Uniform Generation in Spatial Constraint Databases and Applications

David Gross-Amblard

Conservatoire National des Arts et Métiers-Paris, équipe Vertigo, 292, rue St. Martin
75141 Paris Cedex 03, France

and

Michel de Rougemont

Université Paris II and L.R.I., Université Paris XI, Bât. 490
91405 Orsay cedex, France

E-mail: dgram@cnam.fr; mdr@lri.fr

We study the efficient approximation of queries in linear constraint databases using sampling techniques. We define the notion of an almost uniform generator for a generalized relation and extend the classical generator of Dyer, Frieze and Kannan for convex sets to the union and the projection of relations. For the intersection and the difference, we give sufficient conditions for the existence of such generators.

We show how such generators give relative estimations of the volume and approximations of generalized relations as the composition of convex hulls obtained from the samples.

Key Words: constraint databases, approximation, estimator, generator, sampling, volume, dimension

1. INTRODUCTION

The constraint database model, introduced by [?], offers a uniform way to handle spatial information. This model allows the manipulation of arbitrary high dimensional geometric sets in a unified framework. But, as noticed in [?] and many other papers, the complexity of constraint query languages behaves badly with the dimension of the geometric sets (basically exponentially in the dimension). In this paper we present a general technique to approximate queries, based on the uniform generation, i.e. on the generation of random points in a definable set with a uniform distribution. We show the relationships between uniform generation, approximate computation of the volume and approximation of the set defined by a first-order formula. We use randomized algorithms, i.e. procedures that succeed with a high probability, and many papers show that only randomized algorithms can produce such approximations (see [?]).

Uniform random sampling has many applications in databases: statistical analysis, decision support, and estimation of aggregate queries where an approximate result is sufficient ([?, ?]). These methods are of primary interest for Geographical Information Systems (GIS), because many applications are of a statistical nature ([?, ?, ?]). Since constraint databases are well-suited for GIS applications, it is natural to consider sampling operations in this setting.

Algorithms that generate random points with an almost uniform distribution in a given set are called *uniform generators* and, in case of convex sets, can be related to the computation of approximate volumes in polynomial time in the dimension with a relative error. Relative approximation necessarily requires randomized algorithms.

We use two basic tools: the almost uniform polynomial time generator of Dyer, Frieze and Kannan [?] for a convex set, and the approximation of a polytope by convex hull built from uniform samples [?]. We show how to obtain almost uniform generator, approximate volumes and approximate sets for generalized relations. Our main results are:

1. An almost uniform generator for generalized relations in DNF form and for the projection of a convex relation. In the general case, we give a sufficient condition for the existence of a uniform generator and for the relative approximation of the volume.
2. A set reconstruction method for positive existential queries.

Related work. The use of sampling in classical databases has been widely studied for the approximation of aggregate $COUNT$ queries ([?, ?]). From the practical point of view, Olken and Rotem [?] study the uniform generation from a collection of spatial objects stored in an R-tree. They point out that their sampling algorithm scales up to any dimension, but they consider sampling from one spatial object as a black box. The volume approximation (with an additive error) of definable sets has been studied by Koiran, Karpinski and Macintyre ([?, ?]) who considered logical formulas that derandomize a Monte-Carlo integration method. The problem of designing good query languages with a volume operator and an approximate volume operator (for an additive error) is studied by Benedikt and Libkin ([?]).

On classical discrete domains, the uniform generation of points in an NP relation has been studied by Jerrum, Valiant and Vazirani ([?]): they prove that for self-

reducible NP relations the problem of approximate counting is equivalent to the problem of almost uniform generation. It is then natural to define the notion of a uniform generator for a generalized relation and relate this notion to the approximation of the volume. In the continuous setting, the volume of a convex polytope is $\#P$-hard to compute in the dimension. If we consider relative approximations, Elekes, Bárány and Füredi [?, ?] show that any approximation algorithm must be randomized.The first fully polynomial approximation scheme for the volume of a convex body was given by Dyer, Frieze and Kannan ([?]). This procedure is non trivial and cannot be achieved with the uniform sampling in the unit cube: for example, an exponential number of trials are necessary to obtain a single sample from a d-dimensional sphere (the ratio of the volume of a square and a d-dimensional sphere is $\Omega(\frac{1}{d^d})$).

Organization. In the next section we recall the basic definitions of constraint databases and define the notion of an $(\gamma, \varepsilon, \delta)$-uniform generator. In the third section, we study the relationship between uniform generator and volume approximation. In the fourth section we study applications to the approximation of queries, and the generalization to polynomial constraints.

2. NOTATIONS AND DEFINITIONS

Constraint databases. We are using standard notations of linear constraint databases ([?, ?, ?, ?, ?, ?, ?, ?, ?, ?]). Let $\mathcal{U}$ be an infinite set. We call $\mathcal{M} = \langle \mathcal{U}, \Omega \rangle$ an infinite structure with domain $\mathcal{U}$. The set Ω is the set of interpreted functions, predicates and constants. We restrict our attention to linear constraints over the reals, i.e. constraints associated to the structure $\mathcal{R}_{lin} = \langle \mathbb{R}, +, -, <, 0, 1 \rangle$.

A *d-ary generalized tuple* is a conjunction of atomic formulas in the language of $\mathcal{M}$. A *d-ary finitely representable relation* is a set $S \subseteq \mathcal{U}^d$ such that there exists a first-order formula ϕ over the language of $\mathcal{M}$ with:

$$\forall \bar{a} \in \mathcal{U}, \mathcal{M} \models \phi(\bar{a}) \text{ if and only if } \bar{a} \in S.$$

It is also called a *generalized relation*. Since the structure $\mathcal{R}_{lin}$ admits elimination of quantifiers, the formula ϕ is equivalent to a quantifier-free formula. This formula is equivalent to a disjunctive normal form, thus each generalized relation is a finite union of generalized tuples. The size of a relation S is the number of symbols of the formula defining S.

A *relational database schema* is a set of relation names $\{R_1, \ldots, R_l\}$. A *finitely representable instance* is a collection of generalized relations $\{S^1, \ldots, S^l\}$, each associated with its corresponding name in the schema. Our query language will be the first-order logic over the structure $\mathcal{R}_{lin}$ and the database schema, denoted by $FO + LIN$. It consists of the atomic formulas over the schema and $\{+, -, <, 0, 1\}$, and the natural composition of boolean connectives and quantifiers.

Geometry. A relation S represented only by one generalized tuple over the language of $\mathcal{R}_{lin}$ is a finite intersection of open or closed halfspaces. This means that this relation S is convex. If we are given two rational numbers r_{inf} and r_{sup} such that S contains a ball of radius r_{inf} and is totally contained in a ball of radius r_{sup}, we say that S is a *well-bounded convex relation* ([?]). Since a generalized relation

S is represented by a finite union of generalized tuples, this is also a finite union of convex. We call S a *well-bounded relation* if all these convex are well-bounded.

In the sequel, μ_S will be the d-dimensional volume of the relation S. This value is well defined since all bounded finitely representable relations in $FO + LIN$ are measurable. Since we focus on complexity issues, we consider sequences of instances: we will denote by R the sequence of relations $(R_d)_{d\in\mathbb{N}}$, where, for each d, R_d has dimension d. The union (resp. intersection) of a finite set of sequences $S^1,\ldots,S^k$ is the sequence T defined for each $d \in \mathbb{N}$ by $T_d = S^1_d \cup \ldots \cup S^k_d$ (resp. $T_d = S^1_d \cap \ldots \cap S^k_d$). The minimum (relatively to the the volume) of a finite set of sequences, denoted by $\min(S^1,\ldots,S^k)$, is the sequence T such that for each $d \in \mathbb{N}$, $T_d = S^{j_0}_d$, where j_0 the smallest element of $\{1,\ldots,k\}$ such that $\mu_{S^{j_0}_d} = \min_i(\mu_{S^i_d})$, $i \in \{1,\ldots,k\}$.

DEFINITION 2.1. Two sequences of relations R and S are polynomially related (poly-related for short) if there exists a constant $k \in \mathbb{N}$ such that, for every $d \in \mathbb{N}$:

$$\max\{\frac{\mu_{R_d}}{\mu_{S_d}}, \frac{\mu_{S_d}}{\mu_{R_d}}\} \leq d^k.$$

We will sometimes use "relation" instead of sequence of relations when the exact meaning is clear from the context.

Generators and estimators. We shall consider the problem of sampling points uniformly from a relation S. Since the domain of $\mathcal{R}_{lin}$ is infinite, we consider a discretization of S. We call a *grid of step p* the set $\mathcal{G}_p$ of points in $\mathbb{R}^d$ whose coordinates are multiple of p. For a relation $S \subseteq \mathbb{R}^d$, *the graph induced by $\mathcal{G}_p$ on S* is the graph with vertices $V = \mathcal{G}_p \cap S$. Edges of this graph are pairs $(\bar{a}, \bar{b}) \in V \times V$ such that $\bar{a}$ and $\bar{b}$ are at distance p. We use a small enough grid such that the number of vertices induced on S is closely related to the volume of S. All approximations are relative, as in the classical literature on fully polynomial-time randomized approximation schemes ($FPRAS$).

A randomized algorithm has the ability to pick a random bit b at each step, and to adapt its computation according to the value of b. Hence a given computation is a path in the tree of all possible random choices along with its corresponding probability: both form a probability space Ω. We note for short $j \in_R S$ the random choice of an element in S with a prescribed probability.

If α, β and ε are positive reals with $0 < \varepsilon < 1$, we say that α approximates β with ratio $1+\varepsilon$ if $(1+\varepsilon)^{-1}\beta \leq \alpha \leq (1+\varepsilon)\beta$. A randomized algorithm is said to be an (ε,δ)-*volume estimator* for a relation S if, given $0 < \varepsilon, \delta < 1$ as parameters, it computes a value $\hat{\mu}_S$ such that:

$$Pr_\Omega[\hat{\mu}_S \text{ approximates } \mu_S \text{ with ratio } 1+\varepsilon] \geq 1-\delta.$$

Its running time must be polynomial in the description size of S, $\frac{1}{\varepsilon}$ and $\ln(\frac{1}{\delta})$. The $\ln(\frac{1}{\delta})$ bound on complexity is a classical assumption.

Let $0 < \gamma < 1$ and p such that the value $|V|.p^d$ approximates μ_S with ratio $1+\gamma$. In order to avoid too small grids, we suppose that p is polynomial in γ and $\frac{1}{d}$. If these two conditions are met, we call $\mathcal{G}_p$ a γ-*grid for* S. Except for the discretization parameters, we use standard notions from [**?**]. A generator for S is a randomized

algorithm which generates a point uniformly in the graph induced on S by a γ-grid for S. For several reasons, it is convenient to consider such algorithms that may succeed with high probability and may fail, i.e. stop and abandon, with a small probability δ. The distribution of the output is also allowed to deviate from the uniform distribution by a little amount (prescribed by ε).

DEFINITION 2.2. An $(\gamma, \varepsilon, \delta)$-*generator* is a randomized algorithm which, given a relation S and real numbers $0 < \varepsilon, \delta, \gamma < 1$, computes a γ-grid $\mathcal{G}_p$ and outputs points from $V = \mathcal{G}_p \cap S$ such that:

1. When a computation is successful, for all vertices v of V,

$$\frac{1}{1+\varepsilon}\frac{1}{|V|} \leq Pr_\Omega[\text{output is } v] \leq (1+\varepsilon)\frac{1}{|V|}.$$

2. The algorithm fails with probability smaller than δ.
3. The algorithm runs in time polynomial in the description size of S, d, $\frac{1}{\varepsilon}$, $\frac{1}{\gamma}$ and $\ln\frac{1}{\delta}$.

Notice that γ controls the size of the grid $\mathcal{G}_p$ in dimension d, i.e. $|V|.p^d$ must be an approximation of the volume of S with ratio $(1+\gamma)$. The parameter ε controls the quality of the distribution.

A relation that possesses both a generator and a volume estimator is said to be *observable* and we will often use this notion.

Uniform sampling from a convex and volume estimation. Given a well-bounded convex body K by a membership oracle (i.e. an algorithm that tells if a point belongs to the set) the Dyer-Frieze-Kannan technique [**?**] first computes a non-singular affine transformation Q that makes the body K "well-rounded". The transformed body $Q(K)$ contains the unit ball B and is totally contained in a ball of radius $\sqrt{d}(d+1)$, depending only on the dimension of the space (this is possible only if the convex is well-bounded). In a second step, they consider a random walk on the graph $\mathcal{G}$ induced by a γ-grid on the set $Q(K)$, starting at the origin vertex. This random walk is rapidly mixing: after a polynomial number of steps, the random walk is almost uniformly distributed on all the vertices of $\mathcal{G}$. Finally, they consider a sequence of convex sets $B = K_0 \subseteq K_1 \subseteq \ldots \subseteq K_q = Q(K)$ such that $\frac{\mu_{K_{i+1}}}{\mu_{K_i}}$ is bounded by a constant (taking homothetic K_i' s is sufficient). The uniform generator for each convex is used to estimate each ratio (by a classical Chernoff estimator). The product of each ratio give the approximate volume of $Q(K)$, thus of K.

A membership oracle for a finitely representable relation S is easy to compute in linear time in its description size: it is sufficient to check each constraint for the given assignment of variables. This leads to the following fundamental result:

THEOREM. (Dyer,Frieze,Kannan) If for each $d \in \mathbb{N}$, S_d is convex and well-bounded, then S is observable.

For ε, γ, δ and S as input, the grid size p will be $O(\gamma/d^{\frac{3}{2}})$. The mixing time of the random walk is $O(\frac{d^{19}}{\varepsilon\gamma}\ln\frac{1}{\delta})$, so a random point can be generated in polynomial time. The volume estimator, using the generator, has complexity $O(\frac{d^{19}}{\varepsilon}\ln\frac{1}{\delta})$. Several improvements reduce this complexity to $O(d^5)$ [**?**].

3. UNIFORM GENERATION AND VOLUME APPROXIMATION

The relationship between almost uniform generation and approximate counting is well studied ([**?**]). For the continuous setting, we will generalize the method of [**?**] to combinations of observable relations.

3.1. Boolean operations

We will define different algorithms that samples from a relation and approximate its volume. For an observable relation S, we consider $G_S = (V_S, E_S)$ the graph induced by a γ-grid $\mathcal{G}_S$ for S. We denote by $ApproxGen(S, \gamma, \varepsilon, \delta)$ the generator for S and by $ApproxVol(S, \varepsilon, \delta)$ the volume estimator for S. We now study how to compose these operators with the union, the intersection and the difference of observable relations.

3.1.1. Union

To sample the union of observable relations, we use an argument similar to the one given in [**?**] for the approximation of $\#DNF$, but in the geometrical setting. Remark that a direct random walk on the union is not likely to succeed, because the mixing rate is not known (one can construct union of relations such that the random walk does not mix in polynomial time).

THEOREM 3.1. *Given observable relations $S^1, \ldots, S^m$, there exists an $(\gamma, \varepsilon, \delta)$-generator for the relation T defined by $\bigcup_{i=1}^m S^i$.*

We will first choose one of the relations with probability proportional to its volume, and choose a random point in this relation with a uniform distribution. In order to deal with overlapping relations and ensure that each point is chosen only once, we output the point only if it is chosen from the first possible relation in the natural ordering over $S^1, \ldots, S^m$. Below is an inductive definition of $ApproxGen$ for T given algorithms $ApproxGen$ and $ApproxVol$ for base relations $S^1, \ldots, S^m$.

ALGORITHM 1 ($ApproxGen(T, \gamma, \varepsilon, \delta)$).

Repeat k times:

1. For each $i \in \{1, \ldots, m\}$, computes $\hat{\mu}_i = ApproxVol(S^i, \frac{\varepsilon}{3}, \frac{1}{4m})$.
2. Let $\hat{\mu} = \sum_{i=1}^m \hat{\mu}_i$.
3. Choose a $j \in_R \{1, \ldots, m\}$ with probability $\frac{\hat{\mu}_j}{\hat{\mu}}$.
4. Let $x = ApproxGen(S^j, \gamma, \frac{\varepsilon}{3}, \frac{1}{4m})$.
5. If $(x \notin S^l)$ for every $l < j$, return x, else fail.

One of the difficulties when composing generators is the grid size. Remark that if a relation is exponentially smaller that the others, one can consider it empty without modifying the approximation ratio. We then suppose that each of the S^i's

have poly-related volumes. For a point x, we denote by $j(x)$ the smallest index i such that S^i contains x.

Proof. We must verify the construction of the grid and the three conditions of definition **??**.

To construct the greatest common grid $\mathcal{G}$ of all the $\mathcal{G}_{S^i}$, one can extend the uniform generator of each S^i to this grid. Since the S^i's are poly-related, the resulting grid size is not too small. Let $V_T = \bigcup_{i=1}^m V_{S^i}$ be the vertices of the graph induced by $\mathcal{G}$ on T.

Let $\mu_i = \mu_{S^i}$ and $\mu = \sum_{i=1}^m \mu_i$. The algorithm *ApproxGen* may fail because of probabilistic steps and the test in (5). Let us first suppose that each step succeeds and that a point x is returned. The only way to return this particular x is that step (3) produces a j equal to $j(x)$. The probability that this particular relation is chosen is $\frac{\hat{\mu}_{j(x)}}{\hat{\mu}}$. The point x is drawn from $S^{j(x)}$ by *ApproxGen* with probability $\frac{1}{\tilde{\mu}_{j(x)}}$. So the output probability of x is $\frac{\hat{\mu}_{j(x)}}{\hat{\mu}}\frac{1}{\tilde{\mu}_{j(x)}}$, which approximates $\frac{1}{\mu}$ with ratio $(1+\frac{\varepsilon}{3})^3 \leq (1+\varepsilon)$. This gives the desired almost uniform distribution.

The algorithm is unreliable at steps (1), (4) and (5). For step (1) and (4) we choose a failure probability $\delta' = \frac{1}{4m}$. For step (5), let x_0 a point in V_T. Let N be the number of different V_{S^i} containing x_0. There is at most N distinct pairs (j, x_0) that can be produced by the algorithm at step (3) and (4), each with the same probability. But only one leads to an accepting state (i.e. pair $(j(x_0), x_0)$). So for any x_0, the success probability of step (5) is $\frac{1}{N}$, which is bigger than $\frac{1}{m}$.

The total failure probability is smaller than $2\delta' + 1 - \frac{1}{m} = 1 - \frac{1}{2m}$. Considering k successive executions of this algorithm, the overall failure probability is smaller than $(1-\frac{1}{2m})^k$. Since $(1-\frac{1}{2m})^k \leq e^{-\frac{k}{2m}}$, repeating the whole algorithm $k = 2m.\ln(\frac{1}{\delta})$ times gives a general failure probability smaller than parameter δ.

The total complexity of this algorithm is bounded by $m \ln \frac{1}{\delta}$ times the complexity of steps (1) and (4). They are both polynomial in d, and $\frac{1}{\varepsilon}$ by hypothesis. The whole algorithm is then polynomial in m, d, $\frac{1}{\varepsilon}$, $\frac{1}{\gamma}$ and $\ln(\frac{1}{\delta})$. ■

THEOREM 3.2. *Given well-bounded observable relations $S^1, \ldots, S^m$, there exists an (ε, δ)-volume estimator for the relation T defined by $\bigcup_{i=1}^m S^i$.*

Proof. Let B_0 be the inner ball of S^1, and B_1 the ball enclosing all S^i's. We apply the same volume estimation method as in the convex case, but now using the generator from the union. Notice that the description size of relations S^1,...,S^m and B_1's radius are related by a linear function. ■

COROLLARY 3.1. *Well-bounded observable relations are closed under finite union.*

3.1.2. Intersection.

Consider T as the intersection of the observable relations S^1, S^2, ..., S^m. The situation is more complex as the intersection may be exponentially smaller than the smallest S^i, in which case we will neither have a uniform generator nor an approximation of the volume.

PROPOSITION 3.1. *The relation T defined as $\bigcap_{i=1}^{m} S^i$ is observable if T and $min(S^1, \ldots, S^m)$ are poly-related.*

Proof. Compute the approximate volumes of $S^1, ..., S^m$. Choose j such that S^j has the smallest volume. Use the generator for S^j and check whether the points are in $S^1, ...S^{j-1}, S^{j+1}, ..., S^m$. If there are, output them, otherwise iterate the process.

We obtain a $(\gamma, \varepsilon, \delta)$-uniform generator for T: the γ-grid of S^j is a γ-grid for T because T and S^j are poly-related. The previous generator is an $(\gamma, \varepsilon, \delta)$-generator. After some polynomial time, we almost surely obtain points in T and their convex hull contains a ball of radius r_{min}. The set T is almost uniformly generated and well bounded: its volume can be approximated by the technique of [**?**]. ■

Notice that a SAT instance can be encoded in the following geometric way : with each literal x (resp. $\bar{x}$), we associate the constraint $\frac{3}{4} < x < 1$ (resp. $0 < x < \frac{1}{4}$). A disjunction is the finite union of such constraints, defining a finite union of convex sets, which are observable. A SAT instance is finally represented as the intersection of such observable sets.

Since relative volume approximation can be used to decide emptiness of a geometric set, an (ε, δ)-volume estimator for the general intersection would yield a polynomial-time algorithm for the SAT problem. Hence, the restriction on the relative size of T is necessary.

3.1.3. Difference.

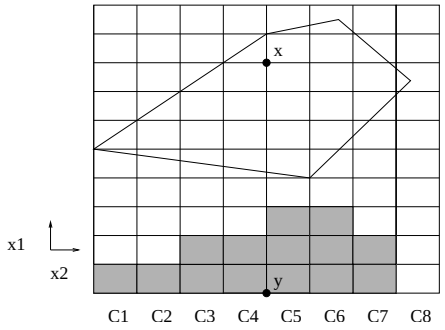
Consider the difference of two observable sets S^1 and S^2. It is neither connected nor convex in general but may still be observable.

PROPOSITION 3.2. *The relation T defined as the difference of two observable relations S^1 and S^2 is observable if the size of T and S^1 are poly-related.*

Proof. Consider the generator for S^1 where we only output points which are not in S^2. The generator selects the γ-grid of S^1. Because T is relatively large, we almost surely obtain points in T after some polynomial time and their convex hull contains a ball of radius r_{min}. The set T is almost uniformly generated: we obtain an $(\gamma, \varepsilon, \delta)$-uniform generator and a (ε, δ)-volume estimator. ■

3.2. Projection

Suppose we have a generator for a convex relation $S \subseteq \mathbb{R}^d$, and let $x = (x_1, \ldots, x_d)$ a uniformly generated point in S. Then the point $y = (x_2, \ldots, x_d)$ belongs to T, the projection of S according to the first coordinate. But y is not uniform in T, as shown in the following figure:



In this example, $d = 2$, and C_1 to C_8 denote the cylinders induced by a γ-grid for S. If a point $x = (x_1, x_2)$ is drawn uniformly *on a grid* for S, it is more likely to appear in the cylinder C_5 than in the smaller C_1. Its projection $y = (x_2)$ on the second coordinate is not uniform in T. It is then necessary to compensate this effect: we reject y with probability proportional to the volume of the cylinder containing x, which can itself be computed by the previous algorithms.

THEOREM 3.3. *The relation T defined by projection of a convex relation S is observable.*

If S has a very elongated form, it can be "well-rounded" (i.e. mapped into a sphere by an affine transform). Then the volume of T and S are related. Given a subset I of coordinates and a point y in T, let $H_S(y)$ be the cylinder of points whose projection on coordinates in I is exactly y. We consider the following generator for the projection:

ALGORITHM 2 ($ApproxGen(T,\ I,\ \gamma, \varepsilon, \delta)$).

1. Repeat k times:
2. Choose $x = ApproxGen(S, \gamma, \frac{\varepsilon}{3}, \frac{1}{12})$.
3. Let y be the projection of x on the coordinates in I.
4. Compute $\hat{h} = ApproxVol(H_S(y), \varepsilon, \frac{1}{12})$.
5. Return y with probability $\frac{1}{2\hat{h}}$, else fail.

Proof. One can use the projection of a γ-grid for S as a γ-grid for T.

Let $\hat{\mu}$ be the volume of S and suppose that each probabilistic step of the algorithm succeeds. A point $y \in T$ is returned if step (2) generates a point $x \in H_S(y)$, and step (5) accepts. The first event occurs with probability $\frac{\hat{h}}{\hat{\mu}}$, where $\hat{h}$ is the volume of $H_S(y)$. Step (5) accepts with probability $\frac{1}{\hat{h}}$, so the overall probability is $\frac{1}{\hat{\mu}}$. This is almost uniform with ratio $(1+\varepsilon)^3 \leq (1+8\varepsilon)$.

The algorithm is unreliable on step (2) and (4) with probability $\frac{1}{12}$. Since the grid size p is equal to $\frac{\varepsilon}{d^{3/2}}$ and that $H_S(y)$ contains at least one point of the grid, the last step succeeds with probability at least $\frac{2d^{3/2}}{\varepsilon}$. Following the union algorithm in section **??**, repeating the whole algorithm $k = \frac{d^{3/2}}{\varepsilon} ln(\frac{1}{\delta})$ times gives the desired success probability.

Since the projection of the inner (resp. enclosing) balls of S is an inner (resp. enclosing) ball for T, the volume of T can be approximated by the technique of Dyer, Frieze and Kannan. ■

4. APPLICATIONS

We first consider the uniform sampling of a relation defined by a DNF formula. We then show how to use the general technique to approximate queries. Finally, we mention the generalization to polynomial constraints.

4.1. Sampling in relations represented by a *DNF* formula

The DNF-representation is the classical internal format for implemented constraint database systems. A relation S is then a finite union of convex sets S^i. One can use directly the Dyer-Frieze-Kannan generator from [**?**] for each convex and use our algorithm for the union. This yields a top-level sampling procedure for any query but we rely on the classical symbolic methods to obtain the DNF form. The sampling algorithm is polynomial in the dimension.

If the S^i are disjoint, sampling and estimating volume becomes easy. But the decompositions of a relation into disjoint convex components becomes exponentially harder with the dimension (mostly because the algorithms begins with a triangulation).

4.2. Reconstruction of queries

We consider the problem of reconstructing a definable relation from samples. In the classical approach to constraint spatial databases, one manipulates relations in a symbolic way: starting with an arbitrary first-order formula, one first needs to eliminate quantifiers and this is known to be hard. In order to obtain an asymptotic speed-up for queries over high dimensional relations, we would like to avoid symbolic computations. If we have an almost uniform generator, we may be able to approximate a query which defines a relation S, i.e. write formulas that define a relation $\hat{S}$ such that the volume of the symmetric difference will be small.

DEFINITION 4.1. A (ε, δ)-*estimator for a* **relation** S is a randomized algorithm such that, given $0 < \varepsilon, \delta < 1$ as parameters:

1. The algorithm produces the description of a relation $\hat{S}$ that approximates S with failure probability smaller than δ, i.e.:

$$\Pr_{\Omega}[\mu(S \Delta \hat{S}) \geq (1+\varepsilon)\mu(S)] \leq \delta,$$

where $S \Delta \hat{S} = (S - \hat{S}) \cup (\hat{S} - S)$ denotes the symmetric difference between S and $\hat{S}$.

2. The algorithm uses only point membership queries in S.

Notice that the previous generators require only point membership queries. We distinguish the case of convex sets from the general case.

4.2.1. Convex sets

The basic tool is a result from [**?**]: If we have a uniform generator for a convex polytope S, then the convex hull of N uniformly generated points approximates the set S with r vertices, with ratio $1 + \frac{rd}{(d)^{d-2}} \frac{\ln^{d-1} N}{N}$.

LEMMA 4.1. *If N is in $O(\frac{4r^2d^2}{\varepsilon^4 d^{2d-2}} \ln \frac{1}{\delta})$, the convex hull of N samples uniformly generated in a convex polytope S with r vertices is an ε-approximation of S with failure probability smaller than δ.*

Proof. Let N be the number of random generated points in a convex relation S with volume μ_S and r vertices. By the result from [?], we know that the expected volume $\mathbb{E}[V]$ of the convex hull of these points is such that:

$$\mu_S(1 - \frac{rd}{(d)^{d-2}} \frac{\ln^{d-1} N}{N}) \leq \mathbb{E}[V] \leq \mu_S.$$

For big enough N, $\frac{\ln^{d-1} N}{N} \leq \frac{1}{\sqrt{N}}$. Taking $N = \frac{4r^2d^2}{d^{2(d-2)}\varepsilon^2}$ ensure that $|\mu_S - \mathbb{E}[V]| \leq \frac{\varepsilon}{2}\mu_S$.

We now repeat this random process t times, and consider $V_1, \ldots, V_t$ the volume of the corresponding convex hulls. The Convex hull C of all these points is larger than each of the V_i, particularly larger than the mean $S_t = \frac{\sum_{i=1}^{t} V_i}{t}$. Applying the Chernoff bound, we know that:

$$Pr_\Omega[|\frac{S_t}{\mu_S} - \frac{\mathbb{E}[V]}{\mu_S}| \leq a] \geq 1 - 2e^{-2a^2.t}$$

Furthermore, $S_t \leq \mu_C \leq \mu_S$ In order to obtain $|\mu_C - \mathbb{E}[V]| \leq \frac{\varepsilon}{2}\mu_S$, we take $a = \frac{\varepsilon}{2}$. Then $|\mu_C - \mu_S| \leq \varepsilon\mu_S$, and the desired success probability is realized with $t = \frac{1}{\varepsilon^2} \ln \frac{1}{\delta}$. ■

Notice however that one has to effectively compute the convex hull of these N random points, which is known to be an exponential process in the dimension (roughly $O(N^{\lfloor \frac{d}{2} \rfloor})$)([?]). Consider now the relation T defined by the query ϕ on an observable convex relation S in dimension $d + e$:

$$\phi(x_1, \ldots, x_e) \equiv \exists x_{e+1} \exists x_{e+2} \ldots \exists x_{e+d}\, R(x_1, \ldots, x_{d+e})$$

The query ϕ expresses a projection on a e-dimensional subspace of $\mathbb{R}^{d+e}$. Its standard implementation in constraint databases is the Fourier-Motzkin algorithm ([?]) whose complexity is $O(2^{2^k})$, where k is the number of projected variables. In our example, this is 2^{2^d}. In order to get an asymptotic speed-up, consider the following algorithm:

ALGORITHM 3.

1. Generate N random points uniformly in the projection of S with the projection generator.
2. Form the convex hull of these points.

PROPOSITION 4.1. *An (ε, δ)-estimation of the relation T defined by ϕ can be obtained in $O(2^{\frac{e}{2}}.poly(d + e))$ computation steps.*

Proof. Step ?? is polynomial in $d+e$ (and $\varepsilon, \ln \frac{1}{\delta}$) if it uses our projection generator. Step ?? takes exponential time, by any classical convex hull computation, *but in the resulting dimension e.* ■

4.2.2. General sets

The set defined by a first-order formula may not be convex. We show how to generalize the previous approach: we will only guarantee the approximation of the result when we can compute an approximate volume. Consider a simple example where R_1, R_2, R_3 are given well-bounded convex relations in dimension 2. Let T be the relation defined by the formula $\exists z[(R_1(x,z) \wedge R_2(z,y)) \vee R_4(x,z)]$

An approximation of the result could be obtained by taking the convex hull C_1 of $(R_1(x,z) \wedge R_2(z,y))$ using the uniform generator for the intersection and projecting it on z into C_1'. Similarly we could compute the convex hull C_2 of $R_4(x,z)$ using the Dyer-Frieze-Kannan uniform generator for R_4 and project it on z into C_2'. The result would be the union of C_1' and C_2' but we would not guarantee the approximation because the simple projection of a uniform generator is not necessarily uniform. We would also compute convex hulls in dimension 3 which is not necessary. We can modify our procedure as follows:

ALGORITHM 4 (GUARANTEED APPROXIMATION OF T).

1. Generate uniform points in $\exists z(R_1(x,z) \wedge R_2(z,y))$ (combining the uniform generation for the intersection and the projection), and take their convex hull D_1.
2. Generate uniform points in $\exists z R_4(x,z)$ and take their convex hull D_2.
3. The result is the union of D_1 and D_2.

In order to generate points uniformly in $\exists z(R_1(x,z) \wedge R_2(z,y))$ we still need the condition: R_1 and R_2 are poly-related. Notice that the approximation of D_1 and of D_2 is now guaranteed because we select N uniform points in a polytope. The implicit approach for the general reconstruction of an existential positive formula Ψ is the following algorithm:

ALGORITHM 5 (GUARANTEED APPROXIMATION OF AN EXISTENTIAL POSITIVE FORMULA Ψ).

1. Write the formula as the disjunction of conjunctions and projections: $\bigvee_i \varphi_i$ where each φ_i is built from atomic formulas by conjunction and existential quantification.
2. Generate uniform points in the sets defined by each φ_i (with the techniques of the previous section) and take their convex hull D_i.
3. The result is the union of the D_i.

THEOREM 4.1. *Let Ψ be an existential positive formula equivalent to $\bigvee_i \varphi_i$ where φ_i is obtained by conjunction and projection. For each constraint database, if there is a uniform generator for the set defined by each formula φ_i, then the set defined by the algorithm 4 is an (ε, δ)-estimator for the set defined Ψ.*

Proof. Apply the projection technique to the intersection of convex sets in order to obtain uniform generators for the sets defined by each φ_i. Applying the previous method for convex sets, we obtain an (ε, δ)-estimator for the set defined by φ_i. The union of the estimates is also an (ε, δ)-estimator for the set defined by Ψ. ■

In a case of a negation (or difference), we may still get a uniform generator but the reconstruction is more difficult, since convexity is lost.

4.3. Polynomial constraints

The Dyer-Frieze-Kannan generator supposes only the existence of a membership oracle for a convex set. This oracle can be easily computed for generalized relations given by polynomial constraints if they are convex (the conjunction of polynomial constraints does not necessarily define a convex set.)

Our generators and volume estimators for the union, intersection, difference and projection do not rely on the linearity and will generalize to polynomial observable sets.

Reconstruction of convex sets defined by polynomials involves complicated techniques like interpolation. But if one accepts a good approximation by a simple polytope (i.e. a set defined by linear-only inequations), our previous reconstruction method can be considered.

Let $\mathcal{G}_p$ be a γ-grid for a convex set S. The Dyer-Frieze-Kannan estimator approximates the size of a set $V = \mathcal{G}_p \cap S$, by generating almost uniform points in V. Consider now the polytope $hull(V)$, defined as the convex hull of points in V. Clearly, the estimator does not distinguish between S and $hull(V)$, and their volume are closely related.

Hence, if $hull(V)$ has a given number of vertices r, one can use the algorithm of lemma **??** to approximate the convex S by a convex polytope.

LEMMA 4.2. *If $r = poly(d, \frac{1}{\varepsilon})$, the relation estimator for $hull(V)$ is a relation estimator for S.*

We suspect that for smooth convex bodies defined by polynomial constraints of fixed degree, r always satisfies the previous conditions.

5. CONCLUSION

We studied how to approximate queries in constraint databases with the use of random sampling. We showed how to combine the basic uniform generator of Dyer-Frieze-Kannan for convex sets with the classical logical operators. We obtain uniform generators for the union and the projection of convex sets and gave a sufficient condition for the intersection and the difference.

A uniform generator also yields a method to approximate the volume in polynomial time and gives a general method to approximate queries as a combination

of convex hulls. The algorithms for the uniform generation and for the volume are randomized and polynomial in the size of the parameters. The reconstruction of a d-ary query is also exponential in $d/2$.

ACKNOWLEDGMENT

We would like to thank David Applegate, Luc Segoufin and Emmanuel Waller for many helpful discussions.

REFERENCES

1. F. Affentranger and J. A. Wieacker. On the convex hull of uniform random points in a simple d-polytope. *Discrete Comput. Geom.*, 6:291–305, 1991.
2. I. Bárány and Z. Füredi. Computing the volume is difficult. In *Proceedings of the Eighteenth Annual ACM Symposium on Theory of Computing*, pages 442–447, 1986.
3. M. Benedikt, G. Dong, L. Libkin, and L. Wong. Relational expressive power of constraint query languages. In *JACM*, 45(1):1–34, 1998.
4. M. Benedikt and L. Libkin. Languages for relational databases over interpreted structures. In *JACM*, 47(4):644–680, 2000.
5. M. Benedikt and L. Libkin. Safe constraint queries. In SIAM J. Comput., 29(5):1652–1682, 2000.
6. M. Benedikt and L. Libkin. Exact and approximate aggregation in constraint query languages. In *Symposium on Principles of Databases Systems*, pages 102–113, 1999.
7. W. Cochran. *Sampling Techniques*. Wiley, 1977.
8. M. Dyer, A. Frieze, and R. Kannan. A random polynomial time algorithm for approximating the volume of convex bodies. *Journal of the ACM*, 38:1–17, 1991.
9. G. Elekes. A geometric inequality and the complexity of computing volume. *Discr. Comput. Geom*, 1, 1986.
10. D. M. Flewelling. *Comparing Subsets from Digital Spatial Archives: Point Set Similarity*. PhD thesis, University of Maine, 1997.
11. M. Grötschel, L. Lovász, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization*. Springer-Verlag, 1988.
12. S. Grumbach, P. Rigaux, and L. Segoufin. On the orthographic dimension of constraint databases. In *ICDT*, volume 1540 of *Lecture Notes in Computer Science*, pages 199–216, 1999.
13. S. Grumbach and J. Su. Finitely representable databases. *Journal of Computer and System Sciences*, 55(2):273–298, 1997.
14. S. Grumbach and J. Su. Queries with arithmetical constraints. *Theoretical Computer Science*, 173(1):151–181, 1997.
15. S. Grumbach, J. Su, and C. Tollu. Linear constraint query languages : Expressive power and complexity. In D. Leivant, editor, *Logic and Computational Complexity*, volume 960 of *LNCS*. Springer Verlag, 1994.
16. M. Gyssens, J. V. den Bussche, and D. V. Gucht. Complete geometric query languages. *Journal of Computer and System Sciences*, 58(3):483–511, 1999.
17. W.-C. Hou, G. Özsoyoglu, and B. K. Taneja. Statistical estimators for relational algebra expressions. In *Symposium on Principles of Databases Systems*, pages 276–287, 1988.
18. M. Jerrum, L. Valiant, and V. Vazirani. Random generation of combinatorial structures from a uniform generation. *Theoretical Computer Science*, 43:169–188, 1986.
19. P. C. Kanellakis, G. M. Kuper, and P. Z. Revesz. Constraint query languages. *Journal of Computer and System Sciences*, 51(1):26–52, Aug. 1995.
20. Ravi Kannan, László Lovász, and Miklos Simonovits. Random walks and an $O^*(n^5)$ volume algorithm for convex bodies. *Random Structures and Algorithms*, 11(1):1–50, 1997.
21. R. M. Karp and M. Luby. Monte-carlo algorithms for enumeration and reliability problems. *Proceedings of the 24th Symposium on Foundations of Computer Science*, pages 56–64, 1983.

22. M. Karpinski and A. Macintyre. Approximating the volume of general Pfaffian bodies. In *Structures in Logic and Computer Science*, volume 1261 of *Lecture Notes in Computer Science*, pages 162–173, 1997.

23. P. Koiran. Approximating the volume of definable sets. In *36th Symposium on Foundations of Computer Science*, pages 134–141, 1995.

24. R. J. Lipton and J. F. Naughton. Query size estimation by adaptive sampling. *Journal of Computer and System Sciences*, 51(1):18–25, 1995.

25. F. Olken and D. Rotem. Random sampling form database files: A survey. In Z. Michalewicz, editor, *Fifth International Conference on Statistical and Scientific Database Management*, 1990.

26. F. Olken and D. Rotem. Sampling from spatial databases. In *International Conference on Data Engineering*, pages 199–208, 1993.

27. J. Paredaens, J. V. den Bussche, and D. V. Gucht. Towards a theory of spatial database queries. In *Symposium on Principles of Databases Systems*, pages 279–288, 1994.

28. A. Schrijver. *Theory of linear and integer programming.* Wiley, 1986.

29. Van Leuwen, editor. *Handbook of Theoretical Computer Science, vol A.* Elsevier, 1990.

30. L. Vandeurzen, M. Gyssens, and D. V. Gucht. An expressive language for linear spatial database queries. In *Symposium on Principles of Databases Systems*, 1998.