Variable Independence for First-Order Definable Constraints

LEONID LIBKIN
University of Toronto

Whenever we have data represented by constraints (such as order, linear, polynomial, etc.), running time for many constraint processing algorithms can be considerably lowered if it is known that certain variables in those constraints are independent of each other. For example, when one deals with spatial and temporal databases given by constraints, the projection operation, which corresponds to quantifier elimination, is usually the costliest. Since the behavior of many quantifier elimination algorithms becomes worse as the dimension increases, eliminating certain variables from consideration helps speed up those algorithms.

While these observations have been made in the literature, it remained unknown when the problem of testing if certain variables are independent is decidable, and how to construct efficiently a new representation of a constraint-set in which those variables do not appear together in the same atomic constraints. Here we answer this question. We first consider a general condition that gives us decidability of variable independence; this condition is stated in terms of model-theoretic properties of the structures corresponding to constraint classes. We then show that this condition covers the domains most relevant to spatial and temporal applications. For some of these domains, including linear and polynomial constraints over the reals, we provide a uniform decision procedure which gives us tractability as well. For those constraints, we also present a polynomial-time algorithm for producing nice constraint representations.

Categories and Subject Descriptors: F.2.1 [**Analysis of Algorithms and Problem Complexity**]: Numerical Algorithms and Problems; F.4.1 [**Mathematical Logic and Formal Languages**]: Mathematical Logic; H.2.8 [**Database Management**]: Database Applications; I.1.1 [**Symbolic and Algebraic Manipulation**]: Expressions and Their Representation

General Terms: Languages, Theory

Additional Key Words and Phrases: First-order logic, definable sets, variable independence, polynomial constraints, linear constraints, spatio-temporal databases

1. INTRODUCTION

We start with a simple example. Suppose we have a set $S \subseteq \mathbb{R}^2$ given by simple order-constraints $\varphi(x,y) = (0 < x < 1) \wedge (0 < y < 1)$. Suppose we want to find its projection on the x axis. This means writing the formula $\exists y\ \varphi(x,y)$ as a quantifier-free formula. This can be done, in general, because the theory of

Preliminary version of this paper appeared in the 27th International Colloquium on Automata, Languages and Programming (ICALP'2000). Address: Department of Computer Science, 6 King's College Road, University of Toronto, Ontario M5S 3H5, Canada, email: libkin@cs.toronto.edu.
Research affiliation: Bell Laboratories.

$\langle \mathbb{R}, <, (r)_{r\in\mathbb{R}} \rangle$ admits quantifier elimination. But in this particular case it is very easy to find a quantifier-free formula equivalent to $\exists y\ \varphi(x,y)$ using just standard rules for equivalence of first-order formulae:

$$\exists y\ \varphi(x,y) \ \leftrightarrow\ (0 < x < 1) \wedge \exists y\ (0 < y < 1) \ \leftrightarrow\ (0 < x < 1) \wedge \text{true} \ \leftrightarrow\ 0 < x < 1.$$

Now notice that φ can be considered as a formula in the language of the real field $\langle \mathbb{R}, +, \cdot, 0, 1, < \rangle$ whose theory also admits quantifier elimination. Suppose then that instead of φ, we are given an equivalent formula $\psi(x,y)$:

$$\begin{array}{rl} & \big((0 < x < 1) \wedge (0 < y < 1) \wedge (4x^2 - y - 1 \geq 0)\big) \\ \vee & \big((0 < x < 1) \wedge (0 < y < 1) \wedge (4x^2 - y - 1 \leq 0)\big). \end{array} \tag{1}$$

The first step of quantifier elimination for $\exists y\ \psi$ is easy, as we propagate $\exists y$ inside the disjunction. However, trying to find a quantifier-free equivalent for the first disjunct, that is, a formula equivalent to $\exists y\ \big((0 < x < 1) \wedge (0 < y < 1) \wedge (4x^2 - y - 1 \geq 0)\big)$, one immediately encounters obstacles. Unlike the earlier example, this one requires a bit of thought to come up with the answer $(0.5 \leq x < 1)$. Similarly, some work is needed to compute the answer $(0 < x \leq 1/\sqrt{2})$ for the second disjunct.

Why is it that the first quantifier-elimination procedure is completely elementary, and the second is not, even though both φ and ψ define the same set? The reason is that in the first representation of S, variables x and y are independent, that is, they do not appear in the same atomic formulae. This makes quantifier elimination easy. In the second case, x and y do appear together in the same term $x^2 - 4y - 1$, and this is what causes the problem.

This extremely simple observation can often make constraint processing easier. While it can conceivably be useful in various tasks such as more efficient variable elimination in constraint logic programming [Fordan and Yap 1998; Imbert 1994], here we concentrate on one application area, namely *constraint databases* [Kuper et al. 2000; Kuper et al. 1995] where it found its way into a practical system for querying spatio-temporal databases [Grumbach et al. 1998]. The main goal of constraint databases is to model infinite database objects, which arise in a variety of applications, for example, in Geographical Information Systems.

A particular constraint model is defined over a structure $\mathcal{M} = \langle U, \Omega \rangle$ (where U is the universe and Ω is the vocabulary) which is typically required to have quantifier elimination. Those considered most often in spatial application are the real field $\mathbf{R} = \langle \mathbb{R}, +, \cdot, 0, 1, < \rangle$ and the real ordered group $\mathbf{R}_{\text{lin}} = \langle \mathbb{R}, +, -, 0, 1\ < \rangle$, which give rise to polynomial and linear constraint databases, respectively. A constraint relation of arity n is simply a definable subset of U^n, that is, a set of tuples $\vec{a} \in U^n$ that satisfy a first-order formula. For the above structures, constraint relations are *semi-algebraic* sets for $\mathbf{R}$, and *semi-linear* sets for $\mathbf{R}_{\text{lin}}$ [Bochnak et al. 1998]. A constraint database is a finite set of constraint relations.

A standard constraint query language over $\mathcal{M}$ is $\text{FO} + \mathcal{M}$, that is, first-order logic in the language of $\mathcal{M}$ and symbols for relations in a constraint database. For example, if a database contains a single ternary symbol S, the query $\varphi(x) \equiv \exists u, v\ \forall y, z\ (S(x,y,z) \leftrightarrow z = u \cdot y + v)$ finds all a such that the intersection of S with the plane $x = a$ is a line. Note that if S is a semi-algebraic set, then so is $\varphi(S)$.

One of the standard database operations is projection. In the language of constraint processing, it corresponds to quantifier elimination. That is, given a quantifier-free formula $\varphi(y, x_1, \ldots, x_{n-1})$, one wishes to find a quantifier-free formula $\psi(\vec{x})$ equivalent to $\exists y\ \varphi(y, \vec{x})$. In many cases, the complexity of algorithms to find such a ψ is of the form $O(N^{f(n)})$, where N is the size of the formula, and f is some function. For example, if one uses cylindrical algebraic decomposition [Caviness and Johnson 1998] for the real field, f is $O(2^n)$. In general, even if better algorithms are available, the complexity of constraint processing often increases with dimension to such an extent that it becomes unmanageable for large datasets (see, e.g., [Grumbach et al. 1999]).

Assume now that $\vec{x}$ is split into two disjoint tuples $\vec{u}$ and $\vec{v}$ such that $(y, \vec{u})$ and $\vec{v}$ are independent, that is, they do not appear in the same atomic formulae. Then φ is equivalent to a formula of the form

$$\bigvee_{i=1}^{k} \alpha_i(y, \vec{u}) \wedge \beta_i(\vec{v}).$$

Therefore, the formula $\exists y\ \varphi$ is equivalent to

$$\bigvee_{i=1}^{k} (\exists y\ \alpha_i(y, \vec{u})) \wedge \beta_i(\vec{v}).$$

For a number of operations this is a significant improvement, as the exponent becomes lower. For example, in addition to quantifier elimination, data often has to be represented in a nice format (essentially, as union of cells [Caviness and Johnson 1998]), and algorithms for doing this also benefit from reduction in the dimension [Grumbach et al. 1998; 1999].

Even though such a notion of independence may seem to be too much of a restriction, from the practical point of view it is sometimes necessary to insist on it, as the cost of general quantifier elimination and other operations could be prohibitively expensive. For example, the DEDALE constraint database system [Grumbach et al. 1998] requires that the projection operation only be applied when $\vec{u}$ consists of a single variable. Dealing with spatio-temporal applications, one often queries trajectories of objects, or cadastral (land-ownership) information. These are typically represented as objects in $\mathbb{R}^3$ given by formulae $\varphi(x, y, t)$. To be able to compute $\exists y\ \varphi(x, y, t)$, one approximates φ by a formula $\psi(x, y, t)$ which is a Boolean combination of formulae $\alpha_i(x, y)$ and $\beta_i(t)$. For trajectories, this amounts to saying that an object is in a given region during a given interval of time; thus, it is the information about the speed that is lost in order to have efficient query evaluation. As was further demonstrated in [Grumbach et al. 1999], the difference between the case when at most 2 variables are dependent, and that of 3 or more variables being dependent, is quite dramatic, in the case of linear and polynomial constraints.

What is missing, however, in this picture, is the ability to determine whether a given constraint representation of the data can be converted to the one in the right format, just as in our first example, $\psi(x, y)$ is equivalent to $\varphi(x, y)$, in which variables x and y are independent. It was claimed in [Chomicki et al. 1996] that such a procedure exists for linear constraints, and then [Grumbach et al. 1999] gave a simpler algorithm. However, [Libkin 1999] then showed that both claims were incorrect. It was thus not known if variable independence can be tested for relevant classes of constraints.

Our main goal here is to show that variable independence can be tested for many classes of constraints, and that algorithms for converting a given formula into one in the right form can be obtained. Moreover, those algorithms often work in time polynomial in the size of the formula (assuming the total number of variables is fixed). Among structures for which we prove such results are the real ordered group, the real field, as well as $\langle \mathbb{Z}, +, 0, 1, < \rangle$ extended with all the relations $x = y(\text{mod } k)$, $k > 1$ (which is used in temporal applications). Even if those algorithms are relatively expensive, it is worth putting data in a nice format for two reasons. First, such an algorithm works only once, and then the data is repeatedly queried by different queries, which can be evaluated faster. Second, some queries are known to preserve variable independence; hence, this information can be used for further processing the query output.

Organization. In Section 2, we define the notion of variable independence, and more generally, the notion $\varphi \sim P$ of a formula φ respecting a certain partition P of its free variables. Then, in Section 3, we discuss requirements on the theory of $\mathcal{M}$ that guarantee decidability of this notion, as well as the existence of an algorithm that converts a given formula into a one in the right shape. In Section 4, we discuss specific classes of structures and derive some complexity bounds. In particular, we look at *o-minimal* structures [van den Dries 1998] (which include linear and polynomial constraints over the reals) and give a *uniform* decision procedure. This procedure gives us tractability, and we also show how to find an equivalent formula in the right shape in polynomial time. We also briefly consider other classes of constraints, and spatio-temporal applications.

2. NOTATIONS

All the definitions can be stated for arbitrary first-order structures, although for the algorithmic considerations we shall require at least decidability of the theory, and often quantifier elimination.

Given a structure $\mathcal{M} = \langle U, \Omega \rangle$ (where U is a set always assumed to be infinite, and Ω can contain predicate, function, and constant symbols, and is always assumed to be a recursive set), we say that the theory of $\mathcal{M}$ is decidable if for every first-order sentence Φ in the language of $\mathcal{M}$ it decidable if $\mathcal{M} \models \Phi$. We say that $\mathcal{M}$ admits (effective) quantifier elimination if for every formula $\varphi(\vec{x})$ in the language of $\mathcal{M}$, there exists (and can be effectively found) a quantifier-free formula $\psi(\vec{x})$ such that $\mathcal{M} \models \forall \vec{x}\ \varphi(\vec{x}) \leftrightarrow \psi(\vec{x})$.

Given a formula $\varphi(\vec{x}, \vec{y})$ in the language of $\mathcal{M}$, with $\vec{x}$ of length n and $\vec{y}$ of length m, and $\vec{a} \in U^n$, we write $\varphi(\vec{a}, \mathcal{M})$ for the set $\{\vec{b} \in U^m \mid \mathcal{M} \models \varphi(\vec{a}, \vec{b})\}$. In the absence of variables $\vec{x}$ we write $\varphi(\mathcal{M})$ for $\{\vec{b} \mid \mathcal{M} \models \varphi(\vec{b})\}$. Sets of the form $\varphi(\mathcal{M})$ are called *definable*. A function $f : U^n \to U^m$ is definable if its graph $\{(\vec{a}, \vec{b}) \in U^{n+m} \mid \vec{b} = f(\vec{a})\}$ is a definable set.

Given a tuple of variables $\vec{x} = (x_1, \ldots, x_n)$ and a partition $P = \{B_1, \ldots, B_m\}$ on $\{1, \ldots, n\}$, we let $\vec{x}_{B_i}$ stand for the subtuple of $\vec{x}$ consisting of the x_js with $j \in B_i$. For a formula $\varphi(x_1, \ldots, x_n)$, we then say that φ *respects the partition* P (over $\mathcal{M}$) if φ is equivalent to a Boolean combination of formulae each having its free variables among $\vec{x}_{B_i}$ for some $i \leq k$. This will be written as $\varphi \sim_{\mathcal{M}} P$, or just $\varphi \sim P$ if $\mathcal{M}$ is clear from the context.

In other words (by putting a Boolean combination into DNF), $\varphi \sim_{\mathcal{M}} P$ if there exists a family of formulae $\alpha^i_j(\vec{x}_{B_i})$, $i = 1, \ldots, m$, $j = 1, \ldots, k$, such that

$$\mathcal{M} \models \varphi(\vec{x}) \leftrightarrow \bigvee_{j=1}^{k} (\alpha^1_j(\vec{x}_{B_1}) \wedge \ldots \wedge \alpha^m_j(\vec{x}_{B_m})). \tag{2}$$

When $\mathcal{M}$ has quantifier elimination, all α^i_js are quantifier free. In fact, under the quantifier-elimination assumption, the definition of $\varphi \sim_{\mathcal{M}} P$ can be restated as the equivalence of φ to a quantifier-free formula ψ such that every atomic subformula of ψ uses variables from only one block of P.

We now say that in φ, two variables x_i and x_j are *independent* if there exists a partition P such that $\varphi \sim_{\mathcal{M}} P$, and x_i and x_j are in two different blocks of P. Equivalently, x_i and x_j are independent if there exists a partition $P = (\vec{y}, \vec{z})$ of $\vec{x}$ such that $\varphi \sim_{\mathcal{M}} P$, x_i is in $\vec{y}$ and x_j is in $\vec{z}$. (When convenient notationally, we identify partitions on the indices of variables and variables themselves.)

Structures. After presenting a general decidability result, we shall deal with several important classes of structures. Two of them were mentioned already: the real ordered group $\mathbf{R}_{\rm lin} = \langle \mathbb{R}, +, -, 0, 1, < \rangle$ and the real field $\mathbf{R} = \langle \mathbb{R}, +, \cdot, 0, 1, < \rangle$, corresponding to linear and polynomial constraints over the reals. Some of the results for these structures extend to a larger class of *o-minimal* structures: $\mathcal{M} = \langle U, \Omega \rangle$ is called o-minimal [Pillay and Steinhorn 1988; van den Dries 1998] if one of the symbols in Ω is $<$, interpreted as a linear order on U, and every definable subset of U, $\{a \mid \mathcal{M} \models \varphi(a)\}$, is a finite union of points and open intervals. Both $\mathbf{R}_{\rm lin}$ and $\mathbf{R}$ have quantifier elimination (by Fourier elimination [Ziegler 1994], and Tarski's theorem [Bochnak et al. 1998; Caviness and Johnson 1998], respectively), which easily implies that they are o-minimal. The exponential field $\langle \mathbb{R}, +, \cdot, e^x \rangle$ is an example of a structure which is o-minimal [Wilkie 1996] but does not have quantifier elimination [van den Dries 1984]. For other o-minimal structures on the reals, see [van den Dries 1998].

We shall deal with some structures on the integers. Of most interest to us is $\mathcal{Z}_0 = \langle \mathbb{Z}, +, -, 0, 1, <, (\equiv_k)_{k>1} \rangle$ where $n \equiv_k m$ iff $n = m (\text{mod } k)$. This structure corresponds to constraints given by linear repeating points, which are used for modeling temporal databases [Kabanza et al. 1995]. The structure $\mathcal{Z}_0$ admits effective quantifier elimination, and its theory is decidable [Enderton 1972].

3. GENERAL CONDITIONS FOR DECIDING VARIABLE INDEPENDENCE

Given a structure $\mathcal{M}$, we consider two problems. The *variable independence problem* $\mathrm{VI}_{\mathcal{M}}(\varphi, x_i, x_j)$ is to decide, for $\varphi(x_1, \ldots, x_n)$ in the language of $\mathcal{M}$, if x_i and x_j are independent. The *variable partition problem* $\mathrm{VP}_{\mathcal{M}}(\varphi, P)$ is to decide, for a given formula $\varphi(x_1, \ldots, x_n)$ and a partition P on $\{1, \ldots, n\}$, if $\varphi \sim_{\mathcal{M}} P$.

Note that the variable independence problem is a special case of the variable partition problem, as to solve the former, one needs to solve the latter for some partition $P = (B_1, B_2)$ with $i \in B_1$ and $j \in B_2$.

The above problems are just decision problems, but if the theory of $\mathcal{M}$ is decidable, and the answer to $\mathrm{VP}_{\mathcal{M}}(\varphi, P)$ is 'yes', one can effectively find a representation in the form (2), simply by enumerating all the formulae $\langle\psi(\vec{x})\rangle_i$ which are Boolean combinations of formulae having free variables from at most one block of P, and then checking if $\mathcal{M} \models \forall\vec{x}\ (\varphi(\vec{x}) \leftrightarrow \psi_i(\vec{x}))$. Since $\varphi \sim_{\mathcal{M}} P$, for some finite i, we get a positive answer. In many interesting cases, we shall see better algorithms for finding representation (2) than simple enumeration.

The first easy result shows that the problems $\mathrm{VI}_{\mathcal{M}}(\varphi, x_i, x_j)$ and $\mathrm{VP}_{\mathcal{M}}(\varphi, P)$ are equivalent; this allows us to deal then only with two-block partitions.

Lemma 1. *For any $\mathcal{M}$, the variable independence problem is decidable over $\mathcal{M}$ iff the variable partition problem is decidable over $\mathcal{M}$.*

Proof. The direction from variable partition to variable independence is easy. For the other direction we show that the answer to $\mathrm{VP}_{\mathcal{M}}(\varphi, P)$ is yes iff the answer to $\mathrm{VI}_{\mathcal{M}}(\varphi, x_i, x_j)$ is yes for every pair of variables x_i, x_j from two different blocks of P. Again, only the 'if' direction needs to be proved. If the answer to $\mathrm{VI}_{\mathcal{M}}(\varphi, x_i, x_j)$ is yes, there is a partition P_{ij} such that x_i, x_j are in different blocks of P, and $\varphi \sim_{\mathcal{M}} P_{ij}$. Let $P' = \sqcap_{ij} P_{ij}$ be the meet, in the partition lattice, of all such P_{ij}s. By [Cosmadakis et al. 2001], we have $\varphi \sim_{\mathcal{M}} P'$. Since every block of P is a union of blocks of P', the result follows. □

Next, we discuss conditions for decidability of the variable independence problem. It is clear that one needs decidability of the theory of $\mathcal{M}$. However, decidability alone (and even effective quantifier elimination) are not sufficient.

Proposition 2. *a) If the theory of $\mathcal{M}$ is undecidable, then the variable independence problem is undecidable over $\mathcal{M}$.*
b) There exists a structure $\mathcal{M}$ with a decidable theory and effective quantifier elimination such that the variable independence problem is undecidable over $\mathcal{M}$.

Proof. a) Let Φ be an arbitrary sentence, and $\varphi(x, y) \equiv (x = y) \wedge \neg\Phi$. If $\mathcal{M} \models \Phi$, then φ defines the empty set, and x and y are thus independent. If $\mathcal{M} \models \neg\Phi$, then φ defines $\{(a, a) \mid a \in U\}$, and since U is infinite, x and y are not independent. Thus, the answer to $\mathrm{VI}_{\mathcal{M}}(\varphi, x, y)$ is yes iff $\mathcal{M} \models \Phi$.

b) An example is provided by the theory of traces from [Stolboushkin and Tsaitlin 1999]. Let U be a union of three disjoint sets: descriptions of Turing machines, input words, and traces, or partial computations of machines on input words, all appropriately coded as strings. Let Ω contain a constant symbol for every element of U, and a single ternary predicate $P(m, w, t)$ saying that t is a trace of the machine m on the input word w. Then [Stolboushkin and Tsaitlin 1999] shows that the theory of $\mathcal{M}$ is decidable, and moreover, $\mathcal{M}$ can be extended by finitely many new predicate symbols such that the expanded model has effective quantifier elimination.

Now fix a Turing machine m_0 and an input word w_0 and consider the formula $\varphi(t, t') \equiv (P(m_0, w_0, t) \wedge t = t')$. Suppose m_0 halts on w_0. Then the set $\{t \mid P(m_0, w_0, t)\}$ is finite, and thus the output of φ is finite, and hence variables t and t' are independent, since every element of U is definable. If m_0 does not halt on w_0, then the set $U' = \{t \mid P(m_0, w_0, t)\}$ is infinite, and hence the output of φ is $\{(a, a) \mid a \in U'\}$; this implies that t and t' are not independent. This shows that $\mathrm{VI}_{\mathcal{M}}(\varphi, t, t')$ outputs yes iff m_0 halts on w_0, and thus the variable independence problem is undecidable. □

The proof of Proposition 2, b), shows that it is essential to be able to decide finiteness in order to decide $\mathrm{VI}(\varphi, x_i, x_j)$ (as it is the finiteness of the number of traces that turns out to be equivalent to variable independence).

Recall that a formula $\varphi(x)$ is *algebraic* if $\varphi(\mathcal{M})$ is finite. We say that there is an *effective test for algebraicity* in $\mathcal{M}$ if for every $\varphi(x)$ in the language of $\mathcal{M}$, it is decidable if φ is algebraic. Note that this somewhat technical notion will trivially hold for most relevant classes of constraint.

While the notion of variable independence is needed in the context of constraint databases, for finite relational structures it is assumed to be meaningless as every tuple is represented as a conjunction of constraints of the form $x_i = c_i$, where c_is are constants. For example, the graph $\{(1,2),(3,4)\}$ is given by the formula $((x=1)\wedge(y=2))\vee((x=3)\wedge(y=4))$. Clearly, variables x and y are independent.

However, over arbitrary structures, not every finite definable set would satisfy the variable independence condition. To see this, let $\mathcal{M} = \langle \mathbb{N}, C, E\rangle$, where C is a unary relation interpreted as $\{1,2\}$ and E is a binary relation symbol interpreted as $\{(1,2),(2,1)\}$. A routine argument shows that this $\mathcal{M}$ has quantifier elimination, decidable theory, and there is a test for algebraicity. The formula $\varphi(x,y) \equiv E(x,y)$ then defines a finite set, but variables x and y are not independent: this is because the only definable proper subsets of $\mathbb{N}$ are $\{1,2\}$ and $\mathbb{N} - \{1,2\}$, and no Boolean combination of those gives us E. As another example, consider the field of complex numbers, whose theory is decidable and has quantifier elimination [Marker et al. 1996]. Let $\varphi(x,y) = (x^2+1=0)\wedge(y^2+1=0)\wedge(x+y=0)$. It defines the finite set $\{(i,-i),(-i,i)\}$ but nevertheless x and y are not independent (since i is not definable).

To avoid similar situations, we impose an extra condition on a structure, again, well known in model theory [Chang and Keisler 1990; Hodges 1993]. We say that $\mathcal{M}$ has *definable Skolem functions* if for every formula $\varphi(\vec{x},\vec{y})$ there exists a definable function $f_\varphi(\vec{x})$ with the property that $\mathcal{M} \models \forall\vec{x}\ (\exists\vec{y}\ \varphi(\vec{x},\vec{y}) \to \varphi(\vec{x}, f_\varphi(\vec{x})))$. In other words, $f_\varphi(\vec{a})$ is an element of $\varphi(\vec{a},\mathcal{M})$, assuming $\varphi(\vec{a},\mathcal{M})$ is not empty. We say that a Skolem function f_φ is *invariant* [Marker et al. 1996], if $\varphi(\vec{a}_1,\mathcal{M}) = \varphi(\vec{a}_2,\mathcal{M})$ implies $f_\varphi(\vec{a}_1) = f_\varphi(\vec{a}_2)$. If the existence of such a Skolem function can be guaranteed for every φ, we say that $\mathcal{M}$ has definable invariant Skolem functions.

THEOREM 3. *Assume that $\mathcal{M}$ has the following properties:*

(a) its theory is decidable;

(b) $\mathcal{M}$ has effective test for algebraicity; and

(c) $\mathcal{M}$ has definable invariant Skolem functions.

Then the variable partition and independence problems are decidable over $\mathcal{M}$.

Proof. Let $\mathcal{M}$ be as in the statement of the theorem. We start by showing certain properties of $\mathcal{M}$ that will be needed in the proof. First notice that the definability of Skolem functions is effective; that is, for each φ, a formula defining f_φ can be effectively found. To see this, just enumerate all formulae and test if they define a function, and if this function is a Skolem function for φ. Since the above is a first-order sentence (invariance is tested by $\forall\vec{x}_1\forall\vec{x}_2\ ((\forall\vec{y}\ \varphi(\vec{x}_1,\vec{y}) \leftrightarrow \varphi(\vec{x}_2,\vec{y})) \to (f_\varphi(\vec{x}_1) = f_\varphi(\vec{x}_2))))$, effective definability follows.

A formula $\varphi(\vec{x})$ (with one or more free variables) is algebraic if $\varphi(\mathcal{M})$ is finite. We can assume that we have test for algebraicity for formulae with more than one free variable: indeed, a set $X \subseteq U^k$ is finite iff each of its k projections on U is finite.

Next, we show that $\mathcal{M}$ has in addition the following property (d): if $\varphi(\vec{x})$ is algebraic, then one can effectively find $N = \mathit{card}(\varphi(\mathcal{M}))$, and N formulae $\gamma_i(\vec{x})$, $i = 1, \ldots, N$, such that $\mathit{card}(\gamma_i(\mathcal{M})) = 1$, and $\varphi(\mathcal{M}) = \{\gamma_i(\mathcal{M}) \mid i \leq N\}$ (that is, each element of $\varphi(\mathcal{M})$ is definable, and formulae defining those elements can be effectively found).

To see this, first note that for every N it can be stated in first-order that $N = \mathit{card}(\varphi(\mathcal{M}))$, and thus by decidability we can find N, assuming φ is algebraic. Next we use effective definability of Skolem functions (without parameters) to construct a formula $\gamma_1(\vec{x})$ defining an element of $\varphi(\mathcal{M})$. We then consider $\varphi(\vec{x}) \wedge \neg\gamma_1(\vec{x})$, and apply Skolemization to it, to obtain $\gamma_2(\vec{x})$, defining an element in $\varphi(\mathcal{M}) - \gamma_1(\mathcal{M})$. We continue the process until $\gamma_i(\vec{x}), i < N$ are defined; then $\gamma_N(\vec{x}) = \varphi(\vec{x}) \wedge \bigwedge_{i<N} \neg\gamma_i(\vec{x})$.

Having done this preparatory work, we now prove the theorem. Recall that it suffices to consider the case of two block partitions; that is, to decide, if a formula $\varphi(\vec{x}, \vec{y})$ respects the partition P with blocks $\vec{x}$ and $\vec{y}$. Let $\vec{x}$ have length n and $\vec{y}$ have length l. Define an equivalence relation on U^n by

$$\vec{a}_1 \equiv \vec{a}_2 \quad \text{iff} \quad \varphi(\vec{a}_1, \mathcal{M}) = \varphi(\vec{a}_2, \mathcal{M}).$$

LEMMA 4. *For φ, P and $\equiv$ as above, $\varphi \sim_{\mathcal{M}} P$ iff $\equiv$ has finitely many equivalence classes.*

Proof of the lemma. The only if part is clear: if φ is a Boolean combination of $\alpha_i(\vec{x})$, $\beta_j(\vec{y})$, then for every $\vec{a}_1, \vec{a}_2$ agreeing on all α_i, we have $\varphi(\vec{a}_1, \mathcal{M}) = \varphi(\vec{a}_2, \mathcal{M})$.

For the converse, assume that $\equiv$ has finitely many equivalence classes. Note that $\equiv$ is a definable subset of U^{2n} (it is defined by $\psi(\vec{x}_1, \vec{x}_2) = \forall\vec{y}\ (\varphi(\vec{x}_1, \vec{y}) \leftrightarrow \varphi(\vec{x}_2, \vec{y}))$). Assume that there are N equivalence classes, and each is definable by a formula $\alpha_i(\vec{x})$, $i \leq N$. Define $\beta_i(\vec{y})$ as $\exists\vec{z}\ (\alpha_i(\vec{z}) \wedge \varphi(\vec{z}, \vec{y}))$. Then

$$\mathcal{M} \models \forall\vec{x}\forall\vec{y}\ \Big(\varphi(\vec{x}, \vec{y}) \quad \leftrightarrow \quad \bigvee_{i=1}^{N} \alpha_i(\vec{x}) \wedge \beta_i(\vec{y})\Big). \tag{3}$$

Indeed, let $\mathcal{M} \models \varphi(\vec{a}, \vec{b})$, and assume that $\vec{a}$ is in the ith class of $\equiv$; that is, $\mathcal{M} \models \alpha_i(\vec{a})$. This implies $\mathcal{M} \models \beta_i(\vec{b})$. Conversely, if $\mathcal{M} \models \alpha_i(\vec{a}) \wedge \beta_i(\vec{b})$, for some $\vec{c}$ we have $\alpha_i(\vec{c}) \wedge \varphi(\vec{c}, \vec{b})$. Since $\varphi(\vec{c}, \mathcal{M}) = \varphi(\vec{a}, \mathcal{M})$, we have $\vec{b} \in \varphi(\vec{a}, \mathcal{M})$ and thus $\mathcal{M} \models \varphi(\vec{a}, \vec{b})$. This proves (3).

It thus remains to show how to define α_is. First, we find (effectively) the invariant Skolem function $f_\psi(\vec{x}_1)$ for the formula $\psi(\vec{x}_1, \vec{x}_2)$ defining $\equiv$. Then the formula $\chi(\vec{x}_2) = \exists\vec{x}_1\ (\vec{x}_2 = f_\psi(\vec{x}_1))$ defines the range of f_ψ, that is, a set of representatives of the equivalence classes of $\equiv$. By the assumption that the number of classes is finite, we get that χ is algebraic. Hence, by condition (d), we can find effectively the number N of elements satisfying χ (that is, the number of classes of $\equiv$), and formulae $\gamma_i(\vec{x})$, $i \leq N$, defining representatives of the equivalence classes. The equivalence classes themselves can now be defined as $\alpha_i(\vec{x}) = \exists\vec{z}\ (\gamma_i(\vec{z}) \wedge \psi(\vec{x}, \vec{z}))$. This concludes the proof of the lemma.

To prove the theorem, it remains to show how to test if $\equiv$ has finitely many equivalence classes. Following the proof of lemma 4, we effectively construct a formula $\chi(\vec{x})$ defining representatives of the equivalence classes of $\equiv$. Since $\equiv$ has finitely many equivalence classes iff $\chi(\vec{x})$ is algebraic, the former is decidable, by the assumptions we made about $\mathcal{M}$. This concludes the proof of decidability of the variable partition problem. $\square$

The proof of Theorem 3 gives an explicit construction for a formula witnessing $\varphi \sim_{\mathcal{M}} P$, where P has two blocks. We now show how it works for the case of formula $\psi(x, y)$ given by (1) in Section 1.

There are two equivalence classes with respect to relation $\equiv$ given by $x_1 \equiv x_2 \;\Leftrightarrow\; \psi(x_1, \mathbf{R}) = \psi(x_2, \mathbf{R})$: one is $C_1 = (-\infty, 0] \cup [1, \infty)$ and the other is $C_2 = (0, 1)$. Let $\alpha_1(x), \alpha_2(x)$ be formulae defining these classes. Then, from (3) we know that $\psi(x, y)$ is equivalent to

$$\Big(\alpha_1(x) \wedge \beta_1(y)\Big) \;\vee\; \Big(\alpha_2(x) \wedge \beta_2(y)\Big), \tag{4}$$

where

$$\beta_i(y) \;\; = \;\; \exists z \; (\alpha_i(z) \wedge \psi(z, y)), \; i = 1, 2. \tag{5}$$

By the decision procedure for $\mathbf{R}$, we obtain that β_1 is equivalent to *false*, and β_2 to $0 < y < 1$. Hence, combining (4) and (5), we see that (1) is equivalent to

$$(0 < x < 1) \;\wedge\; (0 < y < 1),$$

as expected.

The previous result was for two-block partitions. We now extend it to arbitrary partitions. using the special form of the formulae (3).

Let $\varphi(x_1, \ldots, x_n)$ be given, and let $B \subset \{1, \ldots, n\}$. Let $card(B) = k$. For $\vec{a} \in U^k$, by $\varphi_B(\vec{a}, \mathcal{M})$ we denote the set of $\vec{b} \in U^{n-k}$ such that $\varphi(\vec{c})$ holds, where $\vec{c}$ is obtained from $\vec{a}$ and $\vec{b}$ by putting their elements in the appropriate position, $\vec{a}$ being in the positions specified by B. For example, if $n = 4$, $B = \{2, 4\}$, and $\vec{a} = (a_1, a_2)$, $\vec{b} = (b_1, b_2)$, then $\vec{c}$ is (b_1, a_1, b_2, a_2). Formally, for $i \in [1, n]$, let k_1 be the number of $j \in B$ with $j \leq i$, and k_2 be the number of $j \notin B$ with $j \leq i$. Then c_i is a_{k_1} if $i \in B$, and b_{k_2}, if $i \notin B$.

We use the notation

$$\vec{a}_1 \equiv^{\varphi}_{B_i} \vec{a}_2 \qquad \Leftrightarrow \qquad \varphi_{B_i}(\vec{a}_1, \mathcal{M}) = \varphi_{B_i}(\vec{a}_2, \mathcal{M}).$$

We now obtain the following characterization of $\mathrm{VP}_{\mathcal{M}}(\varphi, P)$.

Corollary 5. *Let $\mathcal{M}$ be as in Theorem 3, and let $\varphi(x_1, \ldots, x_n)$ and a partition $P = (B_1, \ldots, B_m)$ on $\{1, \ldots, n\}$ be given. Then:*

(1) *For each $i \leq m$, it is decidable if the equivalence relation $\equiv^{\varphi}_{B_i}$ has finitely many equivalence classes. Furthermore, $\varphi \sim_{\mathcal{M}} P$ iff each $\equiv^{\varphi}_{B_i}$ has finitely many classes.*

(2) *If $\varphi \sim_{\mathcal{M}} P$, then one can further effectively find integers $N_1, \ldots, N_m > 0$ and formulae $\alpha^i_j(\vec{x}_{B_i})$, $i = 1, \ldots, m$, $j = 1, \ldots, N_i$, such that $\equiv^{\varphi}_{B_i}$ has N_i equivalence*

classes, which are definable by the formulae $\alpha^i_j(\vec{x}_{B_i})$, $j \leq N_i$. *Furthermore,*

$$\mathcal{M} \models \forall \vec{x} \left(\varphi(\vec{x}) \leftrightarrow \bigvee_{(j_1,\ldots,j_m)\in K} \alpha^1_{j_1}(\vec{x}_{B_1}) \wedge \ldots \wedge \alpha^m_{j_m}(\vec{x}_{B_m}) \right), \quad (6)$$

where

$$K = \{(j_1,\ldots,j_m) \mid \mathcal{M} \models \exists \vec{x} \left(\alpha^1_{j_1}(\vec{x}_{B_1}) \wedge \ldots \wedge \alpha^m_{j_m}(\vec{x}_{B_m}) \wedge \varphi(\vec{x}) \right)\}.$$

Proof. Let P_i be the partition with two blocks: B_i and $C_i = \bigcup_{j\neq i} B_j$. If $\varphi \sim_{\mathcal{M}} P$, then $\varphi \sim_{\mathcal{M}} P_i$ for all i. From the proof of Theorem 3 we know that $\varphi \sim_{\mathcal{M}} P_i$ iff $\equiv^{\varphi}_{B_i}$ has finitely many equivalence classes. Furthermore, there exist formulae $\alpha^i_j(\vec{x}_{B_i}), \xi^i_j(\vec{x}_{C_i})$, $j \leq N_i$, such that φ is equivalent to

$$\bigvee_{j=1}^{N_i} \alpha^i_j(\vec{x}_{B_i}) \wedge \xi^i_j(\vec{x}_{C_i})$$

and

$$\mathcal{M} \models \forall \vec{x} \bigwedge_{j_1\neq j_2} \neg(\alpha^i_{j_1}(\vec{x}_{B_i}) \leftrightarrow \alpha^i_{j_2}(\vec{x}_{B_i})) \quad \wedge \quad \forall \vec{x} \bigvee_{j=1}^{N_i} \alpha^i_j(\vec{x}_{B_i})$$

(because α^i_js define equivalence classes that partition $U^{card(B_i)}$).

We then claim that α^i_js are the formulae for the representation in (6). Indeed, suppose $\varphi(\vec{a})$ holds in $\mathcal{M}$. Let $j_i \leq N_i$ be such that $\alpha^i_{j_i}(\vec{a}_{B_i})$ holds. Then $(j_1,\ldots,j_m) \in K$ and $\bigwedge_i \alpha^i_{j_i}(\vec{a}_{B_i})$ holds.

Conversely, assume that for some $(j_1,\ldots,j_m) \in K$ (that is, $\mathcal{M} \models \bigwedge_i \alpha^i_{j_i}(\vec{b}_{B_i}) \wedge \varphi(\vec{b})$ for some $\vec{b}$) we have $\bigwedge_i \alpha^i_{j_i}(\vec{a}_{B_i})$. We write $(\vec{a},\vec{b})^P_k$ for the tuple composed of $\vec{a}_{B_1},\ldots,\vec{a}_{B_k},\vec{b}_{B_{k+1}},\ldots,\vec{b}_{B_m}$ (all elements appearing in positions specified by the indices in B_is), $k \geq 0$. We now prove by induction on k that $\mathcal{M} \models \varphi((\vec{a},\vec{b})^P_k)$. For $k = 0$ we know that $\varphi(\vec{b})$ holds. If $\varphi((\vec{a},\vec{b})^P_k)$ holds, then the fact that both $\alpha^{k+1}_{j_{k+1}}(\vec{a}_{B_{k+1}})$ and $\alpha^{k+1}_{j_{k+1}}(\vec{b}_{B_{k+1}})$ hold in $\mathcal{M}$ implies that $\varphi_{B_{k+1}}(\vec{a}_{B_{k+1}},\mathcal{M}) = \varphi_{B_{k+1}}(\vec{b}_{B_{k+1}},\mathcal{M})$, and hence $\varphi((\vec{a},\vec{b})^P_{k+1})$ holds. Thus, for $k = m$, we conclude $\mathcal{M} \models \varphi(\vec{a})$.

This finally shows the representation (6), and that finiteness of the number of equivalence classes of all $\equiv^{\varphi}_{B_i}$ implies $\varphi \sim_{\mathcal{M}} P$. □

4. DECIDABILITY FOR SPECIFIC CLASSES OF CONSTRAINTS

The general decidability result can be applied to a variety of structures, most notably, those that we listed earlier as the ones particularly relevant to constraint database applications (especially to spatial and temporal databases). In fact, the problem will be shown to be decidable for linear constraints over the rationals and the reals (this corresponds to structures $\langle \mathbb{Q}, +, -, 0, 1, < \rangle$ and $\mathbf{R}_{\text{lin}}$), polynomial constraints over the reals ($\mathbf{R}$), and linear repeating points [Kabanza et al. 1995] ($\mathcal{Z}_0$).

4.1 Constraints on the integers

Here the result follows easily form Theorem 3.

Proposition 6. *Let $\mathcal{M}$ be $\langle \mathbb{N}, <, \ldots \rangle$ or $\langle \mathbb{Z}, <, \ldots \rangle$, and let its theory be decidable. Assume, in the latter case, that there is at least one definable constant in $\mathcal{M}$. Then the variable partition and independence problems are decidable over $\mathcal{M}$.*

Proof. We check conditions of Theorem 3. One can test algebraicity due to the presence of a discrete order: given $\varphi(x)$, the sentence $\exists n \exists m \forall x\ (\varphi(x) \to n \le x \le m)$ tests if $\varphi(\mathcal{M})$ is finite. Over $\langle \mathbb{N}, <, \ldots \rangle$ one has definable invariant Skolem functions simply by choosing lexicographically least $\vec{y}$ satisfying $\varphi(\vec{x}, \vec{y})$, for each $\vec{x}$. For $\varphi(\vec{x}, y)$ over $\langle \mathbb{Z}, <, \ldots \rangle$, let $\psi(\vec{x}, y)$ hold if y is the least element above c satisfying $\varphi(\vec{x}, y)$, if such an element exists, or y is the greatest element not exceeding c satisfying $\varphi(\vec{x}, y)$, if no element above c satisfies $\varphi(\vec{x}, \cdot)$. Here c is a definable constant. Clearly, this defines an invariant Skolem function, and the construction easily generalizes to tuples of variables $\vec{y}$. □

Corollary 7. *The variable partition problem is decidable over $\mathcal{Z}_0 = \langle \mathbb{Z}, +, -, 0, 1, <, (\equiv_k)_{k>1} \rangle$.* □

4.2 Linear and polynomial constraints over the reals

The linear constraints over the reals (corresponding to $\mathbf{R}_{\text{lin}} = \langle \mathbb{R}, +, -, 0, 1, < \rangle$) and the polynomial constraints over the reals (corresponding to $\mathbf{R} = \langle \mathbb{R}, +, \cdot, 0, 1, < \rangle$) are the most useful constraints for spatial and spatio-temporal applications, where the problem of variable independence originated, and where variable independence is used in system prototypes. We thus concentrate on these constraints.

In many cases, however, we can state the results in greater generality using the concept of o-minimality (cf. section 2). This concept originated in model theory about a decade ago [Pillay and Steinhorn 1988; van den Dries 1998], and found some computer science applications too, most notably in databases [Benedikt and Libkin 2000] and hybrid systems [Lafferriere et al. 2000].

It is known that every o-minimal expansion of the $\mathbf{R}_{\text{lin}}$ has definable invariant Skolem functions [Marker et al. 1996; van den Dries 1998]. Since every definable subset of U is a finite union of points and open intervals, one can test algebraicity, assuming that the order is dense: given $\varphi(x)$, the sentence $\exists u \exists v \forall x\ (u < x < v \to \varphi(x)$ tests if $\varphi(\mathcal{M})$ is infinite. This shows

Corollary 8. *Let $\mathcal{M} = \langle \mathbb{R}, +, 0, 1, <, \ldots \rangle$ be o-minimal, and have a decidable theory. Then the variable partition and independence problems are decidable over $\mathcal{M}$. In particular, these problems are decidable over $\mathbf{R}_{\text{lin}}$ and $\mathbf{R}$.* □

Since $\langle \mathbb{Q}, +, -, 0, 1, < \rangle$ is elementarily equivalent to $\mathbf{R}_{\text{lin}}$, we conclude that the variable partition problem is decidable over it, too.

4.3 Uniform decidability and complexity bounds

Our next goal is to present a *uniform* procedure for solving the problem $\text{VI}_{\mathcal{M}}(\varphi, P)$. More precisely, we say that the variable partition problem is *uniformly decidable* over $\mathcal{M}$ if the theory of $\mathcal{M}$ is decidable, and for every partition P on $\{1, \ldots, n\}$, there exists a single sentence Φ_P in the language of $\mathcal{M}$ expanded with an n-ary relation symbol S such that for any formula $\varphi(x_1, \ldots, x_n)$,

$$\varphi \sim_{\mathcal{M}} P \quad \text{iff} \quad (\mathcal{M}, \varphi(\mathcal{M})) \models \Phi_P.$$

Here $(\mathcal{M}, \varphi(\mathcal{M}))$ is the expansion of $\mathcal{M}$ where the new symbol S is interpreted as $\{\vec{a} \mid \mathcal{M} \models \varphi(\vec{a})\}$. Note that the decidability of the theory of $\mathcal{M}$ implies that $(\mathcal{M}, \varphi(\mathcal{M})) \models \Phi_P$ is decidable.

We also say that the variable independence problem is uniformly decidable, if for $n > 1$ and any $i, j \leq n$, there exists a sentence $\Phi^n_{i,j}$ in in the language of $\mathcal{M}$ expanded with an n-ary relation symbol S such that for any formula $\varphi(x_1, \ldots, x_n)$, x_i and x_j are independent iff $(\mathcal{M}, \varphi(\mathcal{M})) \models \Phi^n_{i,j}$. As in Lemma 1, it is easy to show that the uniform decidability of the variable partition problem is equivalent to the uniform decidability of the variable independence problem.

PROPOSITION 9. *Let $\mathcal{M} = \langle \mathbb{R}, +, 0, 1, <, \ldots \rangle$ be o-minimal and have a decidable theory. Then the variable independence and partition problems are uniformly decidable over $\mathcal{M}$.*

Proof. It suffices to show, in view of Corollary 5 that for every positive integers n, m there exists a sentence $\Phi_{n,m}$ in the language of $\mathcal{M}$ expanded with one $(n+m)$-ary symbol S, such that for every formula $\varphi(x_1, \ldots, x_n, y_1, \ldots, y_m)$ over $\mathcal{M}$, $(\mathcal{M}, \varphi(\mathcal{M})) \models \Phi_{n,m}$ iff the equivalence relation $\equiv$ on U^n given by

$$\vec{a}_1 \equiv \vec{a}_2 \quad \Leftrightarrow \quad \varphi(\vec{a}_1, \mathcal{M}) = \varphi(\vec{a}_2, \mathcal{M})$$

has finitely many equivalence classes.

Note that the expansion of $\mathcal{M}$ by an extra predicate symbol to be interpreted as a definable predicate is o-minimal as well. Now, let $\chi(x, \vec{y})$ be a formula in the expanded structure. We then let $endp_\chi(x, \vec{y})$ be a formula such that $endp_\chi(a, \vec{b})$ holds iff a is an endpoint of one of the intervals that form the set $\{c \mid \chi(c, \vec{b})\}$. This is clearly definable just with order.

Next, for any $\chi(x, \vec{y})$, define $rep^x_\chi(x, \vec{y})$ by

$$\begin{array}{l}
\forall z \chi(z, \vec{y}) \wedge x = 0 \\[1em]
\vee\ \exists! z\, endp_\chi(z, \vec{y}) \ \wedge\ \left(\begin{array}{l} \quad\ \chi(z, \vec{y}) \wedge x = z \\ \vee\ \neg\chi(z, \vec{y}) \wedge (\forall v < z\ \chi(v, \vec{y})) \wedge (x + 1 = z) \\ \vee\ (\forall v > z\ \chi(v, \vec{y})) \wedge (\forall v \leq z\ \neg\chi(v, \vec{y})) \wedge (x = z + 1) \end{array} \right) \\[2em]
\vee\ \exists z_1, z_2\ endp_\chi(z_1, \vec{y}) \wedge endp_\chi(z_2, \vec{y}) \wedge (z_1 < z_2) \wedge \\
\qquad (\forall v\ endp_\chi(v, \vec{y}) \rightarrow (v < z_2 \rightarrow v = z_1)) \wedge \\
\qquad\quad \left(\begin{array}{l} \quad\ \chi(z_1, \vec{y}) \wedge x = z_1 \\ \vee\ \neg\chi(z_1, \vec{y}) \wedge (\forall v < z_1\ \chi(v, \vec{y})) \wedge (x + 1 = z_1) \\ \vee\ (\forall v \leq z_1\ \neg\chi(v, \vec{y})) \wedge (2x = z_1 + z_2) \end{array} \right)
\end{array}$$

This formula says that either every real number satisfies $\chi(\cdot, \vec{y})$ and $x = 0$, or there is a single endpoint z of $\chi(\mathcal{M}, \vec{y})$, and then x is either z, or $z - 1$, or $z + 1$, depending on which intervals are included in $\chi(\mathcal{M}, \vec{y})$, or there are two or more endpoints of $\chi(\mathcal{M}, \vec{y})$, and, for $z_1 < z_2$ being the two smallest one, x equals z_1 if $\chi(z_1, \vec{y})$ holds, or $x = z_1 - 1$ if $(-\infty, z_1)$ is in $\chi(\mathcal{M}, \vec{y})$, or otherwise $x = (z_1 + z_2)/2$. It is easy to see then that this formula has the property that for o-minimal $\mathcal{M}$, if $\chi(\mathcal{M}, \vec{b}) = \emptyset$, then for any a, $rep^x_\chi(a, \vec{b})$ does not hold, and if $\chi(\mathcal{M}, \vec{b}) \neq \emptyset$, then $rep^x_\chi(a, \vec{b})$ holds for a single element $a \in \chi(\mathcal{M}, \vec{b})$.

We now prove uniform decidability. Let $\psi(x_1,\ldots,x_n,z_1,\ldots,z_n)$ define the equivalence relation $\equiv$; that is, $\psi = \forall\vec{y}\ (\varphi(\vec{x},\vec{y}) \leftrightarrow \varphi(\vec{z},\vec{y}))$. Define ψ_0 to be ψ, and let

$$\psi_i \quad = \quad rep^{x_i}_{\psi_{i-1}}(\vec{x},\vec{z}), \qquad 1 \le i \le n.$$

Since for each $\vec{a}_1 \equiv \vec{a}_2$ we have $\psi(\mathcal{M},\vec{a}_1) = \psi(\mathcal{M},\vec{a}_2) =$ the equivalence class of $\vec{a}_1$, we obtain from the construction of *rep* that $\psi_n(\vec{b},\vec{a}_1) \leftrightarrow \psi_n(\vec{b},\vec{a}_2)$, and in fact there is a single $\vec{b}$ for which $\psi_n(\vec{b},\vec{a}_1)$ holds, and for this $\vec{b}$ we have $\psi(\vec{b},\vec{a}_1)$, that is, $\vec{b} \equiv \vec{a}_1 \equiv \vec{a}_2$. Therefore, the formula

$$\gamma'(\vec{x}) \quad = \quad \exists\vec{z}\ \psi_n(\vec{x},\vec{z})$$

(in the language of $\mathcal{M}$ expanded with S) defines a set of representatives of the equivalence classes of $\equiv$. Therefore, the formula

$$\gamma(x) \quad = \quad \bigvee_{i=1}^{n} \exists y_1,\ldots,y_{n-1}\ \gamma'((x,\vec{y})_i),$$

where $(x,\vec{y})_i$ is the tuple in which x is inserted in the ith position, defines the set of all coordinates of the representatives of $\equiv$ chosen by ψ_n. Thus, $\equiv$ has finitely many classes iff γ is algebraic. Therefore, the sentence

$$\neg\exists x_1 \exists x_2 \Big((x_1 < x_2) \wedge (\forall x\ (x_1 < x < x_2) \rightarrow \gamma(x))\Big)$$

in the expanded language tests if the equivalence relation $\equiv$ has finitely many classes. This proves the proposition. $\Box$

Proposition 9 implies that the variable independence problem is uniformly decidable over $\mathbf{R}_{\text{lin}}$ and $\mathbf{R}$. The main application of this result is in establishing complexity bounds.

Since $\mathbf{R}$ admits quantifier elimination, every semi-algebraic set is given by a Boolean combination of polynomial inequalities. Thus, a standard way to represent a semi-algebraic set in $\mathbb{R}^n$ [Basu 1999; Caviness and Johnson 1998; Renegar 1992] is by specifying a collection of polynomials $p_1,\ldots,p_k \in \mathbb{Z}[x_1,\ldots,x_n]$, and defining a set X as a Boolean combination of sets of the form $\{\vec{a} \mid p_i(\vec{a})\ \theta\ 0\}$, where θ is either $=$ or $>$. Here $\mathbb{Z}[x_1,\ldots,x_n]$, as usual, is the set of all polynomials in n variables with coefficients from $\mathbb{Z}$. One can use coefficients from $\mathbb{Q}$ as well, but this would not affect the class of definable sets.

Thus, when we study complexity of $\mathrm{VP}_{\mathbf{R}}(\varphi,P)$, we assume that φ is given as a Boolean combination of polynomial equalities and inequalities, with all polynomials having integer coefficients. The size of the input formula is then defined in a standard way, assuming that all integer coefficients are given in binary. All the above applies to semi-linear sets (that is, sets definable over $\mathbf{R}_{\text{lin}}$); we just restrict our attention to polynomials of degree 1.

Corollary 10. *Let $\mathcal{M}$ be $\mathbf{R}_{\text{lin}}$ or $\mathbf{R}$. Let P be a fixed partition on $\{1,\ldots,n\}$. Then, for a semi-algebraic (semi-linear) set given by a Boolean combination $\varphi(\vec{x})$ of polynomial inequalities (of degree 1), the problem $\mathrm{VI}_{\mathcal{M}}(\varphi,P)$ is solvable in time polynomial in the size of φ.*

Proof. Let Φ_P be the sentence for uniform decidability of the variable partition problem. Assume that Φ_P is in the prenex form. Using the standard bounds for quantifier elimination over $\mathbf{R}$ [Basu 1999; Renegar 1992], one obtains that there exist constants c_1 and c_2 that depend on Φ_P only, such that for any $\varphi(\vec{x})$, the complexity of deciding Φ'_P, obtained from Φ_P by using φ in place of the extra predicate S, is bounded by $(c_1 N)^{c_2}$, where N is the size of φ. Indeed, if S occurs s times in the quantifier-free part of Φ_P, then the size of the quantifier-free part of Φ'_P is $c_0 + s \cdot N$, for some constant c_0 depending only on Φ_P. From this, and bounds of [Basu 1999; Renegar 1992], the complexity bound follows. As c_1 and c_2 depend only on Φ_P, and hence only on P (by Proposition 9), the result follows. For semi-linear sets, the proof repeats the one above verbatim, as one can guarantee the same bounds for quantifier elimination. $\square$

Another reason to consider the uniform decision procedure for variable independence is that it gives us a test for variable independence under some transformations. For example, linear coordinate change in general would destroy variable independence, although it has relatively little effect on shapes on objects in $\mathbb{R}^n$. Consider, for example, the following version of the variable independence problem $\mathrm{LVI}(X, x_i, x_j)$: Given a semi-algebraic set $X \subseteq \mathbb{R}^n$ (defined by a formula over $\mathbf{R}$), is there a linear change of coordinates such that in the new coordinate system, variables x_i and x_j are independent?

The general decision procedure of Theorem 3 does not give us a decision procedure for LVI. However, using uniformity, we easily obtain:

Corollary 11. *The problem* $\mathrm{LVI}(X, x_i, x_j)$ *is decidable.*

Proof. Let X be defined by a formula φ. For each partition P, x_i and x_j being in two different blocks, consider the sentence

$$\Psi_P \quad = \quad \exists a_{11} \ldots \exists a_{nn} \ \chi(\mathbf{A}) \wedge \Phi_P(\mathbf{A}S)$$

where $\mathbf{A}$ is the matrix given by $a_{11}, \ldots, a_{nn}$, $\chi(\mathbf{A})$ is a sentence over $\mathbf{R}$ stating that $\det(\mathbf{A}) \neq 0$, and $\Phi_P(\mathbf{A}S)$ is obtained by replacing each occurrence of $S(\vec{x})$ by $\exists \vec{y}\ (\varphi(\vec{y}) \wedge \vec{x} = \mathbf{A}\vec{y})$. The answer to $\mathrm{LVI}(X, x_i, x_j)$ is yes iff $\mathbf{R} \models \Psi_P$ for one such P. The corollary follows from the decidability of $\mathbf{R}$. $\square$

It turns out that not only the decision part of $\mathrm{VI}_{\mathcal{M}}(\varphi, P)$ and $\mathrm{VP}_{\mathcal{M}}(\varphi, P)$ can be solved in polynomial time for a fixed P over $\mathbf{R}_{\mathrm{lin}}$ and $\mathbf{R}$, but there is also a polynomial time algorithm for finding a formula equivalent to φ that witnesses $\varphi \sim_{\mathcal{M}} P$.

Theorem 12. *a) Given $n > 1$, and a partition $P = (B_1, \ldots, B_m)$ on $\{1, \ldots, n\}$, there exists an algorithm that, for every semi-algebraic set given by a formula $\varphi(x_1, \ldots, x_n)$ which is a Boolean combination of polynomial equalities and inequalities, tests if $\varphi \sim_{\mathcal{M}} P$, and in the case of the positive answer, computes quantifier-free formulae $\alpha^i_j(\vec{x}_{B_i})$ such that each $\alpha^i_j(\vec{x}_{B_i})$ is a Boolean combination of polynomial (in)equalities (where polynomials depend only on $\vec{x}_{B_i}$ and all coefficients are integers), and $\varphi(\vec{x})$ is equivalent to $\bigvee_j \bigwedge_i \alpha^i_j(\vec{x}_{B_i})$. Moreover the algorithm works in time polynomial in the size of φ.*

b) The same statement is true when on replaces semi-algebraic by semi-linear, and all polynomials are of degree 1.

Proof. We start with a). We saw (Corollary 10) that $\varphi \sim_{\mathcal{M}} P$ can be decided in polynomial time. Assume thus that $\varphi \sim_{\mathcal{M}} P$.

We say that a collection of formulae $\alpha^i_j(\vec{x}_{B_i})$, $i \leq m, j \leq M_i$, *witnesses* $\varphi \sim_{\mathcal{M}} P$ if each $\alpha^i_j(\vec{x}_{B_i})$ defines an equivalence class of the relation $\equiv^{\varphi}_{B_i}$, and each equivalence class of $\equiv^{\varphi}_{B_i}$ is definable by some $\alpha^i_j(\vec{x}_{B_i})$ (this means, in particular, that some $\alpha^i_j(\vec{x}_{B_i})$ and $\alpha^i_l(\vec{x}_{B_i})$ could be equivalent).

The first step of the proof is to show that it suffices to construct, in polynomial time, a family of formulae witnessing $\varphi \sim_{\mathcal{M}} P$.

To prove this, we first recall known bounds on quantifier elimination over $\mathbf{R}$. Suppose $\chi(\vec{y})$ is a formula in the prenex form, whose quantifier-free part is of size N, and the degrees of all polynomials used in it do not exceed d. Then there exist two constants c_1 and c_2 that depend only on the quantifier block of χ and the length of $\vec{y}$ such that $N^{c_1} \cdot d^{c_2}$ is a bound on both the number of operations needed to compute a quantifier-free $\chi_0(\vec{y})$ equivalent to $\chi(\vec{x})$, and the size of χ_0. Moreover, the degrees of polynomials appearing in χ_0 are bounded by d^{c_2}, and all coefficients of polynomials used in χ_0 belong to the minimal subring of $\mathbb{R}$ that contains all coefficients of polynomials used in χ (in our case, $\mathbb{Z}$, since in the input formula we have polynomials with integer coefficients). This follows from the results of [Basu 1999; Renegar 1992]; in fact, those papers provide more detailed complexity analysis, but the above will suffice for our purposes.

Now suppose that we have constructed, in polynomial time, a family $\{\alpha^i_j(\vec{x}_{B_i})\}$, witnessing $\varphi \sim_{\mathcal{M}} P$. In view of Corollary 5, we have

$$\varphi(\vec{x}) \quad \leftrightarrow \quad \bigvee_{(j_1,\ldots,j_m)\in K} \alpha^1_{j_1}(\vec{x}_{B_1}) \wedge \ldots \wedge \alpha^m_{j_m}(\vec{x}_{B_m}), \tag{7}$$

where $K = \{(j_1,\ldots,j_m) \mid \mathbf{R} \models \exists\vec{x}\ (\alpha^1_{j_1}(\vec{x}_{B_1}) \wedge \ldots \wedge \alpha^m_{j_m}(\vec{x}_{B_m}) \wedge \varphi(\vec{x}))\}$. (Indeed, the only difference with Corollary 5 is that for a fixed i, some $\alpha^i_j(\vec{x}_{B_i})$, $\alpha^i_{j'}(\vec{x}_{B_i})$ may be equivalent, but this only adds a number of equivalent disjuncts to the right hand side of (7), which does not affect the truth value.)

Thus, we must show how to find K in polynomial time. Since P (and thus m) is fixed, we enumerate all tuples $(j_1,\ldots,j_m) \leq (M_1,\ldots,M_m)$ in polynomial time. For each $\vec{\jmath} = (j_1,\ldots,j_m)$, consider the sentence $\Psi_{\vec{\jmath}} = \exists\vec{x}\ (\alpha^1_{j_1}(\vec{x}_{B_1}) \wedge \ldots \wedge \alpha^m_{j_m}(\vec{x}_{B_m}) \wedge \varphi(\vec{x}))$. The size of each α^i_j is polynomial in the size of φ, by the assumption. That is, the size of each α^i_j does not exceed $c \cdot N^p$, for some constants c, p, where N is the size of φ. Thus, the size of the quantifier-free part of $\Psi_{\vec{\jmath}}$, N_1, is at most $O(N^p)$, and the same is a bound on the degrees of polynomials used. By the bound on quantifier-elimination, $\mathcal{M} \models \Psi_{\vec{\jmath}}$ can be decided in time $O(N_1^{c'})$, where c' depends only on the fixed quantifier-prefix $\exists\vec{x}$ (recall that the length of $\vec{x}$ is fixed, since the partition P is fixed). We thus obtain that there is a polynomial p such that for each $\vec{\jmath}$, the decision procedure $\mathcal{M} \models \Psi_{\vec{\jmath}}$ takes time $p(N)$. This, and the bound on M_is, imply that K can be found in polynomial time, if P is fixed. Hence, if α^i_js are found in polynomial time, so is the representation (7).

Thus, to prove the theorem, it suffices to show how to construct a family $\alpha^i_j(\vec{x}_{B_i})$ witnessing $\varphi \sim_{\mathcal{M}} P$ in polynomial time. Without loss of generality, assume that we are given $\varphi(\vec{y},\vec{z})$, with $\vec{y}$ of length n and $\vec{z}$ of length m. Let $\vec{a}_1 \equiv \vec{a}_2$ iff $\varphi(\vec{a}_1,\mathbf{R}) = \varphi(\vec{a}_2,\mathbf{R})$; this is an equivalence relation on $\mathbb{R}^n$. Assume that we know already that $\equiv$ has finitely many classes. To complete the proof, it thus suffices to show how to produce formulae $\alpha_j(\vec{y})$ defining equivalence classes, in time polynomial in the size of φ.

From the proof of Proposition 9, we know that there exists a formula $\gamma(x)$ in the language of the real field plus an $n+m$-ary relation S, such that γ depends on n and m only, and $(\mathbf{R}, \varphi(\mathbf{R})) \models \gamma(a)$ iff for some canonically chosen set of representatives of $\equiv$, a is in one of $\vec{a}$ in this set. We now substitute the definition of φ for S in γ, and perform quantifier-elimination. Let $\delta(x)$ be the resulting formula. From the bounds on quantifier-elimination, we obtain that $\delta(x)$ can be found in time polynomial in N, where N is the size of φ. This is because γ is determined by n and m, which are fixed (as n corresponds to the size of a block in P, m to the number of the remaining variables). By putting φ into the definition of γ, we obtain a formula whose quantifier-free part is linear in N, and then the bounds on quantifier-elimination imply that the parameters in the exponent depend only on n and m, that is, on P. Thus, $\delta(x)$ is obtained in polynomial time.

Since $\delta(x)$ is quantifier-free, it is a Boolean combination of polynomial equalities and inequalities involving polynomials from a set $\mathcal{P} = \{p_1(x), \ldots, p_l(x)\}$. Furthermore, δ is algebraic. We now claim that every a such that $\mathbf{R} \models \delta(a)$ is a root of one of p_is. Assume this is not the case: $\mathbf{R} \models \delta(a)$ and a is not a root. Then there is a small neighborhood of a in which signs of all p_is are the same as the signs of $p_i(a)$s. Thus, since $\delta(x)$ is quantifier-free, we obtain $\mathbf{R} \models \delta(b)$ for each b from this neighborhood of a, which implies that $\delta(\mathbf{R})$ is infinite.

Now suppose p_i is of degree d_i, and suppose we have formulae $\rho_{ik}(x)$ saying that x is the kth real root of p_i (or 0, if there is no such root), $k \leq d_i$. Then we would define formulae

$$\tau_{(i_1,k_1),\ldots,(i_n,k_n)}(y_1, \ldots, y_n) \quad = \quad \bigwedge_{j=1}^{n} \rho_{i_j k_j}(y_j), \qquad k_j \leq d_j$$

producing n-tuples of real roots of polynomials in $\mathcal{P}$ (some entries in those tuples can be 0 as well). We know for every equivalence class of $\equiv$, there is a tuple in it that satisfies one of these formulae. Moreover, the number of formulae $\tau_{(i_1,k_1),\ldots,(i_n,k_n)}$ is at most $(l \cdot D)^n$, where D is the maximum degree of a polynomial in $\mathcal{P}$. From each formula $\tau_{(i_1,k_1),\ldots,(i_n,k_n)}$, we define the equivalence class as

$$\alpha_{(i_1,k_1),\ldots,(i_n,k_n)}(\vec{y}) \quad = \quad \exists \vec{y}_1 \forall \vec{z}\; (\tau_{(i_1,k_1),\ldots,(i_n,k_n)}(\vec{y}_1) \wedge (\varphi(\vec{y}, \vec{z}) \leftrightarrow \varphi(\vec{y}_1, \vec{z}))).$$

Since the quantifier prefix $\exists \vec{y}_1 \forall \vec{z}$ is fixed (as n and m are fixed), we conclude that a quantifier-free formula equivalent to $\alpha_{(i_1,k_1),\ldots,(i_n,k_n)}$ can be found in time $O((N + nN')^s)$, where s is determined by P, and N' is an upper bound on the size of $\rho_{ik}(x)$. This, and the estimate on the number of formulae $\tau_{(i_1,k_1),\ldots,(i_n,k_n)}$, show that the required collection of formulae defining equivalence classes can be produced in time polynomial in the size of φ, provided two conditions hold:

(1) The set $\mathcal{P}$ can be found in polynomial time (in N, with P fixed);

(2) Each formula $\rho_{ik}(x)$ can be constructed in time polynomial in N.

The first item follows from the fact that $\delta(x)$ is found in time polynomial in N (see above). To show the second item, consider each polynomial $p_i(x) \in \mathcal{P}$. Using an algorithm for root isolation (see, for example, [Caviness and Johnson 1998; Collins and Loos 1983]), we find a sequence $a_1 < a_2 < \ldots < a_r$, where r is at most the degree of p_i plus one, such that each interval (a_i, a_{i+1}), $1 \leq i < r$, contains exactly one root of p_i. This sequence can be found in time polynomial in the size of the representation of p_i [Caviness and Johnson 1998; Collins and Loos 1983]. We now define

$$\rho_{ik}(x) \;=\; \begin{cases} (p_i(x) = 0) \wedge (x > a_k) \wedge (x < a_{k+1}) & k < r \\ x = 0 & k \geq r \end{cases}$$

Clearly, these formulae define the roots (and perhaps 0), and due to the bounds on root isolation, they can be found in polynomial time. This completes the proof of item 2, and thus the theorem for the case of $\mathbf{R}$.

For linear constraints ($\mathbf{R}_{\text{lin}}$), the proof follows along the same lines, but is a bit easier. Since there similar bounds on quantifier elimination over $\mathbf{R}_{\text{lin}}$, where the exponent depends only on the number of free variables in the quantifier-prefix [Caviness and Johnson 1998], we can use the same proof as above, except that we do not have to deal with the real root isolation, as a linear function in one variable has at most one root, which is definable. □

4.4 Other constraints

In this section, we consider two examples of nicely-behaving structures over which the variable independence problem is decidable, despite the fact that they do not satisfy all the conditions of Theorem 3. Admittedly, this is more of purely theoretical interest, although algebraically closed fields were studied in connection with some expressivity problems inspired by constraint databases [Chapuis and Koiran 1999].

The first structure we consider is the field of complex numbers $\mathcal{C} = \langle \mathbb{C}, +, \cdot \rangle$. It has a decidable theory and admits quantifier-elimination; furthermore, it is strongly minimal (every definable subset of $\mathbb{C}$ is either finite or cofinite) [Hodges 1993]. The latter property implies that it does not have definable Skolem functions (invariant or not). Indeed, if the equivalence relation $x^2 = y^2$ had a definable set of representatives, this set and its complement would be infinite. Hence, Theorem 3 does not apply to $\mathcal{C}$. Still, we can prove a weaker result. We say that variables x_i and x_j are weakly independent in $\varphi(\vec{x})$ if there exists a finite set $C \subset \mathbb{C}$ and a finite collection of formulae $\alpha_k(\vec{x})$ in the language of $\mathcal{C}$ extended with constants for C, such that no α_k mentions both x_i and x_j, and φ is equivalent to a Boolean combination of α_ks.

PROPOSITION 13. *Let $\varphi(\vec{x})$ be a formula over $\mathcal{C}$, and x_i, x_j two distinct variables. Then it is decidable if x_i and x_j are weakly independent in φ.*

Proof. The proof follows closely the proof of Theorem 3. Instead of Skolemization to find representatives of equivalence classes, we use a weaker property of *elimination of imaginaries* [Hodges 1993; Marker et al. 1996], which says that for every definable equivalence relation E on $\mathbb{C}^k$, there exists a function $f : \mathbb{C}^k \to \mathbb{C}^m$ such that $E(a, b)$ implies $f(a) = f(b)$. Furthermore, such a function can be found effectively (because the theory is decidable), and it can also be tested effectively if a formula $\psi(x)$ is algebraic: this follows from quantifier elimination and strong minimality. Thus, we prove an analog of lemma 4 (since we do not have definable Skolem functions, we must use finitely many new constant symbols to identify equivalence classes), and then use the function that eliminates imaginaries and test for algebraicity to check if a given equivalence relation is finite. □

Another example is that of the random graph, that is, a countable structure that satisfies every sentence that is true in almost all finite graphs (this theory is ω-categorical; hence we speak of *the* random graph). Again, its theory is decidable, and has quantifier elimination, but does not have definable Skolem functions. Still, a simple argument shows:

PROPOSITION 14. *The variable independence problem is decidable over the random graph.*

Proof. There are only finitely many non-equivalent formulae in n variables, and they can all be effectively listed. We look at all of them in which two given variables do not occur in the same atomic formula, and check if any of them is equivalent to a given formula φ. □

4.5 Spatio-temporal applications

Even though we do have polynomial-time algorithms for testing variable independence for linear and polynomial constraints, the exponent becomes quite large as the dimension increases. This kind of situation is not new at all; it is the case, for example, for some quantifier-elimination algorithms that, although polynomial for a fixed dimension, become unmanageable for high dimensions.

Fortunately, in practical applications of variable independence encountered so far, the dimension is not very high, as typically this concept is important for spatio-temporal applications [Grumbach et al. 1998; 1999]. In this case, one deals with formulae $\varphi(x, y, t)$ over $\mathbf{R}_{\text{lin}}$ or $\mathbf{R}$; that is, linear or polynomial constraints in three variables x, y and t, where x, y describe the spatial component and t describes the temporal component; one is then interested in showing the independence of (x, y) and t. A typical application is cadastral information, that is, information about land ownership over time. As ownership does not change continuously with time, often variables x, y are independent of t.

We now show that there is a simpler way of testing variable independence in this setting than in the general setting of Theorem 12.

Recall that a *cylindrical algebraic decomposition (CAD)* of $\mathbb{R}^n$ [Bochnak et al. 1998; Caviness and Johnson 1998] is a partition of $\mathbb{R}^n$ into finitely many sets, called cells, such that each cell is homeomorphic to $\mathbb{R}^i$, $i \leq n$. A CAD of $\mathbb{R}^1$ is a decomposition into points and open intervals. A CAD of $\mathbb{R}^n$ is defined as follows. Let $C_1, \ldots, C_p$ be the cells of a CAD of $\mathbb{R}^{n-1}$. Suppose that for each C_i we have a collection of continuous functions $f_1^i, \ldots, f_{m_i}^i : C_i \to \mathbb{R}$, $m_i \geq 0$, such that for each $\vec{x} \in C_i$, $f_1^i(x) < \ldots < f_{m_i}^i(\vec{x})$. Then the cells of a CAD in $\mathbb{R}^n$ are $\{(\vec{x}, c) \mid \vec{x} \in C_i, c < f_1^i(\vec{x})\}$, $\{(\vec{x}, c) \mid \vec{x} \in C_i, c > f_{m_i}^i(\vec{x})\}$, $\{(\vec{x}, c) \mid \vec{x} \in C_i, c = f_j^i(\vec{x})\}$, $j = 1, \ldots, m_i$, and $\{(\vec{x}, c) \mid \vec{x} \in C_i, f_j^i(\vec{x}) < c < f_{j+1}^i(\vec{x})\}$, $j < m_i$, $i = 1, \ldots, p$.

A classical result on cell decomposition says that given a set $p_1, \ldots, p_k$ of polynomials in $\mathbb{Z}[x_1, \ldots, x_n]$, one can effectively construct a CAD of $\mathbb{R}^n$ such that all the functions f_j^i (for all steps of the inductive construction) are definable over $\mathbb{R}$, and polynomials p_ls do not change their sign on any cell. In particular, if we have a formula $\varphi(\vec{x})$ which is a Boolean combination of polynomial inequalities involving p_ls, then $\varphi(\mathbf{R})$ is a union of some cells of this CAD.

Now consider a formula $\varphi(x, y, t)$ which is a Boolean combination of equalities and inequalities involving polynomials $p_1, \ldots, p_k \in \mathbb{Z}[x, y, t]$, and let $P = \{\{x, y\}, \{t\}\}$.

First notice that if $\varphi \sim P$, then φ is equivalent to a formula of the form $\bigvee_i \alpha_i(x,y) \wedge \beta_i(t)$, where each $\beta_i(t)$ is either $t = c_i$, or $c_i < t$, or $t < d_i$, or $c_i < t < d_i$, where c_i and d_i are constants; this follows from o-minimality of the real field.

Suppose then that we do a CAD using the polynomials p_ls, and let $C_1, \ldots, C_p$ be the cells in the xy-plane, and f^i_j the functions on C_is which define the cells of a three-dimensional CAD. Each such function f^i_j on C_i is called *φ-significant* if for the cell $\{(x,y,t) \mid (x,y) \in C_i, t = f^i_j(x,y)\}$ and two cells in $C_i \times \mathbb{R}$ adjacent to it, it is not the case that the three simultaneously belong to $\varphi(\mathbf{R})$ or $\mathbb{R}^3 - \varphi(\mathbf{R})$. (Since φ is a Boolean combination of constraints involving p_ls, several adjacent cells may belong to $\varphi(\mathbf{R})$ or its complement.)

Then, if f^i_j is φ-significant, C_i is not a single point, and f^i_j takes at least two different (and hence infinitely many) distinct values, we can easily see that there is no representation for φ in which t occurs only in the subformulae defining intervals with constant endpoints. This implies:

Proposition 15. *Given a formula $\varphi(x,y,t)$ which is a Boolean combination of polynomial (in)equalities involving $p_1, \ldots, p_k \in \mathbb{Z}[x,y,t]$, the variables x,y are independent from t iff in any CAD for the polynomials p_ls, every φ-significant function f on a non-singleton cell in the xy-plane, is a constant.* $\square$

Since there exist specialized algorithms for constructing CAD in the three-dimensional space that have good enough complexity bounds to be applicable in practice [Arnon et al. 1988], this gives us a good method for testing variable independence in spatio-temporal applications.

5. CONCLUSION

We looked at the problem of deciding, for a set represented by a collection of constraints, whether some variables in those constraints are independent of each other. Knowing this can considerably improve the running time of several constraint processing algorithms, in particular, quantifier elimination. The problem originated in the field of spatio-temporal databases represented by constraints (linear or polynomial over the reals, for example); it was demonstrated that on large datasets, reasonable performance can only be achieved if variables comprise small independent groups. It had not been known, however, if such independence conditions are decidable.

Here we showed that these conditions are decidable for a large class of constraints, including those relevant to spatial and temporal applications. Moreover, for linear and polynomial constraints over the reals, we gave a uniform decision procedure that implies tractability, and we showed that a given constraint set can be converted into one in a nice shape in polynomial time, too. We also considered specialized algorithms suitable for spatio-temporal applications.

ACKNOWLEDGMENTS

I thank Stavros Cosmadakis and Gabi Kuper for bringing the problem to my attention, and Michael Benedikt, Luc Segoufin, and anonymous referees for their comments.

REFERENCES

ARNON, D., COLLINS, G., AND MCCALLUM, S. 1988. Cylindrical algebraic decomposition: an adjacency algorithm for three-dimensional space. *J. Symbolic Computation 5*, 163–187.

BASU, S. 1999. New results on quantifier elimination over real closed fields and applications to constraint databases. *Journal of the ACM 46*, 537–555.

BENEDIKT, M. AND LIBKIN, L. 2000. Relational queries over interpreted structures. *Journal of the ACM 47*, 644–680.

BOCHNAK, J., COSTE, M, AND ROY, M.-F. 1998. *Real Algebraic Geometry.* Springer Verlag, Berlin.

CAVINESS, B. AND JOHNSON, J. EDS. 1998. *Quantifier Elimination and Cylindrical Algebraic Decomposition.* Springer Verlag, Berlin.

CHANG, C., AND KEISLER, H.J. 1990. *Model Theory.* North Holland, Amsterdam.

CHAPUIS, O. AND KOIRAN, P. 1999. Definability of geometric properties in algebraically closed fields. *Mathematical Logic Quarterly 45*, 533–550.

CHOMICKI, J., GOLDIN, D., AND KUPER, G. 1996. Variable independence and aggregation closure. In *Symp. on Principles of Database Systems*, ACM Press, New York, pages 40–48.

COLLINS, G.E., AND LOOS, R. 1983. Real zeros of polynomials. In *Computer Algebra: Symbolic and Algebraic Computation*, Springer-Verlag, pages 83–94.

COSMADAKIS, S., KUPER, G. AND LIBKIN, L. 2001. On the orthographic dimension of definable sets. *Information Processing Letters 79*, 141–145.

ENDERTON, H.B. 1972. *A Mathematical Introduction to Logic.* Academic Press, New York.

FORDAN, A. AND YAP, R. 1998. Early projection in CLP(R). In *Constraint Programming*, Springer, Berlin, pages 177–191.

GRUMBACH, S., RIGAUX, P., AND SEGOUFIN, L. 1998. The DEDALE system for complex spatial queries. In *ACM SIGMOD*, ACM Press, New York, pages 213–224.

GRUMBACH, S., RIGAUX, P., AND SEGOUFIN, L. 1999. On the orthographic dimension of constraint databases. In *Int. Conf. on Database Theory*, Springer Verlag, Berlin, pages 199–216.

HODGES, W. 1993. *Model Theory.* Cambridge Univ. Press, Cambridge.

IMBERT, J.-L. 1994. Redundancy, variable elimination and linear disequations. In *Symo. on Logic Programming*, pages 139–153.

KABANZA, F.,STEVENNE, J.-M., AND WOLPER, P. 1995. Handling infinite temporal data. *Journal of Computer and System Sciences 51*, 3–17.

KANELLAKIS, P, KUPER, G. AND REVESZ, P. 1995. Constraint query languages. *Journal of Computer and System Sciences 51*, 26–52.

KUPER, G, LIBKIN, L. AND PAREDAENS, J., EDS. 2000. *Constraint Databases.* Springer Verlag, Berlin.

LAFFERRIERE, G., PAPPAS, G., AND SASTRY, S. 2000. O-minimal hybrid systems. *Mathematics of Control, Signals, and Systems 13*, 1–21.

L. Libkin. Some remarks on variable independence, closure, and orthographic dimension in constraint databases. *SIGMOD Record 28*, 4, 24–28.

MARKER, D, MESSMER, M. AND PILLAY, A. 1996. *Model Theory of Fields.* Springer Verlag, Berlin.

PILLAY, A. AND STEINHORN, C. 1988. Definable sets in ordered structures. III. *Transactions of the AMS 309*, 469–476.

RENEGAR, J. 1992. On the computational complexity and geometry of the first-order theory of the reals. *J. Symb. Comp. 13*, 255–352.

STOLBOUSHKIN, A. AND TSAITLIN, M. 1999. Finite queries do not have effective syntax. *Information and Computation 153*, 99–116.

VAN DEN DRIES, L. 1984. Remarks on Tarski's problem concerning (R,+,*,exp). In *Logic Colloquium'82*, North Holland, Amsterdam, pages 97–121.

VAN DEN DRIES, L. 1998. *Tame Topology and O-Minimal Structures.* Cambridge Univ. Press, Cambridge.

WILKIE, A. 1996. Model completeness results for expansions of the ordered field of real numbers by restricted Pfaffian functions and the exponential function. *J. Amer. Math. Soc. 9*, 1051–1094.

Ziegler, G. M. 1994. *Lectures on Polytopes.* Springer-Verlag, Berlin.

Received June 2001; revised April 2002; accepted April 2002