $A \setminus I$ is forced to 0. Furthermore, any combination of 0 and 1 on the attributes of I is allowed as long as all the other attributes are 0. Thus $|Q(D)| = 2^{\alpha(G)}$. Note that a relation R with $|A_R \cap I| = 0$ contains only one tuple in the definition above. To satisfy the requirement $|R(D)| = N_R = 2$, we can add an arbitrary tuple to each such relation R; this cannot decrease $|Q(D)|$.

Next we show that if $|R(D)| = 2$ for every relation $R \in \sigma$, then $|Q(D)| \leq 2^{\alpha(G)}$. Since $|R(D)| = 2$ for every relation, every attribute in A can have at most two values in $Q(D)$; without loss of generality it can be assumed that $Q(D) \subseteq \{0,1\}^{|A|}$. Furthermore, it can be assumed (by a mapping of the domain of the attributes) that the all-0 tuple is in $Q(D)$.

Let S be the set of those attributes that have two values in $Q(D)$, i.e.,

$$S = \{a \in A : |\pi_{\{a\}}(Q(D))| = 2\}.$$

For every $a \in S$, let S_a be the set of those attributes that are the same as a in every tuple of $Q(D)$, i.e.,

$$S_a = \{b \in S : t(a) = t(b) \text{ for every } t \in Q(D)\}.$$

We define a sequence $a_1, a_2, \ldots$ of attributes by letting a_i be an arbitrary attribute in $S \setminus \bigcup_{j<i} S_{a_j}$. Let a_t be the last element in this sequence, which means that $\bigcup_{i=1}^{t} S_{a_i} = S$. We claim that $a_1, \ldots, a_t$ are independent in G, implying $t \leq \alpha(G)$. Assume that a_i and a_j $(i < j)$ are adjacent in G; this means that there is an $R \in \sigma$ with $a_i, a_j \in A_R$. By assumption, the all-0 tuple is in $R(D)$. As $a_i, a_j \in S$, there has to be a $t_1 \in R(D)$ with $t_1(a_i) = 1$ and a $t_2 \in R(D)$ with $t_2(a_j) = 1$. Since $|R(D)| = 2$ and the all-0 tuple is in $R(D)$, we have $t_1 = t_2$. But this means that a_i and a_j have the same value in both tuples in $R(D)$, implying $a_j \in S_{a_i}$. However, this contradicts the way the sequence was defined.

Now it is easy to see that $|Q(D)| \leq 2^t \leq 2^{\alpha(G)}$: by setting the value of $a_1, \ldots, a_t$, the value of every attribute in S is uniquely determined and the attributes in $A \setminus S$ are the same in every tuple of $Q(D)$. $\qquad \square$

Proof of Theorem 11. We show that if such M_L and M_U could be determined in polynomial time, then we would be able to distinguish between the two cases of Theorem 12. Given an n-vertex graph $G = (V, E)$, we construct a query Q with attributes V and schema $\sigma = E$. For each edge $uv \in E$, there is a relation R_{uv} with attributes $\{u, v\}$. We set $N_R = 2$ for every relation $R \in \sigma$. Observe that the primal graph of Q is G. Thus by Lemma 13, $M = 2^{\alpha(G)}$.

Set $\epsilon_0 := \epsilon/2$. In case (1) of Theorem 12, $\alpha(G) \leq n^{\epsilon_0}$, and hence $M_L \leq M \leq 2^{n^{\epsilon_0}}$ and

$$M_U \leq M_L 2^{n^{1-\epsilon}} \leq 2^{n^{\epsilon_0} + n^{1-\epsilon}} < 2^{n^{1-\epsilon_0}}$$

(if n is sufficiently large). On the other hand, in case (2) we have $\alpha(G) \geq n^{1-\epsilon_0}$, which implies $M_U \geq M = 2^{\alpha(G)} \geq 2^{n^{1-\epsilon_0}}$. Thus we can distinguish between the two cases by comparing M_U with $2^{n^{1-\epsilon_0}}$. $\qquad \square$

5. Average-case model. In this section we assume that the database is randomly generated according to the following model. Let σ be a schema and let A_R be the set of attributes of $R \in \sigma$. For every $R \in \sigma$, let $p_R : \mathbb{N} \to (0, 1)$ be a function of N, and let $p(N) = (p_R(N) : R \in \sigma)$. We denote by $\mathcal{D}(N, p(N))$ the probability space on σ-instances with domain $[N]$ defined by placing each tuple $t \in [N]^{A_R}$ in $R(D)$ with probability $p_R(N)$, independently for each tuple t and each $R \in \sigma$. Typical probability functions of interest are $p_R(N) = 1/2$, $p_R(N) = C \cdot N^{1-|A_R|}$, or $p_R(N) = N^{1-|A_R|} \log N$. When $p_R(N) = 1/2$ for every $R \in \sigma$, we are dealing with the uniform distribution over σ-instances with domain $[N]$.

5.1. Size bounds and concentration. Let Q be a join query with schema σ, let n be the number of attributes of Q, and let m be the number of relation names in σ. Let X denote the size of the query answer $Q(D)$ when D is taken from $\mathcal{D}(N, p(N))$. The expectation of X is, trivially,

$$(5.1) \qquad \mathrm{E}[X] = N^n \prod_{R \in \sigma} p_R(N).$$

We want to determine under what circumstances $|Q(D)|$ is concentrated around this value. For this we need to compute the variance of X, which depends on a parameter of Q defined next.

For every $R \in \sigma$, let w_R be a positive real weight, and let $w = (w_R : R \in \sigma)$. The *density* of Q with respect to w is defined as $\delta(Q, w) = \frac{1}{n} \sum_{R \in \sigma} w_R$. Note that if $w_R = 1$ for every R, then the density is m/n. For every $B \subseteq A_\sigma$, let $Q[B]$ denote the subquery induced by B, that is, $Q[B]$ is the subquery formed by all the atoms $R \in \sigma$ that have all attributes in B. The *maximum density* of Q with respect to w is $\overline{\delta}(Q, w) = \max\{\delta(Q[B], w) : B \subseteq A_\sigma, B \neq \emptyset\}$.

In applications to random instances, we typically fix $w_R(N)$ to $\log_2(1/p_R(N))$ and write $\delta(Q[B])$ and $\overline{\delta}(Q)$ instead of $\delta(Q[B], w)$ and $\overline{\delta}(Q, w)$. For this choice of weights, a crucial distinction is made according to whether $\overline{\delta}$ is larger or smaller than $\log_2(N)$. In the first case, there exists subquery $Q[B]$ whose expected number of solutions is smaller than 1, and, therefore, by Markov's inequality, Q itself has no solutions at all with probability bounded away from 0. In the second case, every subquery has at least one solution in expectation, and we can bound the variance of X as a function of $\overline{\delta}$. Since this will be of use later on, we derive it in detail.

PROPOSITION 14. *If $\overline{\delta} \leq \log_2(N)$, then*

$$\text{(5.2)} \qquad \text{V}[X] \leq \text{E}[X]^2 \cdot (2^n - 1)2^{\overline{\delta} - \log_2 N}.$$

Proof. For this proof we write A instead of A_σ. For every $R \in \sigma$, let A_R be the set of attributes of R, and for every $t \in [N]^{A_R}$, let $X(R, t)$ be the indicator for the event $t \in R(D)$. These are mutually independent random variables and the expectation of $X(R, t)$ is $p_R(N)$. For every $t \in [N]^A$, let $X(t)$ be the indicator for the event $t \in Q(D)$. Note that $X(t) = \prod_{R \in \sigma} X(R, t_R)$, where t_R denotes the projection of t to the attributes of R. Also $X = \sum_t X(t)$. Toward proving (5.2), let us bound

$$\text{(5.3)} \qquad \text{E}\left[X^2\right] = \sum_{s,t} \text{E}\left[X(s)X(t)\right].$$

For every fixed $B \subseteq A$, let F_B be the set of pairs $(s, t) \in [N]^A \times [N]^A$ such that $s(a) = t(a)$ for every $a \in B$ and $s(a) \neq t(a)$ for every $a \in A - B$. Clearly, $(F_B)_{B \subseteq A}$ is a partition of $[N]^A \times [N]^A$, and therefore

$$\text{(5.4)} \qquad \sum_{s,t} \text{E}\left[X(s)X(t)\right] = \sum_{B \subseteq A} \sum_{(s,t) \in F_B} \text{E}\left[X(s)X(t)\right].$$

Now fix some $B \subseteq A$ and $(s, t) \in F_B$, and let σ_B be the relations appearing in $Q[B]$. Observe that since s and t agree on B we have $t_R = s_R$ for every $R \in \sigma_B$, and therefore $X(R, s_R)X(R, t_R) = X(R, s_R)$ for every such R. Hence
(5.5)

$$X(s)X(t) = \prod_{R \in \sigma} X(R, s_R) \prod_{R \in \sigma} X(R, t_R) = \prod_{R \in \sigma - \sigma_B} X(R, s_R)X(R, t_R) \prod_{R \in \sigma_B} X(R, s_R).$$

All variables in the right-hand side product are mutually independent because either they involve different relations or different tuples. Therefore,

$$\text{(5.6)} \qquad \text{E}\left[X(s)X(t)\right] = \prod_{R \in \sigma - \sigma_B} p_R^2 \prod_{R \in \sigma_B} p_R = \prod_{R \in \sigma} p_R^2 \prod_{R \in \sigma_B} p_R^{-1}.$$

The number of pairs (s, t) in F_B is bounded by $N^{2|A|-|B|}$. Therefore,

$$\text{(5.7)} \quad \sum_{(s,t) \in F_B} \text{E}\left[X(s)X(t)\right] \leq N^{2|A|-|B|} \prod_{R \in \sigma} p_R^2 \prod_{R \in \sigma_B} p_R^{-1} = \text{E}\left[X\right]^2 \cdot N^{-|B|} \prod_{R \in \sigma_B} p_R^{-1}.$$

For $B = \emptyset$, the second factor in the right-hand side of (5.7) is 1, and we get $\text{E}\left[X\right]^2$. For $B \neq \emptyset$, we have

$$\text{(5.8)} \qquad N^{-|B|} \prod_{R \in \sigma_B} p_R^{-1} = N^{-|B|}2^{|B| \, \delta(Q[B])} \leq \left(N^{-1}2^{\overline{\delta}}\right)^{|B|} \leq 2^{\overline{\delta} - \log_2 N},$$

where the first inequality holds because $\delta(Q[B]) \leq \overline{\delta}$, and the second inequality holds because $|B| > 0$ and $\overline{\delta} \leq \log_2(N)$. Putting it all together we get

$$\text{(5.9)} \quad \text{E}\left[X^2\right] = \sum_{B \subseteq A} \sum_{(s,t) \in F_B} \text{E}\left[X(s)X(t)\right] \leq \text{E}\left[X\right]^2 + (2^n - 1)\text{E}\left[X\right]^2 2^{\overline{\delta} - \log_2 N}.$$

Since $\text{V}\left[X\right] = \text{E}\left[X^2\right] - \text{E}\left[X\right]^2$, this proves (5.2). $\quad\square$

In the following, if X is a random variable defined on the probability space $\mathcal{D}(N, p(N))$, the expression "$X \sim x$ a.s." means that for every $\epsilon > 0$ and $\delta > 0$, there exists N_0 such that, for every $N \geq N_0$, we have $\Pr[|X - x| \leq \epsilon x] \geq 1 - \delta$. With all this notation, we obtain the following threshold behavior as an immediate consequence to Markov's and Chebyshev's inequalities.

THEOREM 15. *Let Q be a join query with schema σ and n attributes. For every $R \in \sigma$, let $p_R : \mathbb{N} \to (0, 1)$, $p(N) = (p_R(N) : R \in \sigma)$, and $\overline{\delta}(N) = \overline{\delta}(Q, w_R(N))$ for $w_R(N) = \log_2(1/p_R(N))$. Let D be drawn from $\mathcal{D}(N, p(N))$ and let X denote the size of $Q(D)$.*
1. *If $\overline{\delta}(N) = \log N - \omega(1)$, then $X \sim N^n \prod_{R \in \sigma} p_R(N)$ a.s.*
2. *If $\overline{\delta}(N) = \log N + \omega(1)$, then $X = 0$ a.s.*

Proof. For this proof we write A instead of A_σ. We start with (2). Suppose that $\overline{\delta}(N) = \log N + \omega(1)$ and fix a large N. Let $B \subseteq A$, $B \neq \emptyset$, be such that $\overline{\delta}(Q, w(N)) = \delta(Q[B], w(N))$. Let $Q_B = Q[B]$, let σ_B be the schema of Q_B, and let $M_B = |Q_B(D)|$. The expectation of M_B is

$$N^{|B|} \prod_{R \in \sigma_B} p_R(N) = 2^{|B|(\log N - |B|^{-1} \sum_{R \in \sigma_B} \log(1/p_R(N)))} = 2^{|B|(\log N - \delta(Q_B))}.$$

Since $\delta(Q_B) = \overline{\delta}(Q)$ and $|B| > 0$, the hypothesis $\overline{\delta}(N) = \log N + \omega(1)$ implies that this quantity approaches 0 as N grows. By Markov's inequality, $M_B = 0$ a.s., and therefore $M = 0$ a.s. because every solution to Q gives a solution to Q_B.

For (1) we use the bound on the variance from (14). Fix $\epsilon > 0$ and $\delta > 0$. By Chebyshev's inequality we have

$$\Pr[|X - \mathrm{E}[X]| \geq \epsilon \, \mathrm{E}[X]] \leq \frac{\mathrm{V}[X]}{\epsilon^2 \, \mathrm{E}[X]^2} \leq \frac{\mathrm{E}[X]^2 (2^n - 1) 2^{\overline{\delta} - \log_2 N}}{\epsilon^2 \, \mathrm{E}[X]^2} \leq \frac{2^n - 1}{\epsilon^2} \cdot 2^{\overline{\delta} - \log_2 N}.$$

Under the hypothesis $\overline{\delta}(N) = \log N - \omega(1)$, the right-hand side is bounded by δ for large enough N. Since $\mathrm{E}[X] = N^n \prod_{R \in \sigma} p_R(N)$, the result follows.   □

In certain applications, the concentration defined by "$X \sim x$ a.s." is not enough. For example, it may sometimes be necessary to conclude that $\Pr[|X - x| \leq \epsilon x] \geq 1 - N^{-d}$ for every $\epsilon > 0$ and $d > 0$ in order to apply a union bound that involves a number of cases that grow polynomially with N. Accordingly, for a random variable X, the expression "$X \sim x$ polynomially a.s." means that for every $\epsilon > 0$ and $d > 0$, there exists N_0 such that for every $N \geq N_0$ we have $\Pr\left[|X - x| \leq \epsilon x\right] \geq 1 - N^{-d}$.

It turns out that such a strong concentration can also be guaranteed at the expense of a wider *threshold width* in Theorem 15: instead of $\log_2(N) - \omega(1)$ versus $\log_2(N) + \omega(1)$, we require $\log_2(N) - \omega(\log \log(N))$ versus $\log_2(N) + \omega(1)$. This does not follow from Chebyshev's inequality, and for the proof we use the polynomial concentration inequality from [12].

THEOREM 16. *Let Q be a join query with schema σ and n attributes. For every $R \in \sigma$, let $p_R : \mathbb{N} \to (0, 1)$, $p(N) = (p_R(N) : R \in \sigma)$, and $\overline{\delta}(N) = \overline{\delta}(Q, w_R(N))$ for $w_R(N) = \log_2(1/p_R(N))$. Let D be drawn from $\mathcal{D}(N, p(N))$ and let X denote the size of $Q(D)$.*
1. *If $\overline{\delta}(N) = \log N - \omega(\log \log N)$, then $X \sim N^n \prod_{R \in \sigma} p_R(N)$ polynomially a.s.*
2. *If $\overline{\delta}(N) = \log N + \omega(1)$, then $X = 0$ a.s.*

For the proof, we will use the polynomial concentration method from [12]. Let $H = (V, E)$ be a hypergraph with $n = |V|$ and $k = \max_{e \in E} |e|$. For every $e \in E$, let $w(e)$ be a positive weight. Let $\{X_u : u \in V\}$ be a collection of mutually independent

random variables, where each X_u is an indicator random variable with expected value p_u. Here $0 \leq p_u \leq 1$ for every $u \in V$. Let M be the following polynomial: $M = \sum_{e \in E} w(e) \prod_{u \in e} X_u$. For every $Y \subseteq V$, let M_Y be the partial derivative of M with respect to $\{X_u : u \in Y\}$, that is, $M_Y = \sum_{e \in E: Y \subseteq e} w(e) \prod_{u \in e \setminus Y} X_u$. Let $E_Y = \mathrm{E}[M_Y]$, and for every $i \in \{0, \dots, k\}$, let $E_i = \max\{E_Y : Y \subseteq V, |Y| = i\}$. Note that $E_0 = \mathrm{E}[M]$. Let $E' = \max\{E_i : 1 \leq i \leq k\}$ and $E = \max\{E_0, E'\}$.

THEOREM 17 (Theorem 7.8.1 in [2]). *For every $\lambda > 1$, it holds that*

$$\Pr\left[|M - \mathrm{E}[M]| > a_k (EE')^{1/2} \lambda^k\right] < d_k e^{-\lambda} n^{k-1},$$

where $a_k = 8^k \sqrt{k!}$ and $d_k = 2e^2$.

For a complete proof of Theorem 16, it is easier to first state the following consequence of Theorem 17 (see also Corollary 4.1.3 in [12]).

COROLLARY 18. *For every two reals $\epsilon > 0$ and $d > 0$ and every integer $k \geq 1$, there exists n_0 such that, in the setting of Theorem 17, if $n \geq n_0$ and $E_i/E_0 \leq (\log n)^{-4k}$ for every $i \in \{1, \dots, k\}$, then*

$$\Pr\left[|M - \mathrm{E}[M]| > \epsilon \, \mathrm{E}[M]\right] \leq n^{-d}.$$

For a proof, it suffices to choose $\lambda(n) = (\epsilon/a_k)^{1/k}(\log n)^2$ and n_0 such that $\lambda > 1$ and $d_k e^{-\lambda} n^{k-1} < n^{-d}$ for every $n \geq n_0$. Note that if $E_i/E_0 \leq (\log n)^{-4k}$, then $E = E_0$ and $E' \leq E_0 (\log n)^{-4k}$, so $a_k (EE')^{1/2} \lambda^k \leq \epsilon E_0$ and the claim follows.

Proof of Theorem 16. The proof of (2) is the same as in Theorem 15. For (1), suppose that $\overline{\delta}(N) = \log N - \omega(\log \log N)$ and fix a large N. For every $R \in \sigma$, let A_R be the set of attributes of R, and for every $t \in [N]^{A_R}$, let $X(R, t)$ be the indicator random variable for the event $t \in R(D)$. Note that these are mutually independent random variables and $\mathrm{E}[X(R, t)] = p_R(N)$ by the definition of the probability space. Note also that

$$M = \sum_t \prod_{R \in \sigma} X(R, t_R),$$

where t ranges over all tuples in $[N]^A$, and t_R denotes the projection of t to the attributes of R. We aim for an application of Corollary 18 with the random variables $X(R, t)$. We define the hypergraph $H = (V, E)$. The set of vertices V is the set of pairs (R, t), where $R \in \sigma$ and $t \in [N]^{A_R}$. There is one hyperedge e_t in E for every $t \in [N]^A$ that consists of all pairs (R, t_R) with $R \in \sigma$. Thus, the number of vertices is bounded by mN^r, where $m = |\sigma|$ and r is the maximum arity of the relations in σ. Furthermore, the maximum size of the edges in H, that is, the k in Corollary 18, is m.

We have

(5.10) $$E_0 = N^n \prod_{R \in \sigma} p_R(N).$$

Let us bound E_i for $i > 0$. Fix a set of vertices of the hypergraph H, say,

$$Y \subseteq \{(R, t) : R \in \sigma, t \in [N]^{r_R}\},$$

with $|Y| = i > 0$. We want to bound E_Y/E_0. Let σ' be the set of relations R that appear in Y. Let B be all the attributes of the relations in σ'. Note that $\sigma' \subseteq \sigma_B$, where σ_B is the schema of $Q[B]$. This will be of use later.

Let T be the set of all $t \in [N]^A$ such that $(R, t_R) \in Y$ for every $R \in \sigma'$, where, as before, t_R denotes the projection of t to the attributes of R. If there exist t_1 and t_2 in T that disagree on some attribute of B, then automatically $E_Y = 0$ because then Y is not included in any hyperedge e_t of H and hence $M_Y = 0$. We may assume then that all $t \in T$ agree on B. This implies $|T| \leq N^{n-|B|}$. Under these conditions we have

$$E_Y = \mathrm{E}\left[\sum_{t \in T} \prod_{R \in \sigma - \sigma'} X(R, t_R)\right] \leq N^{n-|B|} \prod_{R \in \sigma - \sigma'} p_R(N).$$

Therefore, recalling (5.10), we bound E_Y / E_0 by
(5.11)
$$N^{-|B|} \prod_{R \in \sigma'} \frac{1}{p_R(N)} \leq N^{-|B|} \prod_{R \in \sigma_B} \frac{1}{p_R(N)} \leq N^{-|B|\left(1 - \frac{1}{\log N}|B|^{-1} \sum_{R \in \sigma_B} \log(1/p_R(N))\right)},$$

where the first inequality holds because $\sigma' \subseteq \sigma_B$ and each p_R belongs to $(0, 1)$. Using the hypothesis that $\overline{\delta}(N) = \log N - \omega(\log \log N)$, we bound (5.11) by $(\log N)^{-(4m+1)}$. We showed then that

$$E_Y / E_0 \leq (\log N)^{-(4m+1)},$$

and since this holds for an arbitrary Y of size $i > 0$, the bound is also valid for E_i / E_0. Recall now that the number of vertices $|V|$ of the hypergraph H is at most mN^r, and we can bound

$$(\log N)^{-(4m+1)} \leq (\log(mN^r))^{-4m} \leq (\log |V|)^{-4m}$$

for large N. The result follows from Corollary 18.   □

We conclude this section with a max-flow construction to compute $\overline{\delta}(Q, w)$. This is probably folklore, but as the proof is short we include it anyway for the reader's convenience. For every real number $\delta > 0$, we build a network $N(\delta)$ as follows. The network has a source s, a target t, and $|A_\sigma| + |\sigma|$ intermediate nodes. There is a link of capacity δ between s and each $a \in A_\sigma$. Each $a \in A_\sigma$ has a link of infinite capacity to each $R \in \sigma$ with $a \in A_R$. Finally, each $R \in \sigma$ is linked to t with capacity w_R. Recall that a cut in the network is a set of links that disconnects the target from the source. The capacity of the cut is the sum of the capacities of the links in it. Let $\gamma(Q, w, \delta)$ be the minimum capacity of all cuts of $N(\delta)$.

LEMMA 19. *The following are equivalent:*
1. $\gamma(Q, w, \delta) < \sum_{R \in \sigma} w_R$.
2. $\overline{\delta}(Q, w) > \delta$.

Proof. Assume $\overline{\delta}(Q, w) > \delta$ and let $B \subseteq A_\sigma$, $B \neq \emptyset$, be such that $\delta(Q[B], w) > \delta$. Let σ_B be the schema of $Q[B]$. Let S be the cut that consists of all links from the source to the nodes of B and the links from the nodes in $\sigma - \sigma_B$ to the target. The capacity of this cut is

$$|B|\delta + \sum_{R \in \sigma - \sigma_B} w_R < \sum_{R \in \sigma_B} w_R + \sum_{R \in \sigma - \sigma_B} w_R = \sum_{R \in \sigma} w_R,$$

where the inequality follows from $\delta(Q[B], w) > \delta$.

Suppose now $\gamma(Q, w, \delta) < \sum_R w_R$ and let S be a cut of minimum capacity. Let B be the set of $a \in A_\sigma$ for which the link from the source to a is in S. Let σ_B be the schema of $Q[B]$. We claim that S does not contain any link from an $R \in \sigma_B$ to the target. For if it did, $S - \{(R, t)\}$ would also be a cut of smaller capacity. Also S contains all links from an $R \in \sigma - \sigma_B$ to t. For if it did not, S would not be a cut. (We assume all R have a nonempty set of attributes.) Finally, S does not contain any link from an $a \in A_\sigma$ to an $R \in \sigma$ because those have infinite capacity. Therefore, the capacity of S is $\delta|B| + \sum_{R \in \sigma - \sigma_B} w_R$ and smaller than $\sum_{R \in \sigma} w_R$ by hypothesis. Hence $B \neq \emptyset$ and $\delta(Q[B], w) > \delta$. $\square$

By the max-flow min-cut algorithm, it follows that $\overline{\delta}(Q, w)$ is computable in polynomial time.

5.2. Execution plans. Theorem 7 shows that certain queries admit a join-project plan that cannot be converted into a join plan without causing a superpolynomial increase in the worst-case running time. The following result shows that when we are considering average-case running time in the random database model, projections may be eliminated at a very small expected cost.

THEOREM 20. *Let Q be a join query with schema σ and n attributes. Let φ be a join-project plan for Q. For every $R \in \sigma$, let $p_R : \mathbb{N} \to (0, 1)$, $p(N) = (p_R(N) : R \in \sigma)$ and let D_N be drawn from $\mathcal{D}(N, p(N))$. There exists a constant c_φ depending only φ, such that for every large enough N and every T, if $E[|\psi(D_N)|] \leq T$ for every subplan ψ of φ, then there is a join plan φ_N^* for Q such that $E[|\psi(D_N)|] \leq c_\varphi T$ for every subplan ψ of φ_N^*.*

That is, informally speaking, for every join-project plan there is a join plan that is "almost as good." Note that in Theorem 20, the join plan depends on N. This is an unavoidable artifact of our model: the probabilities $p_R(N)$ can be completely different for different values of N, and hence it is unavoidable that different plans could be needed for different N.

The join plan φ^* is obtained by iteratively using a procedure that is capable of reducing the number of projections by one in such a way that the expected size of each subplan increases only by a factor depending only on the query. In each iteration, the procedure selects a subplan $\pi_A(\varphi_0)$ of φ such that φ_0 contains no projections, i.e., this projection π_A is lowest in the tree representation of φ. In the first step of the procedure, we replace φ_0 with a join plan φ_0' that contains only those relations appearing in φ_0 whose attributes are completely contained in A^*, where $A^* \supseteq A$ is an appropriate set of attributes from A_σ. In the second step, the projection π_A is removed (or, in other words, π_A is replaced by π_{A^*}, making it redundant). The key step of the algorithm is choosing the right A^*. If A^* is too small, then φ_0' is much less restrictive than φ_0, and hence $|\pi_A(\varphi_0'(D))|$ can be much larger than $|\pi_A(\varphi_0(D))|$. On the other hand, if A^* is too large, then $|\pi_{A^*}(\varphi_0'(D))|$ can be much larger than $|\pi_A(\varphi_0'(D))|$. The algorithm carefully balances the size of A^* between these two opposing constraints; the choice of A^* is based on the minimization of a submodular function defined below.

Let $S \subseteq \sigma$ be a set of relations over the attributes A_σ and denote by A_R the attributes of a relation R. For a subset $A \subseteq A_\sigma$, let

$$f_S(A) := |A|(\log N - n - 1) - \sum_{R \in S[A]} w_R(N),$$

where the set $S[A]$ contains those relations $R \in S$ whose attributes are contained in

A. It is easy to see that $f_S(A)$ is submodular, i.e.,

$$f_S(A) + f_S(B) \geq f_S(A \cup B) + f_S(A \cap B)$$

for every $A, B \subseteq A_\sigma$. It follows that A has a unique minimum-value extension.

PROPOSITION 21. *For every $A \subseteq A_\sigma$ and $S \subseteq \sigma$, there is a unique $C_S(A) \supseteq A$ such that $f_S(C_S(A))$ is minimal and, among such sets, $|C_S(A)|$ is maximal.*

Proof. Suppose that there are two such sets B and C with this property. By the minimality of $f_S(B) = f_S(C)$, we have $f_S(B \cup C) \geq f_S(B)$ and $f_S(B \cap C) \geq f_s(C)$. Furthermore, as $|B \cap C| < |C|$, the maximality of $|C|$ among the sets minimizing f_S ensures that $f_S(B \cap C)$ is strictly greater than $f_S(C)$. It follows that $f_S(B \cup C) + f_S(B \cap C) > f_S(B) + f_S(C)$, violating the submodularity of f_S. □

We prove Theorem 20 by presenting an algorithm that iteratively removes projections. We describe the algorithm below, and then we prove in sections 5.2.1–5.2.3 that the algorithm transforms the execution plan the required way.

The algorithm. First, we can assume that φ is of the form $(((\varphi' \bowtie R_1) \bowtie R_2) \bowtie \ldots \bowtie R_{|\sigma|})$, where $R_1, \ldots, R_{|\sigma|}$ is an ordering of the relations in σ: if φ' is a join-project plan for the query Q, then joining any relation R_i with φ' does not change φ'. However, this assumption will ensure that if we make any changes in φ', then φ will remain a join-project plan for the query Q. Furthermore, we assume that for every attribute $a \in A_\sigma$, there is a dummy unary relation R_a with $p_{R_a}(N) = 1$ and hence $w_{R_a}(N) = 0$. (We can always remove joins with these dummy relations from our final join plan without increasing the size of any intermediate joins.)

The two steps described below reduce the number of projections in φ in such a way that the maximum expected size of a subplan is at most a constant factor larger in the new plan than in φ (with a constant depending only on φ). This procedure is repeated as many times as the number of projections, thus the total increase of the maximum expected size is only a constant c_φ.

Let $\pi_A(\varphi_0)$ be a subplan of φ such that φ_0 does not contain any projections. Let $S \subseteq \sigma$ be the relation names appearing in φ_0, which means that $\varphi_0(D) = \bowtie_{R \in S} R(D)$. Let $A^* = C_S(A)$.

- Step 1 (removing joins). If a relation $R \in S \setminus S[A^*]$ appears in φ_0, then R is removed from φ_0. For the sake of analysis, we implement the removal by replacing the relation R in φ_0 with the 0-ary relation. (Note that joining the 0-ary relation with any relation R' gives exactly R'.) This way, it will be clear that there is a correspondence between the subplans of the original and modified execution plans. We can assume that for every $a \in A^*$, the unary relation R_a appears in φ_0. (Such relations can be joined with φ_0 without increasing the number of tuples.) This ensures that $A^* \subseteq A_{\varphi_0'}$. Replacing subplan φ_0 of φ with φ_0' gives a new join-project plan φ'. By our initial assumption on the structure of φ (that all the relations are joined just below the root), φ' is also a join-project plan for Q.
- Step 2 (removing projections). In the second step of the procedure, we obtain a join-project plan φ'' from φ' by replacing the subplan $\pi_A(\varphi_0')$ with $\pi_{A^*}(\varphi_0')$. (This makes the projection redundant, and it can be eliminated, but it is more convenient to analyze the step this way, since this step does not change the structure of the join-project plan.) Note that this change does not have any effect on the size of $\psi'(D)$ for any subplan ψ' of φ_0'.

See Figure 1 for an example. In sections 5.2.2 and 5.2.3, we show that the two steps increase the expected size of the query only by a constant factor. In section 5.2.1, we review and introduce the probabilistic tools that are required for this analysis.

$$R_1 : \{a, c\}$$
$$R_2 : \{b, c\}$$
$$R_3 : \{c, e\}$$
$$R_4 : \{b, d\}$$
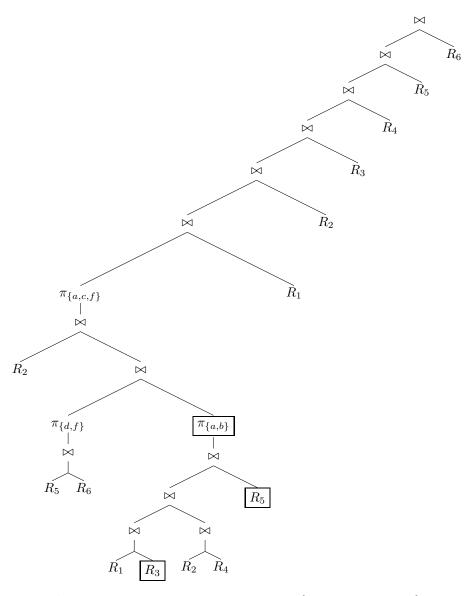$$R_5 : \{d, f\}$$
$$R_6 : \{e, f\}$$

FIG. 1. *A join-project plan over a schema of six relations $\{R_1, R_2, R_3, R_4, R_5, R_6\}$, represented as a binary tree. We demonstrate the removal of a projection from the join-project plan in the proof of Theorem 20. Consider the two steps when removing the projection $\pi_{\{a,b\}}$. Let us assume that $C_S(\{a,b\}) = \{a, b, c, d\}$. In Step 1, the two framed relations R_3 and R_5 are removed. In Step 2, $\pi_{\{a,b\}}$ is replaced by $\pi_{\{a,b,c,d\}}$.*

5.2.1. Probabilistic tools. The Fortuin–Kasteleyn–Ginibre (FKG) inequality is a general tool for determining the correlation between monotone (antimonotone) events.

FACT 22. *Let V be a finite set, and let $f, g : \{0,1\}^V \to \mathbb{R}$ be monotone functions on these variables. Let $\mu : \{0,1\}^V \to \mathbb{R}^+$ be a function satisfying $\mu(x)\mu(y) \leq \mu(x \vee y)\mu(x \wedge y)$ for every $x, y \in \{0,1\}^V$ (where $x \vee y$ and $x \wedge y$ denote the coordinatewise disjunction and conjunction of the two tuples, respectively.) Then*

$$\left(\sum_x f(x)g(x)\mu(x) \right) \left(\sum_y \mu(y) \right) \geq \left(\sum_x f(x)\mu(x) \right) \left(\sum_y g(y)\mu(y) \right),$$

where in all four cases the sums range over $\{0,1\}^V$.

A proof can be found in [2].

In general, if f, g, h are three monotone 0-1 functions of a set of independent 0-1 random variables, then (somewhat counterintuitively) it is not necessarily true that $\Pr[f = 1 \mid g = 1] \leq \Pr[f = 1 \mid gh = 1]$. That is, the condition that a more restrictive monotone function (gh instead of g) is 1 does not necessarily increase the probability that the monotone function is 1. As an example, suppose that x_1, x_2, x_3 are independent 0-1 random variables, each having probability $1/2$ of being 1. Let

$$f = (x_1 \wedge x_2) \vee (x_2 \wedge x_3) \vee (x_1 \wedge x_3),$$
$$g = (x_1 \wedge x_2) \vee x_3,$$
$$h = x_3.$$

Now $\Pr[f = 1 \mid g = 1] = 4/5$, but $\Pr[f = 1 \mid gh = 1] = \Pr[f = 1 \mid h = 1] = 3/4$. However, the statement is true in the special case when the g and h are products of the random variables:

LEMMA 23. *Let V be a set of independent 0-1 random variables, let $M' \subseteq M \subseteq V$ be two subsets of these variables, and let $f : \{0,1\}^V \to \{0,1\}$ be a monotone function. Then*

$$\Pr[f = 1 \mid \textstyle\prod M = 1] \geq \Pr[f = 1 \mid \textstyle\prod M' = 1].$$

Proof. For every $x \in \{0,1\}^V$, if every variable of M' is 1 in x, then let $\mu(x)$ be the probability of tuple x, otherwise let $\mu(x) = 0$. It is easy to verify that $\mu(x)\mu(y) \leq \mu(x \vee y)\mu(x \wedge y)$ for every $x, y \in \{0,1\}^V$. With f and $g = \prod(M \setminus M')$, Theorem 22 implies

$$\Pr[f = 1 \wedge \textstyle\prod M = 1] \cdot \Pr[\textstyle\prod M' = 1] \geq \Pr[f = 1 \wedge \textstyle\prod M' = 1] \cdot \Pr[\textstyle\prod M = 1],$$

which, rewritten, is

$$\Pr[f = 1 \mid \textstyle\prod M = 1] \geq \Pr[f = 1 \mid \textstyle\prod M' = 1],$$

which is what we had to show. $\quad\square$

The choice of the random database D can be thought of as a set of independent 0-1 random variables. (N^r variables describe an r-ary random relation.) For a join-project plan φ and a tuple t, let $I_{\varphi(D),t}$ be the indicator random variable that is 1 if and only if $t \in \varphi(D)$; clearly $I_{\varphi(D),t}$ is a monotone function of the random variables describing the database. Since this function is monotone, it can be expressed as the disjunction of minterms, i.e., $I_{\varphi(D),t} = \bigvee_{i=1}^{M} I_{\varphi(D),t}^{(i)}$, where each minterm $I_{\varphi(D),t}^{(i)}$ is

the product of a subset of the 0-1 random variables. We say that the *rank* of a monotone function is the maximum size of a minterm of the function. The following two lemmas will be useful for determining conditional probabilities between these monotone functions.

LEMMA 24. *Let φ be a join-project plan whose tree has ℓ leaves and let t be a tuple in* $\mathrm{tup}(\varphi)$.

1. *The rank of $I_{\varphi(D),t}$ is at most ℓ.*
2. *If φ_0 is a subplan of φ, then $I_{\varphi(D),t}$ can be written as $\bigvee_{t'\in\mathrm{tup}(\varphi_0)}(I_{\varphi_0(D),t'} \wedge J_{t'})$, where each $J_{t'}$ is a monotone function of the random variables. Moreover, if φ' is obtained from φ by replacing φ_0 with some other subplan φ_0' satisfying $A_{\varphi_0} = A_{\varphi_0'}$, then $I_{\varphi'(D),t} = \bigvee_{t'\in\mathrm{tup}(\varphi_0')}(I_{\varphi_0'(D),t'} \wedge J_{t'})$ with the same functions $J_{t'}$.*

Proof. Statement 1 can be proved by a simple induction on the size of the tree of φ. First, observe that the rank of disjunction of functions is at most the maximum of the ranks of the functions, while the rank of the conjunction of functions is at most the sum of the ranks of the functions. If the tree consists of a single leaf (i.e., φ consists of a single relation symbol), then $I_{\varphi(D),t}$ is equal to one of the random variables, i.e., its rank is 1. If $\varphi = \pi_X(\varphi^*)$, then $I_{\varphi(D),t} = \bigvee_{t'\in\mathrm{tup}(\varphi^*),\pi_X(t')=\pi_X(t)} I_{\varphi^*(D),t'}$. By induction, the rank of each $I_{\varphi^*(D),t'}$ is at most ℓ, and hence the rank of this disjunction is also at most ℓ. Finally, if $\varphi = \varphi_1 \bowtie \varphi_2$, then $I_{\varphi(D),t} = I_{\varphi_1(D),\pi_{A_{\varphi_1}}(t)} \wedge I_{\varphi_2(D),\pi_{A_{\varphi_2}}(t)}$. The number ℓ of leaves of φ is the sum of the number of leaves of φ_1 and φ_2, and hence the rank of this conjunction is at most the number of leaves of φ.

To prove statement 2, we build the function $I_{\varphi(D),t}$ using disjunctions and conjunctions as in the previous paragraph, but the functions $I_{\varphi_0(D),t'}$ are not decomposed any further. This way, $I_{\varphi(D),t}$ is expressed as a monotone function of the random variables and of the functions $I_{\varphi_0(D),t'}$ ($t' \in \mathrm{tup}(\varphi_0)$). Thus $I_{\varphi(D),t}$ can be written in the required form, and it is clear that $J_{t'}$ does not depend on the structure of the subplan φ_0. $\quad\square$

Let f and g be two functions on a set of independent 0-1 variables. Suppose that for each minterm of f, the probability that $g = 1$ on the condition that the minterm is 1 is at least p. Somewhat counterintuitively, this does not necessarily mean that the probability of $g = 1$ on the condition that $f = 1$ is at least p. For example, consider two independent variables x_1 and x_2 with probability $1/2$ of being 1, and let $g = x_1 \wedge x_2$ and $f = x_1 \vee x_2$. Now $\Pr[g = 1 \mid f = 1] = 1/3$, but $\Pr[g = 1 \mid x_1 = 1] = \Pr[g = 1 \mid x_2 = 1] = 1/2$, i.e., the probability is larger conditioned on either minterm of f than on f itself. The following lemma shows that if the functions have bounded rank, then we can bound the ratio of these conditional probabilities.

LEMMA 25. *Let f, g be two monotone 0-1 functions of rank at most ℓ on a set V of independent 0-1 random variables.*

1. *If the probability of 1 for each random variable is decreased by at most a factor $c > 1$, then the expected value of f is decreased by at most a factor c^ℓ.*
2. *If $\Pr[g = 1 \mid f^{(i)} = 1] \geq p$ for every minterm $f^{(i)}$ of f, then $\Pr[g = 1 \mid f = 1] \geq p \cdot 2^{-2\ell}$.*

Proof. To prove statement 1, let $x_1 : V \to \{0,1\}$ be a random assignment of the variables chosen according to the original probabilities, and let x_2 be a the independent random assignment, where $x_2(v) = 1$ with probability $1/c$ uniformly and independently for each v. Let m be the number of minterms of f and for $1 \leq i \leq m$, let $f^{(i)}$ be the ith minterm of f. Let X_i be the event that $f^{(i)}(x_1) = 1$ and $f^{(j)}(x_1) = 0$

for every $j < i$, i.e., the ith minterm is the first satisfied minterm of f. The events X_i are disjoint, thus $\Pr[f(x_1) = 1] = \sum_{i=1}^{m} \Pr[X_i]$. Let Y_i be the event that $f^{(i)}(x_2) = 1$. Clearly, X_i and Y_i are independent and $\Pr[Y_i] \geq c^{-\ell}$. Let $x_1 \wedge x_2$ be the conjunction of x_1 and x_2; observe that in $x_1 \wedge x_2$, the probability of a variable being 1 is exactly $1/c$ times the original probability. Each event $X_i \wedge Y_i$ implies $f(x_1 \wedge x_2) = 1$ (as it implies $f^{(i)}(x_1 \wedge x_2) = 1$) and these events are are disjoint (since the events X_i are already disjoint). Therefore,

$$\Pr[f(x_1 \wedge x_2) = 1] \geq \sum_{i=1}^{m} \Pr[X_i \wedge Y_i] = \sum_{i=1}^{m} \Pr[X_i]\Pr[Y_i] \geq \Pr[f(x_1) = 1] \cdot c^{-\ell},$$

which is what we had to show.

For statement 2, let x_1, x_2 be two independent random assignments, chosen according to the original probabilities of the random variables. Let X_i be the event as above and let Z_i be the event that $g(x_2 \vee m_i) = 1$, where m_i is the assignment that sets the ith minterm of f to 1 and every other random variable to 0. Clearly, X_i and Z_i are independent. Observe that $\Pr[Z_i] = \Pr[g(x_2) = 1 \mid f^{(i)}(x_2) = 1] \geq p$, since Z_i depends only on what x_2 assigns to the variables not in the ith minterm.

In order to bound the conditional probability $\Pr[g(x_1) = 1 \mid f(x_1) = 1]$, we need to bound the probability $\Pr[g(x_1) = 1 \wedge f(x_1) = 1]$. To bound this probability, we calculate $\Pr[g(x_1 \vee x_2) = 1 \wedge f(x_1 \vee x_2) = 1]$ (where $x_1 \vee x_2$ is the assignment defined as the disjunction of x_1 and x_2). Observe that for each variable v, $\Pr[(x_1 \vee x_2)(v) = 1] \leq 2\Pr[x_1(v) = 1]$. Thus, by statement 1, $\Pr[g(x_1 \vee x_2) = 1 \wedge f(x_1 \vee x_2) = 1] \leq 2^{2\ell}\Pr[g(x_1) = 1 \wedge f(x_1) = 1]$. (Here we used that the rank of the conjunction of two rank ℓ functions is at most 2ℓ.) Now we can bound the conditional probability as follows: Start at

$$\Pr[g(x_1) = 1 \mid f(x_1) = 1] = \frac{\Pr[f(x_1) = 1 \wedge g(x_1) = 1]}{\Pr[f(x_1) = 1]},$$

which by statement 1 is at least

$$\frac{2^{-2\ell}\Pr[f(x_1 \vee x_2) = 1 \wedge g(x_1 \vee x_2) = 1]}{\Pr[f(x_1) = 1]}.$$

Since $f(x_1) = 1$ implies $f(x_1 \vee x_2) = 1$ and the event $f(x_1) = 1$ is the union of the disjoint events X_i's, this is at least

$$\frac{2^{-2\ell}\sum_{i=1}^{m}\Pr[X_i \wedge g(x_1 \vee x_2) = 1]}{\Pr[f(x_1) = 1]}.$$

Since $X_i \wedge Z_i$ implies $g(x_1 \vee x_2) = 1$ and X_i and Z_i are independent, this is at least

$$\frac{2^{-2\ell}\sum_{i=1}^{m}\Pr[X_i]\Pr[Z_i]}{\Pr[f(x_1) = 1]} \geq \frac{2^{-2\ell} \cdot \Pr[f(x_1) = 1] \cdot p}{\Pr[f(x_1) = 1]} = 2^{-2\ell}p. \qquad \square$$

5.2.2. Analysis of Step 1. The analysis of Step 1 relies on the following lemma, which shows that we do not get many additional tuples if we take the join of only those relations whose attributes are in $C_S(A)$.

LEMMA 26. *Let $S \subseteq \sigma$ be a set of relation names and $A \subseteq A_\sigma$ a set of attributes. Let $A^* = C_S(A)$ and let $S^* = S[A^*]$. For every A^*-tuple t^*,*

$$\Pr[t^* \in \pi_{A^*}(\bowtie_{R \in S} R(D)) \mid t^* \in \bowtie_{R \in S^*} R(D)] \geq 1/2.$$

Proof. By definition, $t^* \in \pi_{A^*}(\bowtie_{R \in S} R(D))$ if and only if there is a $t \in \bowtie_{R \in S} R(D)$ with $\pi_{A^*}(t) = t^*$. Note that if $t^* \in \bowtie_{R \in S^*} R(D)$ and $\pi_{A^*}(t) = t^*$, then t satisfies all the relations in S^*, and hence the probability that such a t is in $\bowtie_{R \in S} R(D)$ (assuming $t^* \in \bowtie_{R \in S^*} R(D)$) depends only on the relations in $S \setminus S^*$. We claim that this conditional probability is equal to the probability that a certain query Q' with schema σ' has at least one solution. The query Q' is over the attributes $A_\sigma \setminus A^*$. The schema σ' contains a relational symbol R' for each $R \in S \setminus S^*$; the set of attributes of R' is $A_R \setminus X^*$. We define the probability of placing a tuple into R' as $p_{R'}(N) = p_R(N)$ for every $R' \in \sigma'$. It is not difficult to see that $\Pr[t^* \in \pi_{A^*}(\bowtie_{R \in S} R(D)) \mid t^* \in \bowtie_{R \in S^*} R(D)]$ is equal to the probability that Q' has at least one solution.

Observe that if A' is a subset of the attributes in Q', then the relations in $\sigma'[A']$ were obtained from the relations in $S[A^* \cup A'] \setminus S[A^*]$, which means that the weight of these relations is counted in $f_S(A^* \cup A')$ but not in $f_S(A^*)$. If the weight of the relations in $\sigma'[A']$ is greater than $|A'|(\log N - n - 1)$, then $f_S(A^* \cup A') < f_S(A^*)$ would follow, contradicting the minimality of $f_S(A^*) = f_S(C_S(A))$. This means that the maximum density $\overline{\delta}$ of Q' is at most $\log N - n - 1$. Writing $X := |Q'(D)|$, by Proposition 14 the variance of X is

$$\mathrm{V}[X] \le \mathrm{E}[X]^2 \cdot (2^{n'} - 1)2^{\overline{\delta} - \log N} < \mathrm{E}[X]^2 \cdot (2^{n'} - 1)2^{-(n+1)} \le \mathrm{E}[X]^2/2.$$

Therefore, by Chebyshev's inequality, the probability that there is no solution can be bounded as

$$\Pr[X = 0] \le \Pr[|X - \mathrm{E}[X]| \ge \mathrm{E}[X]] \le \mathrm{V}[X]/\mathrm{E}[X]^2 \le 1/2. \qquad \square$$

Let φ' be the new join-project plan obtained from φ by an application of Step 1. For every subplan ψ' of φ', there is a corresponding subplan ψ of φ. We claim that

$$\mathrm{E}[|\psi'(D)|] \le 2^{2\ell+1}\, \mathrm{E}[|\psi(D)|],$$

where ℓ is the number of leaves in ψ. If subplan ψ is disjoint from φ_0, then ψ' and ψ are the same, and we are done (e.g., if ψ is the subplan rooted at $\pi_{\{d,f\}}$ in Figure 1). Thus we have to consider only two cases: ψ is either completely contained in φ_0, or ψ contains φ_0.

Case 1. ψ is contained in φ_0 (e.g., ψ is one of the join nodes below $\pi_{\{a,b\}}$ in Figure 1). Let B be the set of all attributes of the relations appearing in ψ. If $B \subseteq A^*$, then the attributes of each relation appearing in ψ are fully contained in A^* and we are done: $\psi' = \psi$. Otherwise, let w (resp., w') be the total weight of the relations appearing in ψ (resp., ψ'). Observe that $w - w' \le |B \setminus A^*|(\log N - n - 1)$: otherwise we would have $f_S(A^* \cup B) < f_S(A^*)$, contradicting the minimality of A^*. Clearly, $A_{\psi'} \subseteq B \cap A^*$. Thus the expected size of $|\psi'(D)|$ is

$$\mathrm{E}[|\psi'(D)|] = 2^{|A_{\psi'}| \log N - w'} \le 2^{|B \cap A^*| \log N - w' + |B \setminus A^*|(\log N - n - 1) - (w - w')}$$
$$\le 2^{|B| \log N - w} = \mathrm{E}[|\psi(D)|].$$

Case 2. ψ is not contained in φ_0, which implies that ψ contains $\pi_A(\varphi_0)$ as subplan (including the possibility that $\psi' = \pi_A(\varphi_0)$. As an example, consider the projection $\pi_{\{a,c,f\}}$ in Figure 1. Note that $A_\psi = A_{\psi'}$ (because we are above the projection π_A) and $\psi(D) \subseteq \psi'(D)$ follows from $\pi_A(\varphi_0(D)) \subseteq \pi_A(\varphi_0'(D))$. Let ℓ be the number of leaves of ψ. We claim that for every tuple $t \in \mathrm{tup}(\psi')$ we have

$\Pr[t \in \psi'(D)] \leq 2^{2\ell+1} \Pr[t \in \psi(D)]$, which implies $\mathrm{E}[|\psi'(D)|] \leq 2^{2\ell+1} \mathrm{E}[|\psi(D)|]$. To prove this, we show that for every minterm $I^{(i)}_{\psi'(D),t}$ of $I_{\psi'(D),t}$, we have

$$(5.12) \qquad \Pr[I_{\psi(D),t} = 1 \mid I^{(i)}_{\psi'(D),t} = 1] \geq 1/2.$$

Thus by Lemma 25.2, we get $\Pr[I_{\psi(D),t} = 1 \mid I_{\psi'(D),t} = 1] \geq 1/2^{2\ell+1}$, which is what we need.

By Lemma 24.2, $I_{\psi'(D),t}$ can be written as

$$(5.13) \qquad I_{\psi'(D),t} = \bigvee_{t' \in \pi_A(\mathrm{tup}(\varphi'_0))} (I_{\pi_A(\varphi'_0(D)),t'} \wedge J_{t'}).$$

Consider a particular minterm $I^{(i)}_{\psi'(D),t}$ for some i, which is the product of a subset V_i of the random variables. If $\prod V_i = 1$, then $I_{\psi'(D),t} = 1$, implying that $I_{\pi_A(\varphi'_0(D)),t'} \wedge J_{t'} = 1$ for some tuple $t' \in \mathrm{tup}(\pi_A(\varphi'_0))$, which further implies that there is a tuple $t'' \in \mathrm{tup}(\varphi'_0)$ such that $\pi_A(t'') = t'$ and $I_{\varphi'_0(D),t''} = 1$. Let us fix such a t' and t''. Since φ'_0 does not contain projections, $I_{\varphi'_0(D),t''}$ is the product of a set V'_i of random variables. As $I^{(i)}_{\psi'(D),t} = 1$ implies $I_{\varphi'_0(D),t''} = 1$, we have $V'_i \subseteq V_i$. By a consequence of the FKG inequality (see Lemma 23),

$$(5.14) \qquad \Pr[I_{\pi_A(\varphi_0(D)),t'} = 1 \mid \prod V_i = 1] \geq \Pr[I_{\pi_A(\varphi_0(D)),t'} = 1 \mid \prod V'_i = 1]$$

(since $V'_i \subseteq V_i$ and $I_{\pi_A(\varphi_0(D)),t'}$ is monotone). Note that it is φ_0 and not φ'_0 that appears in (5.14). By Lemma 24.2, $I_{\psi(D),t}$ can be also written in the form (5.13), and hence $I_{\pi_A(\varphi_0(D)),t'} \wedge J_{t'} = 1$ implies $I_{\psi(D),t} = 1$. Thus we have

$$\begin{aligned}
\Pr[I_{\psi(D),t} = 1 \mid \prod V_i = 1] &\geq \Pr[I_{\pi_A(\varphi_0(D)),t'} = 1 \wedge J_{t'} = 1 \mid \prod V_i = 1] \\
&= \Pr[I_{\pi_A(\varphi_0(D)),t'} = 1 \mid \prod V_i = 1] \\
&\geq \Pr[I_{\pi_A(\varphi_0(D)),t'} = 1 \mid \prod V'_i = 1] \\
&\geq \Pr[I_{\pi_{A^*}(\varphi_0(D)),t''} = 1 \mid \prod V'_i = 1] \\
&= \Pr[t'' \in \pi_{A^*}(\varphi_0(D)) \mid t'' \in \varphi'_0(D)] \\
&\geq 1/2,
\end{aligned}$$

which is what we had to show. (The first inequality follows from the fact that $I_{\pi_A(\varphi_0(D)),t'} \wedge J_{t'} = 1$ implies $I_{\psi(D),t} = 1$; the equality after that from the fact that $\prod V_i = 1$ implies $J_{t'} = 1$; the second inequality follows from (5.14); the third inequality follows from $\pi_A(t'') = t'$; the last inequality follows from Lemma 26.)

5.2.3. Analysis of Step 2. The analysis of Step 2 relies on the following lemma, which shows that extending the projection from A to A^* does not increase the number of tuples too much: a tuple in the projection to A does not have too many extensions to A^*.

LEMMA 27. *Let $S \subseteq \sigma$ be a set of relation names and $A \subseteq A_\sigma$ be a set of attributes. Let $A^* = C_S(A)$ and let $S^* = S[A^*]$. For an A-tuple t, let L_t be the set of those A^*-tuples $t^* \in \bowtie_{R \in S^*} R(D)$ that have $\pi_A(t^*) = t$.*
  1. *For every A-tuple t we have $\mathrm{E}[|L_t| \mid L_t \neq \emptyset] \leq 2^{n(n+2)}$.*
  2. *For every A-tuple t and A^*-tuple t^* with $\pi_A(t^*) = t$ we have $\Pr[t^* \in L_t \mid L_t \neq \emptyset] \leq 2^{n(n+2)} N^{-|A^* \setminus A|}$.*

For the proof of Lemma 27, we need the following probability bound.

LEMMA 28. *For every real-valued random variable X,*

$$\mathrm{E}[X \mid X \neq 0]\,\mathrm{E}[X] \leq \mathrm{E}[X^2].$$

Proof. Let $\lambda = \Pr[X \neq 0]$. If $\lambda = 0$, then $\mathrm{E}[X] = \mathrm{E}[X^2] = 0$ and we are done. Assume then $\lambda \neq 0$. Let Y be the random variable defined by $\Pr[Y = a] = \Pr[X = a]/\lambda$ for every $a \neq 0$ and $\Pr[Y = 0] = 0$. Observe that $\mathrm{E}[Y] = \mathrm{E}[X]/\lambda$ and $\mathrm{E}[Y^2] = \mathrm{E}[X^2]/\lambda$. By Jensen's inequality (using that $f(x) = x^2$ is convex), we have $(\mathrm{E}[Y])^2 \leq \mathrm{E}[Y^2]$. Therefore,

$$\mathrm{E}[X \mid X \neq 0]\,\mathrm{E}[X] = \mathrm{E}[X]^2/\lambda = \lambda(\mathrm{E}[Y])^2 \leq \lambda\,\mathrm{E}[Y^2] = \mathrm{E}[X^2],$$

which is what we had to show.   □

Lemma 28 shows that we can prove Lemma 27.1 by bounding $\mathrm{E}[|L_t|^2]$.

Proof of Lemma 27. We show that

$$\mathrm{E}\left[|L_t|^2\right] \leq \mathrm{E}[|L_t|] \cdot 2^{n(n+2)}$$

by an argument similar to the proof of Proposition 14. Statement 1 follows from Lemma 28 with $X = |L_t|$.

Let T be the set of all A^*-tuples whose projection to A is t. Let $X(t^*)$ be the indicator random variable corresponding to the event $t^* \in L_t$. We need to bound

$$\mathrm{E}\left[|L_t|^2\right] = \mathrm{E}\left[\sum_{t_1^*, t_2^* \in T} X(t_1^*)X(t_2^*)\right] = \sum_{t_1^*, t_2^* \in T} \mathrm{E}\left[X(t_1^*)X(t_2^*)\right].$$

For every $A \subseteq B \subseteq A^*$, let F_B be the set of all pairs $(t_1^*, t_2^*) \in T^2$, where t_1^* and t_2^* agree exactly on B. We have that $|F_B| = N^{|B\setminus A|} \cdot N^{|A^*\setminus B|} \cdot (N-1)^{|A^*\setminus B|} \leq N^{|B\setminus A| + 2|A^*\setminus B|} = N^{|A^*\setminus A| + |A^*\setminus B|}$. Let w be the total weight of the relations in S^* and let w_B be the total weight of the relations in $S^*[B]$. Observe that $\mathrm{E}[|L_t|] = N^{|A^*\setminus A|} \cdot 2^{-w}$. Furthermore, we have $w - w_B \geq |A^*\setminus B|(\log N - n - 1)$: otherwise $f_S(B)$ would be strictly less than $f_S(A^*)$, contradicting the minimality of $A^* = C_S(A)$. For every $(t_1^*, t_2^*) \in F_B$, the event $X(t_1^*)X(t_2^*) = 1$ implies that for every relation $R \in S^*$, the projections of t_1^* and t_2^* to the attributes of R is in $R(D)$. For different relations, these events are clearly independent. For every $R \in S^*[B]$, the projections of t_1^* and t_2^* to the attributes of R are the same. Thus the probability that the projections are in the relation for every $R \in S^*[B]$ is exactly 2^{-w_B}. Consider now a relation in $R \in S^* \setminus S^*[B]$. For such an R, the projections of t_1^* and t_2^* are different, thus their appearance in $R(D)$ are independent events. As the total weight of the relations in $S^* \setminus S^*[B]$ is $w - w_B$, we get the following bound:

$$\mathrm{E}\left[X(t_1^*)X(t_2^*)\right] \leq 2^{-w_B - 2(w-w_B)}.$$

It follows that

$$\mathrm{E}\left[|L_t|^2\right] = \sum_{t_1^*, t_2^* \in T} \mathrm{E}\left[X(t_1^*)X(t_2^*)\right] = \sum_{B \subseteq A^*} \sum_{(t_1^*, t_2^*) \in F_B} \mathrm{E}\left[X(t_1^*)X(t_2^*)\right]$$

$$\leq 2^{|A^*|} \cdot N^{|A^*\setminus A| + |A^*\setminus B|} \cdot 2^{-w_B - 2(w-w_B)}$$

$$= (N^{|A^*\setminus A|} \cdot 2^{-w}) \cdot 2^{|A^*|} \cdot 2^{|A^*\setminus B|\log N - (w-w_B)}$$

$$\leq \mathrm{E}[|L_t|] \cdot 2^{|A^*|} \cdot 2^{|A^*\setminus B|(n+1)} \leq \mathrm{E}[|L_t|] \cdot 2^n \cdot 2^{n(n+1)} \leq \mathrm{E}[|L_t|] \cdot 2^{n(n+2)},$$

which is what we had to show.

To prove the second statement, observe first that if we fix an A-tuple t, then by symmetry, $\Pr[t^* \in L_t \mid L_t \neq \emptyset]$ has the same value for every A^*-tuple t^* with $\pi_A(t^*) = t$. There are $N^{|A^* \setminus A|}$ such tuples t^*, and the size of L_t is the sum of the indicator variables corresponding to these tuples. It follows that $\Pr[t^* \in L_t \mid L_t \neq \emptyset] = \mathrm{E}[|L_t| \mid L_t \neq \emptyset] N^{-|A^* \setminus A|} \leq 2^{n(n+2)} N^{-|A^* \setminus A|}$. □

Let φ'' be the plan obtained from φ' after Step 2 and let ℓ be the number of leaves of the plan. We show that

$$\mathrm{E}[|\psi''(D)|] \leq 2^{n(n+2)+2\ell} \, \mathrm{E}[|\psi'(D)|]$$

for every subplan ψ'' of φ'' and corresponding subplan ψ' of φ'. We consider three cases.

Case 1. If ψ' is a subplan of ψ'_0, then $\psi'' = \psi'$.

Case 2. Suppose that ψ' is a subplan strictly containing $\pi_A(\varphi'_0)$ such that the root of ψ' is a projection and let ψ'' be the corresponding subplan after applying Step 2 (e.g., consider the node $\pi_{\{a,c,f\}}$ in Figure 1). Observe that $A_{\psi'} = A_{\psi''}$: extra attributes can be introduced only by the change from $\pi_A(\varphi'_0)$ to $\pi_{A^*}(\varphi'_0)$, and any such extra attribute is either already present in A'_ψ or projected out by the projection in the root of ψ'. For example, in Figure 1, replacing $\pi_{\{a,b\}}$ with $\pi_{\{a,b,c,d\}}$ has no effect on the attributes of the nodes above the projection $\pi_{\{a,c,f\}}$ as attribute c already appears there and attribute d is projected out.

We show that $\psi''(D) \subseteq \psi'(D)$. Indeed, if $t \in \psi''(D)$, then $\pi_{A^*}(t) \in \pi_{A^*}(\varphi'_0(D))$, which implies $\pi_A(t) \in \pi_A(\varphi'_0(D))$. This implies $t \in \psi'(D)$, since the other subplans of ψ' did not change. Thus Step 2 does not increase the size of $\psi'(D)$ if ψ' is a subplan containing $\pi_A(\varphi'_0)$ and the root of ψ' is a projection. Furthermore, it also follows that if ψ' is a subplan containing $\pi_A(\varphi'_0)$ such that there is a projection node above $\pi_A(\varphi'_0)$, then the size of $\psi'(D)$ does not increase.

Case 3. The only remaining situation that we have to verify is if ψ' is a subplan of φ' containing $\pi_A(\varphi'_0)$ and having no projection node above $\pi_A(\varphi'_0)$. For example, consider any of the two join nodes between $\pi_{\{a,b\}}$ and $\pi_{\{a,c,f\}}$ in Figure 1. We show that in this case $\mathrm{E}[|\psi''(D)|] \leq 2^{n(n+2)+2\ell} \, \mathrm{E}[|\psi'(D)|]$, where ψ'' is the subplan of φ'' corresponding to ψ'.

Note that $A^* \subseteq A_{\psi''}$, since ψ'' contains no projections above π_{A^*}. Let C be $A_{\psi'} \setminus (A^* \setminus A) = A_{\psi''} \setminus (A^* \setminus A)$. If a tuple t is in $\psi''(D)$, then clearly $t_1 := \pi_{A^*}(t)$ is in $\varphi'_0(D)$ and $t_2 := \pi_C(t)$ is in $\pi_C(\psi'(D))$. Thus

$$
\begin{aligned}
\Pr[t \in \psi''(D)] &\leq \Pr[t_1 \in \varphi'_0(D) \wedge t_2 \in \pi_C(\psi'(D))] \\
&= \Pr[\pi_A(t_1) \in \pi_A(\varphi'_0(D))] \cdot \\
&\quad \cdot \Pr[t_1 \in \varphi'_0(D) \mid \pi_A(t_1) \in \pi_A(\varphi'_0(D))] \cdot \\
&\quad \cdot \Pr[t_2 \in \pi_C(\psi'(D)) \mid t_1 \in \varphi'_0(D)].
\end{aligned}
$$

(5.15)

By Lemma 27.2 with $t = t^* = t_1$, the second factor in (5.15) is at most $2^{n(n+2)} \cdot N^{-|A^* \setminus A|}$. To bound the third factor, observe that

$$
I_{\pi_A(\varphi'_0(D)), \pi_A(t_1)} = \bigvee_{\substack{t'_1 \in \mathrm{tup}(\varphi'_0) \\ \pi_A(t'_1) = \pi_A(t_1)}} I_{\varphi'_0(D), t'_1}
$$

by the definition of the projection. As the plan φ'_0 does not contain any projections, each term $I_{\varphi'_0(D), t'_1}$ on the right-hand side is the product of random variables, i.e.,

they are the minterms of $I_{\pi_A(\varphi_0'(D)),\pi_A(t_1)}$. The conditional probability

$$\Pr[I_{\pi_C(\psi'(D)),t_2} = 1 \mid I_{\varphi_0'(D),t_1'} = 1]$$

is the same for every $t_1' \in \operatorname{tup}(\varphi_0')$ with $\pi_A(t_1') = \pi_A(t_1) = \pi_A(t_2)$, since the common attributes of t_1' and t_2 are exactly in A. Therefore, we can use Lemma 25.2 to show that the probability conditioned on a minterm of $I_{\pi_A(\varphi_0'(D)),\pi_A(t_1)}$ can be bounded by the probability conditioned on $I_{\pi_A(\varphi_0'(D)),\pi_A(t_1)}$ itself. In particular, we bound the probability of the event $I_{\pi_C(\psi'(D)),t_2} = 1$ conditioned on the minterm corresponding to t_1, which is exactly the third factor appearing in (5.15):

$$\Pr[I_{\pi_C(\psi'(D)),t_2} = 1 \mid I_{\varphi_0'(D),t_1} = 1] \le 2^{2\ell} \Pr[I_{\pi_C(\psi'(D)),t_2} = 1 \mid I_{\pi_A(\varphi_0'(D)),\pi_A(t_1)} = 1].$$

Therefore, continuing (5.15), we have

$$\begin{aligned}
\Pr[t \in \psi''(D)] &\le \Pr[\pi_A(t_1) \in \pi_A(\varphi_0'(D))]\cdot \\
&\quad \cdot 2^{n(n+2)} N^{-|A^*\setminus A|}\cdot \\
&\quad \cdot 2^{2\ell} \Pr[t_2 \in \pi_C(\psi'(D)) \mid \pi_A(t_1) \in \pi_A(\varphi_0'(D))] \\
&\le 2^{n(n+2)+2\ell} N^{-|A^*\setminus A|} \cdot \Pr[t_2 \in \pi_C(\psi'(D))] \\
&= 2^{n(n+2)+2\ell} N^{-|A^*\setminus A|} \cdot \mathrm{E}[|\pi_C(\psi'(D))|]N^{-|C|}.
\end{aligned}$$

The last equality follows from the fact that the probability $\Pr[t_2 \in \pi_C(\pi'(D))]$ is the same for every fixed t_2 by symmetry. Thus the expected size of $\psi''(D)$ is

$$\begin{aligned}
\mathrm{E}[|\psi''(D)|] &= N^{|A_{\psi''}|} \cdot 2^{n(n+2)+2\ell} N^{-|A^*\setminus A|} \cdot \mathrm{E}[|\pi_C(\psi'(D))|]N^{-|C|} \\
&= 2^{n(n+2)+2\ell} \mathrm{E}[|\pi_C(\psi'(D))|] \\
&\le 2^{n(n+2)+2\ell} \mathrm{E}[|\psi'(D)|].
\end{aligned}$$

6. Conclusions. We have conducted a theoretical study of database queries from the viewpoint of bounding or estimating the size of the answer. In the worst-case model, we showed that the fractional edge cover number, or more generally, the solutions of certain linear programs, can be used to obtain fairly tight bounds. In the random database model, we investigated bounds on the expected size and whether the number of solutions is well concentrated around the expectation. Perhaps the most interesting message of the paper is that from the viewpoint of worst-case size, join-project plans can be significantly more efficient than join plans for the same query, while in the average case model every join-project plan can be turned into a join plan with only a bounded loss of performance.

Let us mention two possible directions in which our results could be further developed. First, one can introduce functional dependencies into the model and generalize the bounds to take these restrictions on the relations into account. This has been investigated in [8] and [15], but the problem has not been fully resolved yet. Another direction to investigate is to understand concentration bounds for join-project plans. In particular, Theorem 20 states a bound only on the expected value, but it does not give an upper bound on size that holds with high probability. It remains a challenging problem to prove a variant of Theorem 20 saying that if every subplan of the join-project plan has bounded size with high probability, then there is also a join plan with this property.

REFERENCES

[1] S. ABITEBOUL, R. HULL, AND V. VIANU, *Foundations of Databases*, Addison-Wesley, Reading, MA, 1995.

[2] N. ALON AND J. SPENCER, *The Probabilistic Method*, 2nd ed., John Wiley, New York, 1992.

[3] S. CHAUDHURI, *An overview of query optimization in relational systems*, in Proceedings of the 17th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS), 1998, pp. 34–43.

[4] F. R. K. CHUNG, R. L. GRAHAM, P. FRANKL, AND J. B. SHEARER, *Some intersection theorems for ordered sets and graphs*, J. Combin. Theory Ser. A, 43 (1986), pp. 23–37.

[5] J. FLUM, M. FRICK, AND M. GROHE, *Query evaluation via tree-decompositions*, J. ACM, 49 (2002), pp. 716–752.

[6] E. FRIEDGUT AND J. KAHN, *On the number of copies of a hypergraph in another*, Israel J. Math., 105 (1998), pp. 251–256.

[7] H. GARCIA-MOLINA, J. WIDOM, AND J. D. ULLMAN, *Database System Implementation*, Prentice-Hall, Englewood Cliffs, NJ, 1999.

[8] G. GOTTLOB, S. T. LEE, G. VALIANT, AND P. VALIANT, *Size and treewidth bounds for conjunctive queries*, J. ACM, 59 (2012), pp. 1–35.

[9] G. GRAEFE, *Query evaluation techniques for large databases*, ACM Computing Surveys, 25 (1993), pp. 73–169.

[10] M. GROHE AND D. MARX, *Constraint solving via fractional edge covers*, in Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), 2006, pp. 289–298.

[11] J. HÅSTAD, *Clique is hard to approximate within $n^{1-\epsilon}$*, Acta Math., 182 (1999), pp. 105–142.

[12] J. H. KIM AND V. H. VU, *Concentration of multivariate polynomials and its applications*, Combinatorica, 20 (2000), pp. 417–434.

[13] C. H. PAPADIMITRIOU, *Computational Complexity*, Addison-Wesley, Reading, MA, 1994.

[14] J. RHADAKRISHNAN, *Entropy and Counting*, http://www.tcs.tifr.res.in/~jaikumar/mypage.html.

[15] G. VALIANT AND P. VALIANT, *Size bounds for conjunctive queries with general functional dependencies*, arXiv:0909.2030v2, 2010.

[16] V. V. VAZIRANI, *Approximation Algorithms*, Springer-Verlag, New York, 2001.