Journal of Pure and Applied Algebra 67 (1990) 1–14
North-Holland

PRECISE SEQUENTIAL AND PARALLEL COMPLEXITY BOUNDS FOR QUANTIFIER ELIMINATION OVER ALGEBRAICALLY CLOSED FIELDS*

Noaï FITCHAS
Instituto Argentino de Matemática, Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET), Viamonte 1636, (1055) Buenos Aires, Argentina

André GALLIGO and Jacques MORGENSTERN
Département de Mathématique, Université de Nice, 06034 Nice Cedex, France and INRIA, Sophia Antipolis, 06560 Valbonne, France

Communicated by M.F. Coste-Roy
Received 18 April 1989

This paper deals mainly with fast quantifier elimination in the elementary theory of algebraically closed fields of any characteristic. It is subdivided into an introduction, a short exposition of the computational model and of our results, and concludes with a section dedicated to proofs.

The new outcomes concern parallelism where the number of processors is controlled by the intrinsic sequential complexity of quantifier elimination. Our algorithms are optimal from the point of view of the overall complexities in parallel and in sequential (number of processors).

Due to recent progress concerning Triviality Testing of Polynomial Ideals (relying on effective affine Nullstellensätze) we are able to give upper bounds in a refined and satisfactory precise form.

1. Introduction

Many interesting geometric and algebraic problems can be formulated as first order statements about algebraically closed fields. In particular, the question whether a given finite set of multivariate polynomials has a zero in a suitable algebraically closed field can be stated in this language and can also be decided. More generally, if the coefficients of the polynomials are indeterminates, purely algebraic conditions can be given for instances of the coefficients in the algebraically closed field such that the polynomials have a common zero. This is a particular case of the general fact that the elementary theory of algebraically closed fields of given characteristic admits quantifier elimination.

In the last decade special effort has been made to find more efficient algorithms for algebraic and geometric problems, in particular for those problems which can be formulated as first order statements about algebraically closed fields.

* A preliminary version of this paper (A. Galligo, J. Heintz and J. Morgenstern: Parallelism and fast quantifier elimination over algebraically (and real) closed fields) has been presented at "Fundamentals of Computation Theory" (FCT'87), Kazan, USSR, 1987.

Special attention has been paid to the polynomial ideal membership problem, i.e. to the question of deciding whether a given polynomial is contained in a polynomial ideal given by a finite set of generators. Two kinds of algorithms have been developed for this purpose, one based on effective versions of Hilbert's Basissatz [19,21,24,25,29], the other one based on term rewriting [5,15,17]. The polynomial ideal membership problem is rather of algebraical than of geometrical nature and its worst case complexity is intrinsically high. (The ideal membership problem is exponential space complete [1,12,26].) This implies *doubly* exponential complexity bounds in the actual sequential algorithms (see e.g. [17,19]). By Hilbert's Nullstellensatz an algorithm which solves the polynomial ideal membership problem leads to a procedure which decides whether a given finite set of polynomials has a zero in an algebraically closed field. Those algorithms which are based on effective versions of Hilbert's Basissatz [21,24,25,29] can be used to compute projections of algebraic varieties. (This has been done in [20] and [19].)

Since quantifier elimination for algebraically closed fields has *intrinsically* doubly exponential sequential complexity [19,30], the doubly exponential degree bounds of the mentioned effective versions of Hilbert's Basissatz do not increase the overall complexity of the algorithms in [20] and [19] (see also [31]).

The only way to 'improve' the results consists in looking more closely at the parameters which determine the complexity bounds of the algorithms. Based on fundamental techniques [19] and [9], this has been done in [10] and [18], and the sequential complexity bounds obtained there are up to now as precise as possible. However, the complexity of the algorithm of [10] and [18] depends on the arithmetic properties of the field of constants from which the coefficients of the polynomial terms appearing in the input formula are taken. (This is due to the fact that polynomial factorization algorithms are used as subalgorithms.) Combining an affine effective Nullstellensatz (e.g. [7,8]) with the methods of [19], one obtains the same precise sequential complexity bounds as in [10] and [18]. The advantage of doing so consists in making the algorithms more transparent and in avoiding the dependence of the complexity bounds on the field of constants mentioned above. However, neither the algorithms of [19,20,31] nor the one of [10,18] are efficiently parallelizable because they contain subalgorithms which are inherently sequential.

In the present paper we construct an arithmetical network (see [16] for this notion) which for a given input formula of the first order theory language of algebraically closed fields of fixed characteristic computes an equivalent quantifier free one.

Our upper bounds for the complexity of quantifier elimination over algebraically closed fields are *precise* with respect to the following parameters: length, number and degree of the polynomials, total number of variables and *number of the quantifier alternations* of the (prenex) input formula. The (intrinsically) doubly exponential behaviour of the parallel complexity of quantifier elimination over algebraically closed fields depends only on the number of quantifier alternations in the input formula. Thus *for a fixed number of quantifier alternations* quantifier elimination over algebraically closed fields has simply exponential sequential and polynomial parallel complexity. This represents a refinement of the complexity results of [19, 20, 31] and also of [10, 18].

The precise and 'fast' algorithm for quantifier elimination presented here can also be used to decide the first order theory of algebraically closed fields of given characteristic within the same complexity limits.

Most of the 'mathematically interesting' decision problems of elementary algebraic geometry involve only a fixed, small number of quantifier alternations: for example, solvability of polynomial equations systems over an algebraically closed field or computation of dimension and degree of an algebraic variety. Our complexity result about quantifier elimination implies that these computational problems can be solved in simply exponential sequential and polynomial parallel time. (See [6, 7] for more details. Compare also [9] for the sequential aspect.)

Let us also observe that our algorithmical results can be transferred mutatis mutandis to the context of Turing complexity. (From our parallel complexity bounds or by direct inspection of our algorithm one easily sees that coefficient growth of intermediate polynomial manipulations remain under control. Compare [19] and [31] for technical aspects.)

The present paper covers also some aspects of *lower bounds.*

We give examples of formulae in the language of the first order theory of algebraically closed fields of given characteristic with the property that *any* quantifier elimination procedure applied to these formulas requires doubly exponential sequential time and simply exponential parallel time to represent the output. Thus our doubly exponential sequential and simply exponential parallel complexity bounds for quantifier elimination over algebraically closed fields are *intrinsic.*

Finally, let us mention that similar complexity results for quantifier elimination over *real closed fields* have recently been obtained by M.F. Coste-Roy and some of the (Noaï Fitchas) coauthors of the present paper: J. Heintz and P. Solernó.

Previous complexity results (upper and lower bounds) concerning parallel quantifier elimination over real and algebraically closed fields are contained in [14].

2. Short exposition of the computational model and of the results

In the sequel let $\mathbf{k}$ be an arbitrary field and $\bar{\mathbf{k}}$ an algebraically closed field containing $\mathbf{k}$ (e.g. its algebraical closure).

In order to speak about $\bar{\mathbf{k}}$ we consider the first order language L with the following non-logical symbols: For each $a \in \mathbf{k}$ we have a constant which we also call 'a'. Furthermore we have the function symbols $+, -, \cdot$ and the relation symbol $=$.

We consider the variables of L as indeterminates $X_1, \ldots, X_n$ over $\bar{\mathbf{k}}$, and we think the terms of our language L represented as multivariate polynomials with coefficients in $\mathbf{k}$ (dense representation).

So, a typical term has the form $F \in \mathbf{k}[X_1, \ldots, X_n]$, and a typical atomic formula has the form $F = 0$. For the negation of this formula, we write $F \neq 0$.

Our language is built up by atomic formulae using the logical connectives $\wedge, \vee, \neg$, and the first order quantifiers $\exists$, $\forall$ running over the elements of $\bar{\mathbf{k}}$ (not over subsets, relations or higher order predicates of $\bar{\mathbf{k}}$). So each formula $\Phi \in L$ is built up by atomic formulae involving polynomials, say $F_1, \ldots, F_s \in \mathbf{k}[X_1, \ldots, X_n]$. $X_1, \ldots, X_n$ are the variables of Φ.

We think of the language L as a set of words over the (infinite) set of symbols of L which consists of the non-logical symbols and the variables of L, the logical connectives, the quantifiers, and the brackets (and) .

So to each formula $\Phi \in L$ there corresponds its length $|\Phi|$ (number of symbols required to write down Φ).

We shall also use the following parameters which measure Φ:

$$\sigma(\Phi) := 2 + \sum_{1 \leq i \leq s} \deg F_i,$$

$n :=$ total number of variables contained in Φ.

If Φ is prenex (i.e. all quantifiers appear at the beginning of Φ) we also consider

$r :=$ number of alternations of blocks of existential and universal quantifiers.

We note that each formula $\Phi \in L$ can be brought in (sequential) time $O(|\Phi|)$ into an equivalent prenex form. This procedure does not change $|\Phi|$ and $\sigma(\Phi)$, and a possible increase of n is bound by the number of quantifiers contained in Φ.

Now we are going to give a short description of our computational model. The core of our algorithms consists in performing arithmetical operations $(+, -, \cdot)$ with multivariate polynomials over $\mathbf{k}$ which we think represented by their coefficient vectors in dense form. From the point of view of parallel complexity it is convenient to compute the coefficients of the resulting polynomials by interpolation using elements of $\bar{\mathbf{k}}$ (note that $\bar{\mathbf{k}}$ is infinite). Sometimes we have to ask whether or not the resulting polynomial is identically zero. So we also allow comparison of elements of $\bar{\mathbf{k}}$. The further development of the algorithm depends then on the result of these comparisons. Therefore we also have to use selectors. Of course, independent computations, comparisons, and selections can be executed simultaneously. To this kind of algorithm there corresponds the model of *'arithmetical network'* [16].

Our arithmetical networks use as input elements of $\mathbf{k}$, admit arithmetical and boolean operations including the use of constants from $\bar{\mathbf{k}}$, comparisons and selections of elements of $\bar{\mathbf{k}}$. They compute elements of $\mathbf{k}$ or boolean values. The notion of arithmetical network may be described by a directed, acyclical graph where each vertex represents an arithmetical or boolean operation, a comparison or a selection.

To each arithmetical network there correspond two notions of complexity:

(i) the parallel complexity or depth of the arithmetical network, i.e. the length of a longest directed path in the corresponding graph;

(ii) the sequential complexity or the number of processors or the size of the arithmetical network, i.e. the number of vertices of the corresponding graph.

(We avoid in the following the expression 'number of processors' for the size of an arithmetical network since it connotes the uneconomical assumption that the same processor is used only once in each execution of the algorithm. We prefer the expression 'sequential complexity'.)

In general fast sequential algorithms are not the best ones from a parallel point of view and vice versa. We are looking for fast parallel algorithms trying to control simultaneously their sequential complexity. This means concretely that the *sequential* complexity of our fast parallel algorithms is equal in order of magnitude to the complexity of other algorithms for the elimination of quantifiers known to be 'fast' in sequential time.

We leave to the imagination of the reader the part of our algorithmic model (network over the symbols of L) which corresponds to pure manipulation of formulae of L. For example, from the input formula Φ we have to extract the polynomials $F_1,\dots,F_s \in \mathbf{k}[X_1,\dots,X_n]$ contained in Φ. $F_1,\dots,F_s$ are then transformed by means of an arithmetical network into polynomials which appear in the final quantifier free output formula. Constructing this output formula requires once more manipulations in L. To be short, we admit the same kind of elementary operations with symbols of L as with elements of $\bar{\mathbf{k}}$: for example, concatenating words, interchanging or inserting symbols of L. In particular, these operations permit to transform thc formula $\Phi \in L$ into an equivalent prenex one in time $O(|\Phi|)$.

In the present paper we treat two different problems: the problem of upper bounds and the problem of lower bounds for the complexity of quantifier elimination in the language of the first order theory of algebraically closed fields of given characteristic.

2.1. Upper bounds for the complexity of quantifier elimination

The bounds for the *sequential* complexity of quantifier elimination over $\bar{\mathbf{k}}$ obtained in [19,20,31] are doubly exponential and of type

$$\sigma(\Phi)^{n^{O(n)}} \cdot |\Phi|.$$

[10] and [18], based on [9] and [19], give the most precise bound with respect to the parameters which measure the input:

$$\sigma(\Phi)^{n^{O(r)}} \cdot |\Phi|,$$

where the (inherent) double exponentiality of the bound depends only on r, the number of quantifier alternations in the formule $\Phi \in L$ which is supposed to be prenex. (However, the complexity changes if the field of constants $\mathbf{k}$ is extended, whereas the complexity of the algorithms presented in the present paper is independent from such extensions.)

Let us remark that the dependency on the term $|\Phi|$ in the complexity bounds above is inessential and omitted by most authors. It is due to some 'preprocessing' of the input formula Φ in order to control the logical connectives and to extract the polynomials which appear in Φ.

We present here the following results concerning upper bounds for the complexity of quantifier elimination in the first order theory of the algebraically closed field $\bar{\mathbf{k}}$:

Theorem 1. *For each $l \in \mathbb{N}$ there exists a network $\mathcal{N}_l$ over the symbols of L of depth $l^{O(l)}$ and size $l^{l^{O(l)}}$ with the following property*:

For each input formula $\Phi \in L$ with $|\Phi| = l$, $\mathcal{N}_l$ computes a quantifier free equivalent one (*with respect to the first order theory of* $\bar{\mathbf{k}}$). □

In other words, there exists an algorithm (the family of networks $\mathcal{N}_l$, $l \in \mathbb{N}$) which eliminates quantifiers (with respect to the first order theory of $\bar{\mathbf{k}}$)

$$\text{in parallel time (depth of } \mathcal{N}_l) \qquad |\Phi|^{O(|\Phi|)},$$

$$\text{and sequential time (size of } \mathcal{N}_l) \qquad |\Phi|^{|\Phi|^{O(\Phi)}},$$

where $\Phi \in L$ is an arbitrary input formula (of length $|\Phi| = l$).

Corollary. *There exists an algorithm which decides the elementary theory of $\bar{\mathbf{k}}$ in simply exponential parallel and doubly exponential sequential time, the input formula being measured by its length.* □

In Theorem 1 we used the length of the input formula as unique parameter.

For the next theorems we assume that $\Phi \in L$ is prenex and suitably 'prepared', this means that the polynomials appearing in Φ are explicitly given and that the quantifier free part of Φ can be written down with respect to the logical connectives in depth $O(\log |\Phi|)$.

We consider the more differentiated parameters $\sigma(\Phi)$, which measures number and degree of the polynomials, n, the total number of variables, and r, the number of quantifier alternations which appear in Φ.

Theorem 2. (i) *There exists an algorithm which eliminates quantifiers* (*with respect to first order theory of* $\bar{\mathbf{k}}$) *and which works*

$$\textit{in parallel time} \qquad n^{O(r)}(\log \sigma(\Phi))^{O(1)} + O(\log |\Phi|),$$

$$\textit{and sequential time} \qquad \sigma(\Phi)^{n^{O(r)}} \cdot |\Phi|.$$

(ii) *The same bounds are true for the complexity of the decision of the first order theory of* $\bar{\mathbf{k}}$.

Theorem 2 implies that the elimination of quantifiers and decision of the elementary theory of $\bar{\mathbf{k}}$ is in the complexity class NC (i.e. it has polylogarithmic parallel and polynomial sequential complexity) if the number of variables n of the input formula Φ is fixed.

The sequential result of Theorem 2 has been shown in [10] and [18] by a somewhat different way.

2.2. Lower bounds for the complexity of quantifier elimination

The aim of this subsection is to explain that our algorithms are optimal from the point of view of overall complexity. This means that the general problem of quantifier elimination over algebraically closed fields has an inherent simply exponential parallel and doubly exponential sequential complexity. Independent from how we *internally* realize quantifier elimination, we do need simply exponential parallel time or doubly exponential sequential time to *print* the output, a suitable quantifier free formula.

We have the following:

Theorem 3. *There exists a sequence of formulae (containing quantifiers and two free variables)* $\Phi_k \in L$, $k \in \mathbb{N}$, *with the following properties*:

(i) $|\Phi_k| = \mathrm{O}(k)$.

(ii) *For each quantifier free formula* $\theta \in L$ *equivalent to* Φ_k *involving the polynomials* $F_1, \dots, F_s$, *there exists* i, $1 \le i \le s$, *such that* $\deg F_i \ge 2^{2^{ck}}$, *where* $c > 0$ *is a suitable constant.*

Property (ii) implies $|\theta| \ge 2^{2^{ck}}$, since in our language L the polynomials (terms) are densely codified. In particular any *sequential* algorithm for quantifier elimination applied to the input Φ_k needs time *doubly* exponential in $|\Phi_k|$ to write down the quantifier free output formula (see [19] and [30]). Moreover any *parallel* algorithm for quantifier elimination applied to the input Φ_k needs time *simply* exponential in $|\Phi_k|$ to print the quantifier free output formula, since the output contains a polynomial of doubly exponential degree. To represent this polynomial we need an arithmetical network of at least simply exponential depth (compare [16]).

Let us remark here that our sequential lower bound depends on the assumption that polynomials are densely coded.

Analogous lower bounds are true for quantifier elimination over real closed fields (see [14]).

3. Proofs

In this section we are going to prove Theorem 2 and Theorem 3 for the first order theory of the algebraically closed field $\bar{\mathbf{k}}$. The proof follows the general lines of the sequential quantifier elimination procedures [19], [10] and [18]. However, since these algorithms do not give satisfactory results in parallel, one has to change crucial points of them. As a byproduct one obtains a new fast sequential algorithm for quantifier elimination over algebraically closed fields.

Let $\Phi \in L$ be an arbitrary *prenex* formula, containing n variables $X_1, \dots, X_n$, where $X_1, \dots, X_{n-m}$ are free and $X_{n-m+1}, \dots, X_n$ are bounded. So Φ has the form

$(Q_{n-m+1}X_{n-m+1})\cdots(Q_nX_n)\Psi(X_1,\dots,X_n)$, where $Q_{n-m+1},\dots,Q_n\in\{\exists,\forall\}$ and $\Psi(X_1,\dots,X_n)$ is a quantifier free formula being a boolean combination of atomic formulae containing polynomials, say $F_1,\dots,F_s\in\mathbf{k}[X_1,\dots,X_n]$ with

$$\sigma := \sigma(\Phi) = 2+\sum_{1\le i\le s}\deg F_i.$$

As in [19], we call a *nonempty* set $Z\subset\bar{\mathbf{k}}^n$ an $F_1,\dots,F_s$-cell, if there exists $\mathcal{M}\subset\{1,\dots,s\}$ such that

$$Z := \{x\in\bar{\mathbf{k}}^n\colon\ F_i(x)=0\ \forall i\in\mathcal{M} \text{ and } F_j(x)\neq 0\ \forall j\in\{1,\dots,s\}-\mathcal{M}\}.$$

Taking into account that $\forall$ can be expressed as $\neg\exists\neg$, we may assume that the last block of quantifiers (and the first to be eliminated) in our formula Φ is existential.

The first step of our algorithm consists in replacing $\Psi(X_1,\dots,X_n)$ by an equivalent disjunction of conjunctions of the form

$$\bigwedge_{i\in\mathcal{M}} F_i=0\wedge \bigwedge_{j\in\{1,\dots,s\}\setminus\mathcal{M}} F_j\neq 0,$$

where $\mathcal{M}\subset\{1,\dots,s\}$ determines a $F_1,\dots,F_s$-cell contained in the subset of $\bar{\mathbf{k}}^n$ defined by $\Psi(X_1,,\dots,X_n)$.

This is possible because the $F_1,\dots,F_s$-cells are the atoms of thc boolean lattice of the subsets of $\bar{\mathbf{k}}^n$ which can be defined by quantifier free formulas involving only the polynomials $F_1,\dots,F_s\in\mathbf{k}[X_1,\dots,X_n]$.

For this purpose we construct a network of depth $O(n^{O(1)}\log^3\sigma)$ and size $O(\sigma^{n^{O(1)}})$ which enumerates all $F_1,\dots,F_s$-cells. This can be done using [19, Corollary 1] (which is a consequence of the Bezout-Inequality), and an affine effective Nullstellensatz (see [7], [8] or [13] for such Nullstellensätze with elementary proofs). For the construction of the network we use a simple divide-and-conquer strategy, testing at each stage whether or not some already constructed conjunctions define really cells, i.e. nonempty subsets of $\bar{\mathbf{k}}^n$.

Stage by stage, we build a binary tree of depth $\log s$:

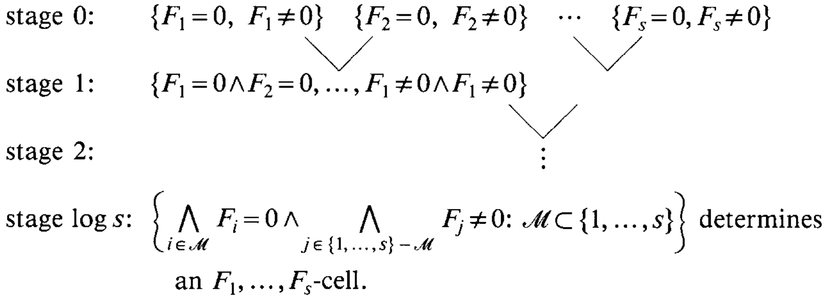
At stage 0 we determine which one of the subsets of $\bar{\mathbf{k}}^n$ defined by $F_i=0$ or $F_i\neq 0$ $(1\le i\le s)$ are non-empty.

At stage 1 we construct $s/2$ sets of *consistent* conjunctions of type:

$$\{F_1=0\wedge F_2=0,\ F_1=0\wedge F_2\neq 0,\ F_1\neq 0\wedge F_2=0,\ F_1\neq 0\wedge F_2\neq 0\},\dots,$$
$$\{F_{s-1}=0\wedge F_s=0, F_{s-1}=0\wedge F_s\neq 0, F_{s-1}\neq 0\wedge F_s=0, F_{s-1}\neq 0\wedge F_s\neq 0\}.$$

At stage k, $0\le k\le\log s$, we construct $s/(2^k)$ sets of consistent conjunctions by means of the adjacent sets of stage $k-1$.

In this way we obtain a tree of the following type:



stage 0: $\{F_1=0,\ F_1\neq 0\}$ $\{F_2=0,\ F_2\neq 0\}$ $\cdots$ $\{F_s=0, F_s\neq 0\}$

stage 1: $\{F_1=0\wedge F_2=0,\dots,F_1\neq 0\wedge F_1\neq 0\}$

stage 2: $\vdots$

stage $\log s$: $\left\{\bigwedge_{i\in\mathcal{M}} F_i=0\wedge \bigwedge_{j\in\{1,\dots,s\}-\mathcal{M}} F_j\neq 0\colon \mathcal{M}\subset\{1,\dots,s\}\right\}$ determines an $F_1,\dots,F_s$-cell.

Each of the $s/(2^k)$ sets constructed at stage k consists of at most 2^{2^k} different conjunctions which involve the polynomials $F_1,\dots,F_s$ whose sum of degrees is bounded by σ.

These conjunctions are consistent and—as mentioned before—constructed by means of consistent conjunctions of adjacent sets of stage $k-1$. This implies that we have to deal with at most σ^n conjunctions at stage k (see [19, Corollary 1]). This leads to at most $s/(2^k)\cdot\sigma^{2n}$ consistency tests at stage k which can be executed independently. In total we have at most $2s\sigma^{2n}$ consistency tests.

The description of our algorithm is complete if we explain how to perform a consistency test for a given conjunction, say $F_1=0\wedge\cdots\wedge F_{s'}=0\wedge F_{s'+1}\neq 0\wedge\cdots\wedge F_s\neq 0$.

We use the well known 'Rabinowitsch trick'.

Put $F:=F_{s'+1}\cdot\cdots\cdot F_s$ and let T be a new variable. According to Hilbert's Nullstellensatz, $F_1=0\wedge\cdots\wedge F_{s'}=0\wedge F_{s'+1}\neq 0\wedge\cdots\wedge F_s\neq 0$ is inconsistent if and only if the ideal generated by $F_1,\dots,F_{s'},1-TF$ in $\mathbf{k}[X_1,\dots,X_n]$ is trivial, that is if $1\in(F_1,\dots,F_{s'},1-TF)$. Now we use an affine effective Nullstellensatz with simply exponential degree bounds, e.g. [7, Theorem 16], or [8, Théorème]. We have

$$\begin{aligned}1\in(F_1,\dots,F_{s'},1-TF)\quad\text{iff}\quad &\exists P_1,\dots,P_{s'},P\in\mathbf{k}[X_1,\dots,X_n]\text{ such that}\\ &\deg P_iF_i,\ \deg P(1-TF)\leq\sigma^{(n+2)(n+3)/2}\\ &\text{for } 1\leq i\leq s',\text{ and such that}\\ &1=P_1F_1+\cdots+P_{s'}F_{s'}+P(1-TF).\end{aligned}$$

By comparison of coefficients our consistency test is now reduced to the question whether some inhomogeneous linear equation system of order $\sigma^{cn^3}\times\sigma^{cn^3}$ (for $c>0$ a suitable constant) whose coefficients come from the coefficients of $F_1,\dots,F_s$ and $1-TF$ is solvable.

In other words, we have to compare the rank of two matrices (over $\mathbf{k}$) of order $\sigma^{cn^3}\times\sigma^{cn^3}$. This leads to the problem of computing the rank of a matrix by an algorithm which is fast both in parallel and sequential time. Here we follow the treatment of [27]. (One could also use the algorithm of [11].)

Introducing a new variable Z, one transforms the given matrix into a new square one with entries from $\mathbf{k}[Z]$ whose characteristic polynomial we compute. The coefficients of the characteristic polynomial are obtained in parallel time $O(n^6\log\sigma^2)$

and in sequential time $\mathrm{O}(\sigma^{\mathrm{O}(n^3)})$ by [2]. From our construction (see [27] for details) it follows that the multiplicity of the root 0 in the characteristic polynomial indicates the rank of the matrix we started from. One eliminates the artificially introduced variable Z evaluating the coefficients of the characteristic polynomial (which are themselves polynomials over $\mathbf{k}$ of degree less than σ^{cn^3} in the indeterminate Z) in σ^{cn^3} points of $\bar{\mathbf{k}}$. Thus one sees which one is the first coefficient of the characteristic polynomial different from zero and hence which is the rank of the given matrix.

In its essence, the depth of our cell enumerating algorithm depends only on $\log s$ and on the depth of the algorithm [2] which computes the rank of a matrix (in our case of order $\sigma^{cn^3}\times\sigma^{cn^3}$). The sequential complexity of the cell enumerating algorithm is essentially controlled by the number of $F_1,\dots,F_s$-cells, which is bounded by σ^n, and the complexity of the rank computation above, which is $\sigma^{\mathrm{O}(n^3)}$.

Once the $F_1,\dots,F_s$-cells are enumerated, $\Psi(X_1,\dots,X_n)$ can be brought into the desired disjunctive form in parallel time $\mathrm{O}(\log|\Psi|)$ and sequential time $\mathrm{O}(|\Psi|)$.

Since the existential quantifiers and disjunctions commute, we have only to characterize the subsets of $\bar{\mathbf{k}}^{l-1}$ which are definable by formulas of type

$$(\exists X_l)\cdots(\exists X_n)\,(F_1=0\wedge\cdots\wedge F_{s'}=0\wedge F_{s'+1}\neq 0\wedge\cdots\wedge F_s\neq 0).$$

(Here $l\geq n-m+1$ indicates the length of the last block of quantifiers in our input formula which is supposed to be existential.)

In other words, by a quantifier free formula we try to characterize some projection of a given subset of $\bar{\mathbf{k}}^n$ which is locally closed in the Zariski topology. Let

$$\begin{aligned}D:=\{x\in\bar{\mathbf{k}}^{l-1}\colon (\exists X_l)\cdots(\exists X_n)\,&F_1(x,X_l,\dots,X_n)=0\wedge\cdots\wedge F_{s'}(x,X_l,\dots,X_n)=0\\ &\wedge F_{s'+1}(x,X_l,\dots,X_n)\neq 0\wedge\cdots\wedge F_s(x,X_l,\dots,X_n)\neq 0\}.\end{aligned}$$

We have to find polynomials $G_1,\dots,G_t\in\mathbf{k}[X_1,\dots,X_{l-1}]$ which describe D by a quantifier free formula.

Let $E:=\bar{\mathbf{k}}^{l-1}-D$, $F:=F_{s'+1}\cdot\cdots\cdot F_s$ and T be a new variable. By Hilbert's Nullstellensatz we have

$$E=\{x\in\bar{\mathbf{k}}^{l-1}\colon 1\in(F_1(x,X_l,\dots,X_n),\dots,F_{s'}(x,X_l,\dots,X_n),1-TF(x,X_l,\dots,X_n))\}.$$

We consider $F_1,\dots,F_{s'},1-TF$ as polynomials in $X_l,\dots,X_n,T$ with coefficients in $\mathbf{k}[X_1,\dots,X_{l-1}]$. We use again the affine effective Nullstellensatz of [7] or [8]. For the moment we replace $(X_1,\dots,X_{l-1})$ by $x\in\bar{\mathbf{k}}^{l-1}$. Then we have

$$1\in(F_1(x,X_l,\dots,X_n),\dots,F_{s'}(x,X_l,\dots,X_n),1-TF(x,X_l,\dots,X_n))$$

iff $\exists P_1,\dots,P_{s'}$, $P\in\bar{\mathbf{k}}[X_l,\dots,X_n]$, such that for $1\leq i\leq s'$,

$$\deg P_iF_i(x,X_l,\dots,X_n),\ \deg P(1-TF)(x,X_l,\dots,X_n)\leq\sigma^{(n-l+2)(n-l+3)/2}$$

and such that

$$1=P_1F_1(x,X_l,\dots,X_n)+\cdots+P_{s'}F_{s'}(x,X_l,\dots,X_n)+P(1-TF)(x,X_l,\dots,X_n)).$$

We compare coefficients of $X_l, \dots, X_n, T$ taking into account the degree bounds of $P_1, \dots, P_{s'}, P$ and remembering that $F_1, \dots, F_{s'}, 1-TF$ are considered as polynomials in $X_l, \dots, X_n, T$ with coefficients in $\mathbf{k}[X_1, \dots, X_{l-1}]$. So our task is reduced to consider the solvability of some inhomogeneous linear equation system. In other words, we have to describe a set of type

$$\left\{x \in \bar{\mathbf{k}}^{l-1} \colon \sum_{1 \leq j \leq p} F_{kj}(x) T_j = F_{kp+1}(x),\ 1 \leq k \leq q, \text{ has a solution in } \bar{\mathbf{k}}^p\right\}$$

by a quantifier free formula involving polynomials, say $G_1, \dots, G_t \in \mathbf{k}[X_1, \dots, X_{l-1}]$. (Here $F_{kj}, F_{kp+1} \in \mathbf{k}[X_1, \dots, X_{l-1}]$, $1 \leq k \leq q$, $1 \leq j \leq p$, are certain coefficients of $F_1, \dots, F_{s'}, 1-TF$ and $T_1, \dots, T_p$ are new variables describing the unknowns of our linear equation system. Moreover, we have $p, q \leq \sigma^{c(n-l)^3}$, where $c > 0$ is a suitable constant.)

Once more we use the algorithms of [27] and [2], essentially in the same way as before. These algorithms permit to exhibit necessary and sufficient polynomial conditions on $x \in \bar{\mathbf{k}}^{l-1}$ for the equality of the ranks of

$$(F_{kj}(x))_{1 \leq k \leq q, 1 \leq j \leq p} \quad \text{and} \quad (F_{kj}(x))_{1 \leq j \leq q, 1 \leq j \leq p+1}.$$

For this purpose we consider the (generic) matrices

$$(F_{kj}(X_1, \dots, X_{l-1}))_{1 \leq k \leq q, 1 \leq j \leq p} \quad \text{and} \quad (F_{kj}(X_1, \dots, X_{l-1}))_{1 \leq k \leq q, 1 \leq j \leq p+1},$$

and, as before, we look at the characteristic polynomials of the matrices obtained from the two given generic matrices applying the procedure of [27]. (Again, we have to use an auxiliary variable Z.)

Applying the algorithm of [2] we compute the coefficients of the two characteristic polynomials. These coefficients are elements of $\mathbf{k}[X_1, \dots, X_{l-1}, Z]$ and we consider them as polynomials in Z. Interpolating these polynomials in sufficiently many elements of $\bar{\mathbf{k}}$ ($\bar{\mathbf{k}}$ is infinite and the algorithm of [2] does not use divisions), i.e. specializing Z, we find their coefficients which represent, for $1 \leq i \leq t$, the $G_i \in \mathbf{k}[X_1, \dots, X_{l-1}]$, we are looking for.

The vanishing (the non-vanishing respectively) of the G_i, $1 \leq i \leq t$, in a given point $x = (x_1, \dots, x_{l-1}) \in \bar{\mathbf{k}}^{l-1}$ expresses the multiplicity of 0 in our two characteristic polynomials after specializing the variables $X_1, \dots, X_{l-1}$ to $x_1, \dots, x_{l-1}$. Thus, by our construction (see [27]) equality of the ranks of $(F_{kj}(x))_{1 \leq k \leq q, 1 \leq j \leq p}$ and $(F_{kj}(x))_{1 \leq k \leq q, 1 \leq j \leq p+1}$ can be expressed as a boolean combination of the atomic formulas $G_1(x) = 0, \dots, G_t(x) = 0$.

This means that the set

$$\left\{x \in \bar{\mathbf{k}}^{l-1} \colon \sum_{1 \leq j \leq p} F_{kj}(x) T_j = F_{kp+1}(x),\ 1 \leq k \leq q, \text{ has a solution in } \bar{\mathbf{k}}^p\right\}$$

can be described by a quantifier free formula involving only $G_1, \dots, G_t$. Since $\deg F_{kj} \leq \sigma$, $F_{kj} \in \mathbf{k}[X_1, \dots, X_{l-1}] \subset \mathbf{k}[X_1, \dots, X_n]$ for all $1 \leq k \leq q$, $1 \leq j \leq p+1$, and since $p, q \leq \sigma^{c(n-l)^3} \leq \sigma^{cn^3}$, an immediate analysis of the algorithms [27] and [2] gives $\sum_{1 \leq i \leq t} \deg G_i \leq \sigma^{cn^3}$. (Here $c > 0$ is a suitable constant.)

In the same way one sees that $G_1,\ldots,G_t$ can be computed in parallel time $\mathrm{O}(n^6\log^2\sigma)$ and sequential time $\sigma^{\mathrm{O}(n^4)}$.

The boolean combination of the atomic formulae $G_1=0,\ldots,G_t=0$ (the quantifier free formula) mentioned above can be represented by a network of depth $\mathrm{O}(n^3\log\sigma)$ and size $\sigma^{\mathrm{O}(n^3)}$.

This means that it is possible to eliminate one block of existential quantifiers in parallel time $\mathrm{O}(n^6\log^2\sigma)$ and sequential time $\sigma^{\mathrm{O}(n^4)}$ by a formula which involves only polynomials whose number and degree is of order $\sigma^{\mathrm{O}(n^3)}$.

The general bounds of Theorem 2 follow now by induction on the number of blocks of quantifiers. □

Theorem 1 is an immediate consequence of Theorem 2.

We are going to prove Theorem 3.

In [19] there is given a sequence of formulae $\Phi_k(X,Y)\in L$, $k\in\mathbb{N}$, in the two free variables X, Y with the following properties:

(i) $|\Phi_k|=\mathrm{O}(k)$,

(ii) Φ_k defines the graph of the application $\bar{\mathbf{k}}\to\bar{\mathbf{k}}$ which maps $x\in\bar{\mathbf{k}}$ onto $x^{2^{2^k}}$. In other words, Φ_k defines the set $M_k:=\{X^{2^{2^k}}-Y=0\}:=\{(x,y)\in\bar{\mathbf{k}}^2\colon x^{2^{2^k}}=y\}$.

Let $\theta\in L$ be a quantifier free formula equivalent to Φ_k which contains the polynomials $F_1,\ldots,F_s\in\mathbf{k}[X,Y]$ and let $G_1,\ldots,G_t$ be the prime factors of $F_1,\ldots,F_s$ in $\bar{\mathbf{k}}[X,Y]$. One has $\max\{\deg F_j\colon 1\le j\le s\}\ge\max\{\deg G_i\colon 1\le i\le t\}$. Therefore it suffices to show that there exists i, $1\le i\le t$, such that $\deg G_i\ge 2^{2^k}$.

One transforms $\theta\in L$ into a formula $\tilde\theta$ of the language $\tilde L$ with the non-logical symbols $\{a\colon a\in\bar{\mathbf{k}}\}\cup\{+,-,\cdot,=\}$, replacing each atomic formula $F_j=0$ with $F_j=G_{i_1}^{r_1}\cdot\cdots\cdot G_{i_m}^{r_m}$ $(i_1,\ldots,i_m\in\{1,\ldots,t\},r_1,\ldots,r_m\in\mathbb{N})$ by the expression $G_{i_1}=0\vee\cdots\vee G_{i_m}=0$.

Consequently $\tilde\theta$ involves only *irreducible* polynomials over $\bar{\mathbf{k}}$: $G_1,\ldots,G_t$. Moreover, we suppose that $\tilde\theta$ is a disjunction of *consistent* expressions:

$$(*)\qquad G_{i_1}=0\wedge\cdots\wedge G_{i_l}=0\wedge G_{i_{l+1}}\neq 0\wedge\cdots\wedge G_{i_t}\neq 0,$$

where all the G_i, $1\le i\le t$, appear. (Observe that, in principle, one cannot exclude the case $l=0$.)

$\tilde\theta$ is a quantifier free formula in two free variables, X,Y, equivalent to θ and consequently to Φ_k. So $\tilde\theta$ defines the subset $M_k=\{X^{2^{2^k}}-Y=0\}$ of $\mathbf{k}^2$, which is closed in the Zariski-topology. M_k is union of sets defined by conjunctions of type $(*)$. Since M_k is different from $\bar{\mathbf{k}}^2$ and is closed in the Zariski topology of $\bar{\mathbf{k}}^2$, we have $l\ge 1$ in each conjunction $(*)$. Taking into account that $G_{i_1},\ldots,G_{i_l}$ are absolutely irreducible, distinct polynomials, one infers from the Dimension Theorem [23, II, 7, Theorem 11] that $(*)$ defines a finite set, if $l\ge 2$. Therefore M_k can be written as a union of sets defined by conjunctions $(*)$ with $l=1$ and of finite sets.

Let us consider a conjunction $(*)$ with $l=1$ which appears in $\tilde\theta$. Since G_{i_1} is irreducible over $\bar{\mathbf{k}}$ and $(*)$ is consistent, the closure of the set defined by $(*)$ is

$\{G_{i_1}=0\}$. On the other hand we know that M_k is closed. So M_k can be written as a union of sets $\{G_{i_1}=0\}$ and of finite sets.

On the other hand, M_k is the graph $\{X^{2^{2^k}}-Y=0\}$ and therefore irreducible. $X^{2^{2^k}}-Y$ is an irreducible polynomial of $\bar{\mathbf{k}}[X,Y]$. With other words, $X^{2^{2^k}}-Y$ is the minimal equation of the irreducible hypersurface M_k of $\bar{\mathbf{k}}^2$.

So we have that M_k is irreducible and infinite and that the sets $\{G_{i_1}=0\}$ and the finite sets which appear in the decomposition of M_k are closed. Therefore there exists $i_1\in\{1,\dots,t\}$ such that $\{X^{2^{2^k}}-Y=0\}=M_k=\{G_{i_1}=0\}$. Since $X^{2^{2^k}}-Y$ is the minimal equation of M_k, $X^{2^{2^k}}-Y$ divides G_{i_1}, whence $\deg G_{i_1}\geq 2^{2^k}$. □

References

[1] P. Bayer and M. Stillman, On the complexity of computing syzygies, J. Symbolic Comput. 6 (1988) 135–147.

[2] S.J. Berkowitz, On computing the determinant in small parallel time using a small number of processors, Inform. Process. Lett. 18 (1984) 147–150.

[3] J. Briançon, Sur le degré des relations entre polynômes, C.R. Acad. Sci. Paris, Sér. I Math. (1982) 553–556.

[4] D. Brownawell, Bounds for the degrees in the Nullstellensatz, Ann. of Math. (2) 126 (3) (1987) 577–591.

[5] B. Buchberger, Ein algorithmisches Kriterium für die Lösbarkeit eines algebraischen gleichungssystems, Aequationes Math. 4 (1970) 374–383.

[6] L. Caniglia, Complejidad de algoritmos en geometría algebráica computacional. Ph.D. Thesis, University of Buenos Aires, 1989.

[7] L. Caniglia, A. Galligo and J. Heintz, Some new effectivity bounds in computational geometry, Lecture Notes in Computer Science 357 (Springer, Berlin, 1989) 131–151.

[8] L. Caniglia, A. Galligo and J. Heintz, Borne simple exponentielle pour les degrés dans le théorème des zéros sur un corps de caractéristique quelconque, C.R. Acad. Sci. Paris Sér. I Math. 307 (1988) 255–258.

[9] A.L. Chistov and D.Yu. Grigor'ev, Subexponential-time solving systems of algebraic equations I, II, LOMI Preprints E-9-83, E-10-83, Leningrad 1983.

[10] A.L. Chistov and D.Yu. Grigor'ev, Complexity of quantifier elimination in the theory of algebraically closed fields, Lecture Notes in Computer Science 176 (Springer, Berlin, 1984) 17–31.

[11] A.L. Chistov, Fast parallel calculation of the rank of matrices over a field of arbitrary characteristic, Lecture Notes in Computer Science 199 (Springer, Berlin, 1985) 63–69.

[12] M. Demazure, Notes informelles de calcul formel. 3. Le monoïde de Mayr-Meyer. 4. Le théorème de complexité de Mayr-Meyer, Prépublications Centre de Mathématiques de l'Ecole Polytechnique Paris, 1985.

[13] N. Fitchas and A. Galligo, Nullstellensatz effectif et Conjuncture de Serre (Théorème de Quillen-Suslin) pour le calcul formel, to appear in Math. Nachr.; preliminary version in Séminaire Structures Algébriques ordonnées 1987-88, UER de Math., Univ. Paris VII, 1989.

[14] N. Fitchas, A. Galligo and J. Morgenstern, Algorithmes rapides en séquentiel et en parallèle pour l'élimination des quantificateurs en géométrie élémentaire, to appear in Séminaire Structures Algébriques Ordonnées, Publ. Univ. Paris VII; preliminary version in Séminaire Structures Algébriques Ordonnées 1986-87, UER de Math., Univ. Paris VII, 1987.

[15] A. Galligo, Algorithmes de construction de bases standard, Preprint, University of Nice, 1983.

[16] J. von zur Gathen, Parallel arithmetic computations: a survey, Lecture Notes in Computer Science 233 (Springer, Berlin, 1986) 93–112.

[17] M. Giusti, Some effectivity problems in polynomial ideal theory, Lecture Notes in Computer Science 174 (Springer, Berlin, 1984) 159–171.

[18] D.Yu. Grigor'ev, The complexity of the decision for the first order theory of algebraically closed fields, Math. USSR-Izv. 29 (2) (1987) 459–475.

[19] J. Heintz, Definability and fast quantifier elimination over algebraically closed fields, Theoret. Comput. Sci. 24 (1983) 239–277; Russian translation in: Kyberneti̇̌českij Sbornik, Novaja Serija, Vyp.22, Mir Moscow (1985) 113–158.

[20] J. Heintz and R. Wüthrich, An efficient quantifier elimination algorithm for algebraically closed fields, SIGSAM Bull. 9 (1975) 11.

[21] G. Herrmann, Die Frage der endlich vielen Schritte in der Theorie der Polynomideale, Math. Ann. 95 (1926) 736–788.

[22] J. Kollár, Sharp effective Nullstellensatz, Preprint, 1988.

[23] S. Lang, Introduction to Algebraic Geometry, Interscience Tracts in Pure and Applied Mathematics (Interscience, New York, 1958).

[24] D. Lazard, Algèbre Linéaire sur $K[X_1, \ldots, X_n]$ et élimination, Bull. Soc. Math. France 105 (1981) 165–190.

[25] D.W. Masser and G. Wüstholz, Fields of large transcendence degree generated by values of elliptic functions, Invent. Math. 72 (1983) 407–464.

[26] E. Mayr and A. Meyer, The complexity of the word problem for commutative semigroups and polynomial ideals, Adv. in Math. 46 (1982) 305–329.

[27] K. Mulmuley, A fast parallel algorithm to compute the rank of a matrix over an arbitrary field, Proc. 18th Ann. ACM Symp. Theory of Computing (1986) 338–339.

[28] P. Philippon, Théorème des zéros effectif d'après J. Kollár, Séminaire I.H.P., 1988.

[29] A. Seidenberg, Constructions in algebra, Trans. Amer. Math. Soc. 197 (1974) 273–313.

[30] V. Weispfenning, The complexity of linear problems in fields. J. Symbolic Comput. 5 (1988) 3–28.

[31] R. Wüthrich, Ein schnelles Quantoreneliminationsverfahren für die Theorie der algebraisch abgeschlossenen Körper, Ph.D. Thesis, University of Zurich, 1977.