ELSEVIER

Journal of Pure and Applied Algebra 129 (1998) 173–200

JOURNAL OF PURE AND APPLIED ALGEBRA

An effective algorithm for quantifier elimination over algebraically closed fields using straight line programs

Susana Puddu*, Juan Sabia

Departamento de Matemática, FCEyN – Univ. de Buenos Aires, Ciudad Universitaria, Pab. I, (1428) Buenos Aires, Argentina

Communicated by M.-F. Roy; received 8 November 1995; received in revised form 15 April 1996

Abstract

In this paper we obtain an effective algorithm for quantifier elimination over algebraically closed fields: For every effective infinite integral domain k, closed under the extraction of pth roots when the characteristic p of k is positive, and every prenex formula φ with r blocks of quantifiers involving s polynomials $F_1,\dots,F_s \in k[X_1,\dots,X_n]$ encoded in dense form, there exists a well-parallelizable algorithm without divisions whose output is a quantifier-free formula equivalent to φ. The sequential complexity of this algorithm is bounded by $O(|\varphi|)+D^{(O(n))^r}$, where $|\varphi|$ is the length of φ and $D \geq n$ is an upper bound for $1+\sum_{i=1}^{s} \deg F_i$, and the polynomials in the output are encoded by means of a straight line program. The complexity bound obtained is better than the bounds of the known elimination algorithms, which are of the type $|\varphi|.D^{n^{cr}}$, where $c \geq 2$ is a constant. This becomes notorious when $r=1$ (i.e., when there is only one block of quantifiers): the complexity bounds known up to now are not less than D^{n^2}, while our bound is $D^{O(n)}$. Moreover, in the particular case that there is only one block of existential quantifiers and the input polynomials are given by a straight line program, we construct an elimination algorithm with even better bounds which depend on the length of this straight line program: Given a formula of the type

$$\exists x_{n-m+1},\dots,\exists x_n\colon F_1(x_1,\dots,x_n)=0 \wedge\cdots\wedge F_s(x_1,\dots,x_n)$$
$$=0 \wedge G_1(x_1,\dots,x_n)\neq 0 \wedge\cdots\wedge G_{s'}(x_1,\dots,x_n)\neq 0,$$

where $F_1,\dots,F_s \in k[X_1,\dots,X_n]$ are polynomials whose degrees in the m variables $X_{n-m+1},\dots,X_n$ are bounded by an integer $d \geq m$ and $G_1,\dots,G_{s'} \in k[X_1,\dots,X_n]$ are polynomials whose degrees in the same variables are bounded by an integer δ, this algorithm eliminates quantifiers in time $L^2.(s.s'.\delta)^{O(1)}.d^{O(m)}$, where L is the length of the straight line program that encodes $F_1,\dots,F_s,G_1,\dots,G_{s'}$.

Finally, we construct a fast algorithm to compute the Chow Form of an irreducible projective variety.

* Corresponding author. E-mail: spuddu@mate.dm.uba.edu.ar.

0022-4049/98/$19.00 © 1998 Elsevier Science B.V. All rights reserved
PII S0022-4049(97)00068-6

The construction of all the algorithms mentioned above relies on a preprocessing whose cost exceedes the complexity classes considered (they are based on the construction of correct test sequences). In this sense, our algorithms are non-uniform but may be considered uniform as randomized algorithms (choosing the correct test sequences randomly). © 1998 Elsevier Science B.V. All rights reserved.

AMS Classification: 68C25

1. Introduction

Let k be an arbitrary field and let $\overline{k}$ be an algebraic closed field such that $k \subseteq \overline{k}$. We will denote by $\mathscr{L}(k)$ the first order language over $\overline{k}$ with constants in k. It is well known that the elementary theory of algebraically closed fields of given characteristic admits quantifier elimination, i.e. for every formula $\varphi \in \mathscr{L}(k)$ there exists a quantifier-free formula $\psi \in \mathscr{L}(k)$ which describes the same subset of $\overline{k}^r$, where r is the number of variables of φ that are not quantified.

Many interesting geometric and algebraic problems can be formulated as first order statements over algebraically closed fields and they can be solved by means of quantifier elimination. This is why, in the last decades special efforts have been made to find efficient algorithms that eliminate quantifiers.

Given a formula $\varphi \in \mathscr{L}(k)$, let $|\varphi|$ be the length of φ, i.e., the number of symbols needed to encode φ, n be the number of indeterminates appearing in φ, $D = 1 + \sum_{i=1}^{s} \deg F_i$, where $F_1, \ldots, F_s \in k[X_1, \ldots, X_n]$ are the polynomials appearing in φ and, when φ is prenex, let r be the number of blocks of quantifiers in φ.

Heintz and Wüthrich (see [22, 26, 41]) exhibited elimination algorithms for algebraically closed fields of given characteristic with sequential time complexity bounded by $|\varphi|.D^{n^{cn}}$, where $c \geq 2$ is a constant. In fact, in the 1940s, Tarski knew the existence of elimination algorithms but he did not describe them explicitly (see [39]). Later, using the fundamental techniques described in [14, 22], Chistov and Grigor'ev considered the problem for prenex formulae and obtained in [15, 21] more precise sequential bounds of order $|\varphi|.D^{n^{cr}}$, where $c \geq 2$ is a constant. However, these bounds depend on arithmetic properties of the base field k because polynomial factorization algorithms are used as subalgorithms. None of the algorithms mentioned before are efficiently parallelizable (they contain subalgorithms which are inherently sequential).

Finally, in [16] a well-parallelizable elimination algorithm with the same sequential complexity bounds obtained in [15, 21] is constructed combining the methods in [22] with effective versions of Hilbert Nullstellensatz (see [1, 8, 9, 13, 29, 35], which improved the results of [3, 7, 11, 12]). Moreover, the complexity of this algorithm does not depend on particular properties of the base field k. Later, the same result was obtained in [27]. An important consequence of the parallelization is that quantifier elimination is possible in EXPSPACE (see [5, 6, 32]).

In the context of quantifier elimination, it is also worth mentioning the work of Renegar (see [36]) on elimination in real closed fields since the bounds obtained there are very sharp and imply the bounds for elimination over complex numbers.

In all these algorithms, the polynomials are coded in dense form (i.e. as arrays of elements of k) and, in this model, the sequential and parallel bounds obtained in [16] are optimal. This shows that it is impossible to get better bounds unless we change the way of coding polynomials.

A way of coding polynomials that showed to be effective to construct efficient algorithms to solve algebraic and geometric problems is the use of straight line programs: arithmetic circuits without branches nor selectors which evaluate the polynomials in any point (see, e.g. [17–20, 25, 28, 31]).

In this paper, we construct an effective elimination algorithm using the techniques to compute the dimension of an affine algebraic variety described in [18]. In order to do this, the polynomials will be encoded sometimes in dense form, sometimes by a straight line program and sometimes combining dense form and straight line programs.

The construction of this algorithm will be done in several steps (in any case, the way of coding the input and output polynomials will be specified).

First, we will consider prenex formulae with only one block of existential quantifiers and no inequalities. Then we will adapt the previous algorithm to prenex formulae with only one block of existential quantifiers which may contain inequalities. Then, we will use these algorithms to construct an algorithm without divisions for the general case.

As an application, we will use the previous algorithms to construct a new one that computes the Chow Form of an irreducible projective variety (see [10]).

The complexity bounds of our algorithms are better than the known bounds. Moreover all the algorithms exhibited in this paper are well-parallelizable and non-uniform in the sense that, for their construction, they require a preprocessing whose cost exceedes the complexity classes considered here. Nevertheless, this preprocessing, which consists in choosing some numbers, can be replaced by a random selection with a low probability of failure. In this sense, our algorithms are uniform with the same order of average complexity if we think of them as randomized algorithms.

2. Preliminaries

We first introduce some basic notions and notation and then mention the algorithmic tools used.

2.1. Notations

Let k be an infinite integral domain. We suppose k to be effective; this means that the arithmetic operations (addition, subtraction, multiplication) and basic equality checking (comparison) between elements of k are realizable by algorithms. If k has positive

characteristic p, we also assume that k is closed under the extraction of pth roots and that the extraction of these roots is effective (i.e. done by an algorithm).

Let k' be the quotient field of k and $\bar{k}$ be an algebraic closure of k'. We denote by $\mathbf{A}^n(\bar{k})$ the n-dimensional affine space over $\bar{k}$, equipped with its Zariski topology and with its coordinate ring of polynomial functions. If $S \subseteq \mathbf{A}^n(\bar{k})$, $\overline{S}$ will denote, as usual, the closure of S with respect to the Zariski topology.

Let $X_1,\ldots,X_n$ be indeterminates over k. We denote the total degree of a polynomial $f \in k[X_1,\ldots,X_n]$ by $\deg(f)$ and its partial degree in $X_1,\ldots,X_i$ $(1 \leq i \leq n)$, by $\deg_{X_1,\ldots,X_i}(f)$. Given $f_1,\ldots,f_r \in k[X_1,\ldots,X_n]$, $\gcd_{X_n}(f_1,\ldots,f_r)$ denotes the greatest common divisor among $f_1,\ldots,f_r$ with respect to X_n (i.e. considering $f_1,\ldots,f_r$ as polynomials in $k(X_1,\ldots,X_{n-1})[X_n]$).

Let φ be a first order formula. We denote by $|\varphi|$ the length of φ, i.e. the number of symbols needed to encode φ.

2.2. Codification of polynomials

The polynomials we deal with in our algorithms will be encoded in one of the following ways:

(a) Dense form, that is, as arrays (vectors) of elements of k.

(b) Straight line programs, which are arithmetic circuits (networks without branches). They contain neither selectors nor (propositional) Boolean operations. (For exact definitions and elementary properties of the notion of straight line program we refer to [23, 37, 38, 40].)

Our straight line programs will not contain any division. This is of particular importance for equality checking.

(c) Combining both dense form and straight line programs (i.e. in dense form with respect to some distinguished variables and their coefficients, which are polinomials in the remaining ones, encoded by a straight line program).

2.3. Algorithmic tools

Our algorithms are essentially based on the techniques used in [18] to compute the dimension of an algebraic set and on the methods of effective linear algebra which rely on well-parallelizable algorithms without divisions. A cornerstone of these techniques is Berkowitz' well-parallelizable polynomial algorithm for computing all the coefficients of the characteristic polynomial of a square matrix over any domain [4]. These coefficients are represented by a straight line program without divisions. For computing the rank of an arbitrary linear equation system over any domain we combine Berkowitz' algorithm with a result of Mulmuley [33], which allows us to express the rank of an arbitrary matrix over any domain by the multiplicity of zero in the characteristic polynomial of some associated square matrix.

When applying these results, some new indeterminates are introduced. To eliminate them from the output, we will use a suitable "correct test sequence" of points with

coordinates in k according to [24, Theorem 4.4]. Although the choice of such a correct sequence could be done algorithmically, the cost of doing so would exceed the main complexity class considered in this paper. However, for fixed input parameters, this choice is independent of the problem. For this reason, we will suppose that the correct test sequence is given by means of a preprocessing whose cost will not be considered in the complexity bounds obtained and, therefore, our algorithms will be non-uniform as they depend on the choice of the correct test sequences. However, the quoted Theorem 4.4 allows us to randomly choose correct test sequences with a probability of failure which is always less than 1/262144 and becomes arbitrarily small as the parameters s, d and m increase. Therefore, our algorithms can be uniformly randomized, within the same order of (average) complexity (see, e.g. [2, 18, 19]).

When the characteristic p of k is positive, we will need to extract pth roots in extended base rings. These extractions will appear only in subroutines with no effect on final results and will have no influence on the global behavior of our algorithms (for more details, see [19, 1.2.2]).

In case $k = \mathbb{Z}$, each node of an arithmetic network corresponding to a fundamental operation in the base ring $\mathbb{Z}$ may be replaced by a Boolean circuit which processes bits. Taking into account the growth of the coefficients of the polynomials which appear as intermediate results of our algorithms, our arithmetic networks may be transformed in a natural way into Boolean ones of the same order of complexity and our results will remain valid *mutatis mutandis* for the bit complexity model of algorithms represented by Boolean networks but this requires a further analysis (for a similar analysis see [31]).

3. The fundamental case

In this section we will show an algorithm which eliminates quantifiers in prenex formulae with only one block of existential quantifiers. This algorithm uses both straight line programs and dense form to represent polynomials. Whenever it is necessary to change the codification of polynomials from straight line program to dense form, we will apply the method described in Section 3.1. In Section 3.2 we will consider formulae without inequalities. Then, in Section 3.3 we will exhibit an example which shows that the complexity bounds obtained are better than the bounds of any algorithm using only dense representation of polynomials. Finally, in Section 3.4 we will adapt the algorithm described in Section 3.2 to the case of formulae containing equalities and inequalities.

3.1. Putting straight line programs into dense form

The following proposition shows how to put some polynomials given by a straight line program into dense form with respect to some distinguished variables. Their coefficients will be given by a straight line program over the remaining variables.

Proposition 3.1.1. *Let m be an integer such that $1 \leq m \leq n$. Let $F_1, \ldots, F_s$ be polynomials in $k[X_1, \ldots, X_n]$ given by a straight line program of length $\mathscr{L}$. If $d \geq m$ is an integer such that $\deg_{X_{n-m+1},\ldots,X_n}(F_i) \leq d$ for all i, $1 \leq i \leq s$, then there exists a well-parallelizable algorithm without divisions of sequential time complexity $\mathscr{L}.d^{O(m)}$ whose output is the same set of polynomials $F_1, \ldots, F_s \in k[X_1, \ldots, X_{n-m}][X_{n-m+1}, \ldots, X_n]$ but now given in dense form in the variables $X_{n-m+1}, \ldots, X_n$. Their coefficients in $k[X_1, \ldots, X_{n-m}]$ will be given by a straight line program of length $\mathscr{L}.d^{O(m)}$.*

Proof. The idea is to put the polynomials into dense form in the last variable and to iterate this procedure m times.

Let $\varepsilon_0, \ldots, \varepsilon_d$ be $d+1$ different elements in k and let $A \in k^{(d+1)\times(d+1)}$ be the following matrix:

$$A = \begin{pmatrix} 1 & \varepsilon_0 & \varepsilon_0^2 & \cdots & \varepsilon_0^d \\ 1 & \varepsilon_1 & \varepsilon_1^2 & \cdots & \varepsilon_1^d \\ 1 & \varepsilon_2 & \varepsilon_2^2 & \cdots & \varepsilon_2^d \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & \varepsilon_d & \varepsilon_d^2 & \cdots & \varepsilon_d^d \end{pmatrix}.$$

First, we compute A^{-1} (as $\varepsilon_i \neq \varepsilon_j$ if $i \neq j$ the matrix A is non-singular). This can be done in sequential time $d^{O(1)}$.

For every i, $1 \leq i \leq s$,

$$F_i = \sum_{j=0}^{d} a_{ij} X_n^j \quad \text{where } a_{ij} \in k[X_1, \ldots, X_{n-1}].$$

Given $(\xi_1, \ldots, \xi_{n-1}) \subset k^{n-1}$, we want to compute $a_{ij}(\xi_1, \ldots, \xi_{n-1})$ for every i, j, $1 \leq i \leq s$, $0 \leq j \leq d$. In order to do this, for each i, $1 \leq i \leq s$, we consider the following linear system with coefficients in k:

$$\begin{cases} \displaystyle\sum_{j=0}^{d} \varepsilon_0^j z_{ij} = F_i(\xi_1, \ldots, \xi_{n-1}, \varepsilon_0), \\ \vdots \\ \displaystyle\sum_{j=0}^{d} \varepsilon_d^j z_{ij} = F_i(\xi_1, \ldots, \xi_{n-1}, \varepsilon_d). \end{cases}$$

As A is precisely the matrix associated to this system and the unique solution is $z_{ij} = a_{ij}(\xi_1, \ldots, \xi_{n-1})$, then

$$\begin{pmatrix} a_{i0}(\xi_1, \ldots, \xi_{n-1}) \\ \vdots \\ a_{id}(\xi_1, \ldots, \xi_{n-1}) \end{pmatrix} = A^{-1} \cdot \begin{pmatrix} F_i(\xi_1, \ldots, \xi_{n-1}, \varepsilon_0) \\ \vdots \\ F_i(\xi_1, \ldots, \xi_{n-1}, \varepsilon_d) \end{pmatrix}.$$

Then, to compute $a_{ij}(\xi_1,\ldots,\xi_{n-1})$ $(1\le i\le s,\ 0\le j\le d)$, it is necessary to evaluate $F_1,\ldots,F_s$ in $d+1$ different points and then to multiply the matrices. The cost of this is $\mathscr{L}.(d+1)+2.s.(d+1)^2$.

Once we have the straight line program for the polynomials a_{ij} $(1\le i\le s,\ 0\le j\le d)$ we repeat the procedure for these polynomials and the next variable X_{n-1}.

After doing this m times, we obtain the desired straight line program of length $\mathscr{L}.d^{O(m)}$. □

3.2. *Formulae with only one block of existential quantifiers and no inequalities*

Let $X_1,\ldots,X_n$ be indeterminates over k and let m be an integer such that $1\le m\le n$. Let $F_1,\ldots,F_s$ be polynomials in $k[X_1,\ldots,X_n]$. Let $d\ge m$ and d' be integers such that $\deg_{X_{n-m+1},\ldots,X_n}(F_i)\le d$ and $\deg_{X_1,\ldots,X_{n-m}}(F_i)\le d'$ for every i, $1\le i\le s$. We will assume that $F_1,\ldots,F_s\in k[X_1,\ldots,X_{n-m}][X_{n-m+1},\ldots,X_n]$ are given in dense form in $X_{n-m+1},\ldots,X_n$ and that their coefficients in $k[X_1,\ldots,X_{n-m}]$ are given by a straight line program of length L. Let $\mathscr{P}\subseteq \mathbf{A}^{n-m}(\bar{k})$ be the set:

$$\mathscr{P}=\{(x_1,\ldots,x_{n-m})\in\bar{k}^{n-m}/\exists(x_{n-m+1},\ldots,x_n)\in\bar{k}^m:$$
$$F_1(x_1,\ldots,x_n)=0\wedge\cdots\wedge F_s(x_1,\ldots,x_n)=0\}.$$

Under these hypotheses, we have the following:

Theorem 3.2.1. *There exists a well-parallelizable algorithm without divisions with sequential time complexity bounded by* $L+s^{O(1)}.d^{O(m)}$ *which describes the set* $\mathscr{P}$ *in the following way*:

$$\mathscr{P}=\{(x_1,\ldots,x_{n-m})\in\bar{k}^{n-m}/\psi(x_1,\ldots,x_{n-m})\},$$

where ψ *is a quantifier-free formula, i.e. a boolean combination of atomic formulae of the type*:

$$G_1(x_1,\ldots,x_{n-m})=0\wedge\cdots\wedge G_\lambda(x_1,\ldots,x_{n-m})$$
$$=0\wedge G_{\lambda+1}(x_1,\ldots,x_{n-m})\neq 0\wedge\cdots\wedge G_\mu(x_1,\ldots,x_{n-m})\neq 0,$$

where $G_1,\ldots,G_\mu$ *are polynomials in* $k[X_1,\ldots,X_{n-m}]$ *with degrees bounded by* $d'.d^{O(m)}$. *Moreover, the length of the formula* ψ *is bounded by* $L+s^{O(1)}.d^{O(m)}$, *the quantity of polynomials appearing is bounded by* $s^{O(1)}.d^{O(m)}$ *and they are given by a straight line program of length* $L+s^{O(1)}.d^{O(m)}$.

Proof. Let $(x_1,\ldots,x_{n-m})\in\bar{k}^{n-m}$ be a fixed point and let $K=k[x_1,\ldots,x_{n-m}]$. Let K' be the quotient field of K and let $\overline{K}$ be an algebraic closure of K'. For every i, $1\le i\le s$ let $f_i\in K[X_{n-m+1},\ldots,X_n]$ be the polynomial

$$f_i=F_i(x_1,\ldots,x_{n-m},X_{n-m+1},\ldots,X_n).$$

Then, $(x_1,\dots,x_{n-m})\in\mathscr{P}$ if and only if the closed subset of $\mathbf{A}^m(\overline{K})$

$$V=\{(x_{n-m+1},\dots,x_n)\in\overline{K}^m/f_1(x_{n-m+1},\dots,x_n)=0\wedge\cdots\wedge f_s(x_{n-m+1},\dots,x_n)=0\}$$

is non-empty. □

As the condition $V\neq\emptyset$ is equivalent to $\dim(V)\geq 0$, we intend to apply the algorithm given in [18] that computes the dimension of V (note that V is defined by s polynomials in m indeterminates whose coefficients are elements of the ring K, these polynomials are given in dense form and their degrees are bounded by d). The main problem that appears when we try to apply this algorithm is the equality checking: $(x_1,\dots,x_{n-m})$ can be any point in $\overline{k}^{n-m}$ and we cannot decide whether a given polynomial with coefficients in k evaluated in this point is zero or not. In order to solve this problem, we are going to modify the algorithm properly (we will consider $(x_1,\dots,x_{n-m})$ as parameters and follow all possible branchings of the process).

Like in [18], we introduce m^2+m new indeterminates T_{rj}, T_r $(1\leq r,j\leq m)$ and let $R=K[T_{rj},T_r]_{1\leq r,j\leq m}$. As before, let R' be the quotient field of R and let $\overline{R}$ be an algebraic closure of R'. If the characteristic p of k is positive, when we apply the algorithm of [18], we will need to extract pth roots of elements of R. These extractions will appear only in subroutines with no effect on final results and, as the number of iterations of this process is bounded by an integer known a priori, can be computed by replacing the variables involved by adequate powers of new indeterminates. For more details, see [19].

For every r, $1\leq r\leq m$ let $\lambda_r\in R[X_{n-m+1},\dots,X_n]$ be the linear form:

$$\lambda_r=T_{r1}X_{n-m+1}+\cdots+T_{rm}X_n+T_r.$$

Remark. Clearly, $V=\emptyset$ if and only if, for every r $(0\leq r\leq m)$ the closed subset of $\mathbf{A}^m(\overline{R})$

$$\begin{aligned}W_r=\{&(x_{n-m+1},\dots,x_n)\in\overline{R}^m/f_1(x_{n-m+1},\dots,x_n)\\&=0\wedge\cdots\wedge f_s(x_{n-m+1},\dots,x_n)=0\wedge\lambda_1(x_{n-m+1},\dots,x_n)\\&=0\wedge\cdots\wedge\lambda_r(x_{n-m+1},\dots,x_n)=0\}\end{aligned}\tag{1}$$

is empty.

Now, we can find equivalent conditions to $V\neq\emptyset$ using $W_0,\dots,W_m$:

Let $\Gamma=\{\gamma^{(1)},\dots,\gamma^{(c)}\}\subseteq k^m$ be a set with $c=(s+m)^{O(1)}.d^{O(m)}=s^{O(1)}.d^{O(m)}$ many elements, like in [18, 3.4.7]. Then, for every effective integral domain $\Re\supseteq k$ (effectivity here includes the extraction of pth roots when the characteristic p of k is positive) and for every closed subset of $\mathbf{A}^m(\overline{\Re})$ (where $\overline{\Re}$ denotes an algebraic closure of the quotient field of $\Re$)

$$\begin{aligned}W=\{&(x_{n-m+1},\dots,x_n)\in\overline{\Re}^m/h_1(x_{n-m+1},\dots,x_n)\\&=0\wedge\cdots\wedge h_{s+m}(x_{n-m+1},\dots,x_n)=0\},\end{aligned}$$

where $h_1,\dots,h_{s+m}\in\Re[X_{n-m+1},\dots,X_n]$ are polynomials given in dense form whose degrees are bounded by d, the well-parallelizable algorithm without divisions in [18, 3.4.7], computes an element $0\neq\alpha\in\Re$, an element $\gamma=(\gamma_1,\dots,\gamma_m)\in\Gamma$ and polynomials $r_1,\dots,r_m\in\Re[Z]$ with degrees in the indeterminate Z bounded by $d^{O(m)}$ such that every isolated point $x\in W$ satisfies

$$\alpha.x=(r_1(y(x)),\dots,r_m(y(x))),$$

where $y=\gamma_1X_{n-m+1}+\cdots+\gamma_mX_n$. (In fact, this is an effective refined version of the *theorem of the primitive element*.)

The sequential time complexity of this algorithm is $(s+m)^{O(1)}.d^{O(m)}=s^{O(1)}.d^{O(m)}$. We can choose $\Gamma\subset k^m$ because the coordinates of the elements in Γ can be chosen in any subset of $\Re$ provided that its cardinal is large enough and, therefore, in k.

Moreover, all the intermediate results of this algorithm are polynomials with coefficients in k, degrees bounded by $d^{O(m)}$ and given by a straight line program of length $s^{O(1)}.d^{O(m)}$, evaluated in the coefficients of $h_1,\dots,h_{s+m}$.

We intend to apply this algorithm to the closed sets $W_0,\dots,W_m$ in (1), i.e. to sets of the following type:

$$W=\{(x_{n-m+1},\dots,x_n)\in\overline{R}^m/h_1(x_{n-m+1},\dots,x_n)=0\wedge\cdots\wedge h_{s+m}(x_{n-m+1},\dots,x_n)=0\},\tag{2}$$

where $h_1,\dots,h_{s+m}$ are polynomials in $k[X_1,\dots,X_n,T_{ij},T_i]_{1\le i,j\le m}$ given in dense form with degrees bounded by d in the indeterminates $X_{n-m+1},\dots,X_n$ and their coefficients are polynomials in $k[X_1,\dots,X_{n-m},T_{ij},T_i]_{1\le i,j\le m}$ with degrees bounded by d' and given by a straight line program of length $L+m^2+m$ evaluated in the fixed point $(x_1,\dots,x_{n-m})$.

First, the algorithm in [18] computes, using techniques of effective linear algebra (to compute monomial bases, matrices of linear forms in a particular basis, standard bases, for instance) a polynomial g in $R[Y_1,\dots,Y_m]$ and polynomials $g_1,\dots,g_m$ in $R[Y_1,\dots,Y_m,Z]$ (where $Y_1,\dots,Y_m,Z$ are new indeterminates over R) that will be used later to find the element α and the polynomials $r_1,\dots,r_m$. As the mentioned algorithm needs to compare elements of the extended base ring and we cannot decide whether an element $\beta\in R$ is zero or not, any time we need to decide this we will consider the two possibilities: $\beta=0$, $\beta\neq0$. For each of them, we will continue with the algorithm until we obtain the polynomials $g,g_1,\dots,g_m$. This will produce branches (selectors) B_j $(1\le j\le b)$, where $b\le s^{O(1)}d^{O(m)}$ of the following type:

$$B_j:\bigwedge_{i\in M}\beta_{ij}=0\wedge\bigwedge_{i\in N}\beta_{ij}\neq0,$$

where $\#(M)+\#(N)\le s^{O(1)}.d^{O(m)}$ and each $\beta_{ij}\in R$ is a polynomial given by a straight line program of length $s^{O(1)}.d^{O(m)}$ with degree bounded by $d^{O(m)}$, in the coefficients of $h_1,\dots,h_{s+m}$.

In this way, for every branch, we obtain polynomials $g,g_1,\dots,g_m$.

Finally, the algorithm in [18] finds the element α and the polynomials $r_1,\dots,r_m$ in $R[Z]$ using the elements of the set $\Gamma=\{\gamma^{(1)},\dots,\gamma^{(c)}\}\subseteq k^m$ in the following way:

First, it computes $g(\gamma^{(1)})$. If $g(\gamma^{(1)})\neq 0$, it produces the output $\alpha=g(\gamma^{(1)})$, $r_1(Z)=g_1(\gamma^{(1)},Z),\dots,r_m(Z)=g_m(\gamma^{(1)},Z)$. If $g(\gamma^{(1)})=0$, it computes $g(\gamma^{(2)})$. If $g(\gamma^{(2)})\neq 0$ the output will be $\alpha=g(\gamma^{(2)})$, $r_1(Z)=g_1(\gamma^{(2)},Z),\dots,r_m(Z)=g_m(\gamma^{(2)},Z)$. If $g(\gamma^{(2)})=0$ the algorithm will continue in a similar way.

For each branch obtained before, we continue with the algorithm using the corresponding polynomials $g,g_1,\dots,g_m$. As we cannot decide if $g(\gamma^{(i)})$ is zero or not for $\gamma^{(i)}\in\Gamma$, taking for every i, $(1\leq i\leq c)$

$$\alpha^{\gamma^{(i)}}=g(\gamma^{(i)}),\qquad r_1^{\gamma^{(i)}}(Z)=g_1(\gamma^{(i)},Z),\ \dots,\ r_m^{\gamma^{(i)}}(Z)=g_m(\gamma^{(i)},Z)$$

we consider all the possibilities and for every condition B_j this will produce new branches. In this way, we obtain a new algorithm containing branches $B_j^{(r)}$ $(1\leq r\leq c,\ 1\leq j\leq b)$ where $c=\#(\Gamma)\leq s^{O(1)}.d^{O(m)}$ and $b\leq s^{O(1)}.d^{O(m)}$ of the following type:

$$B_j^{(r)}:\ \bigwedge_{i\in M}\beta_{ij}=0\wedge\bigwedge_{i\in N}\beta_{ij}\neq 0\wedge\bigwedge_{i=1}^{r-1}\alpha_j^{\gamma^{(i)}}=0\wedge\alpha_j^{\gamma^{(r)}}\neq 0,$$

where $\#(M)+\#(N)\leq s^{O(1)}.d^{O(m)}$ and each $\beta_{ij}\in R$ and each $\alpha_j^{\gamma^{(i)}}\in R$ is a polynomial given by a straight line program of length $s^{O(1)}.d^{O(m)}$ with degree bounded by $d^{O(m)}$, evaluated in the coefficients of $h_1,\dots,h_{s+m}$. For each branch $B_j^{(r)}$, this new algorithm produces the output

$$\alpha=\alpha_j^{\gamma^{(r)}},\qquad r_1=r_1^{\gamma^{(r)}},\ \dots,\ r_m=r_m^{\gamma^{(r)}}.$$

Note that $\alpha_j^{\gamma^{(r)}}$ and the coefficients of $r_1^{\gamma^{(r)}},\dots,r_m^{\gamma^{(r)}}$ are polynomials with coefficients in k in the indeterminates $X_1,\dots,X_{n-m},T_{ij},T_i$ $(1\leq i,j\leq m)$, with degrees bounded by $d'.d^{O(m)}$, evaluated in $(x_1,\dots,x_{n-m})$.

Including the branches in the output, we obtain a new algorithm that, applied to a set W as in (2), produces an output of the type:

$$\{B_j^{(r)};\alpha_j^{\gamma^{(r)}};r_1^{\gamma^{(r)}},\dots,r_m^{\gamma^{(r)}}\}_{\substack{1\leq j\leq b\\ 1\leq r\leq c}}.$$

There exist unique j_0 and r_0 $(1\leq j_0\leq b$ and $1\leq r_0\leq c)$ such that the fixed point $(x_1,\dots,x_{n-m})$ satisfies $B_{j_0}^{(r_0)}$ (the existence is proved in [18, 3.4.7], and the uniqueness is obvious from the definition of $B_j^{(r)}$) and, as the corresponding $\alpha_{j_0}^{\gamma^{(r_0)}}$ is different from zero, every isolated point $x\in W$ satisfies:

$$\alpha_{j_0}^{\gamma^{(r_0)}}.x=(r_1^{\gamma^{(r_0)}}(y(x)),\dots,r_m^{\gamma^{(r_0)}}(y(x))).$$

As it is impossible to decide which is the j_0 and which is the r_0 that correspond to the fixed point $(x_1,\dots,x_{n-m})$, we continue with the algorithm for every j and every r $(1\leq j\leq b$ and $1\leq r\leq c)$ in the following way.

For every h_i $(1 \le i \le s+m)$ that appears in the definition of W, let

$$P_i^{j,r} = (\alpha_j^{\gamma(r)})^d . h_i\left(\frac{r_1^{\gamma(r)}}{\alpha_j^{\gamma(r)}}, \ldots, \frac{r_m^{\gamma(r)}}{\alpha_j^{\gamma(r)}}\right) \in R[Z].$$

When $j = j_0$ and $r = r_0$, as every isolated point $x \in W$ satisfies

$$\alpha_j^{\gamma(r)} . x = (r_1^{\gamma(r)}(y(x)), \ldots, r_m^{\gamma(r)}(y(x)))$$

and $\alpha_j^{\gamma(r)} \neq 0$, the polynomials $P_i^{j,r}$ satisfy:

(•) if W has isolated points, $\gcd(P_1^{j,r}, \ldots, P_{s+m}^{j,r}) \neq 1$ (as polynomials in $R'[Z]$);

(••) if $\gcd(P_1^{j,r}, \ldots, P_{s+m}^{j,r}) \neq 1$ in $R'[Z]$, $W \neq \emptyset$.

Furthermore, if either $W = \emptyset$ or every $x \in W$ is an isolated point, from conditions (•) and (••) we deduce:

$$W = \emptyset \Leftrightarrow \gcd(P_1^{j,r}, \ldots, P_{s+m}^{j,r}) = 1 \quad \text{in } R'[Z].$$

So, when we continue with the algorithm for every j, r $(1 \le j \le b,\ 1 \le r \le c)$, the condition $W = \emptyset$ is equivalent to

$$\bigvee_{\substack{1 \le j \le b \\ 1 \le r \le c}} (B_j^{(r)} \wedge \gcd(P_1^{j,r}, \ldots, P_{s+m}^{j,r}) = 1 \quad \text{in } R'[Z]).$$

Now, given j and r, $\gcd(P_1^{j,r}, \ldots, P_{s+m}^{j,r}) = 1$ in $R'[Z]$ if and only if there exist polynomials $Q_1^{j,r}, \ldots, Q_{s+m}^{j,r} \in R'[Z]$ with degree in Z bounded by $d^{O(m)}$ such that

$$1 = \sum_{1 \le i \le s+m} P_i^{j,r} . Q_i^{j,r} \tag{3}$$

if and only if the non-homogeneous linear system with coefficients in R given by (3) is solvable in R' if and only if the rank of the matrix associated to the system in (3) is equal to the rank of the matrix of its associated homogeneous system. Now we are going to obtain a straight line program for the coefficients of these matrices. As the polynomials h_i $(1 \le i \le s+m)$ are given in dense form in the indeterminates $X_{n-m+1}, \ldots, X_n$, if

$$h_i = \sum_{0 \le s_1 + \cdots + s_m \le d} a_{s_1, \ldots, s_m} (X_{n-m+1})^{s_1} \ldots (X_n)^{s_m}$$

then

$$P_i^{j,r} = \sum_{0 \le s_1 + \cdots + s_m \le d} a_{s_1, \ldots, s_m} (r_1^{\gamma(r)})^{s_1} \ldots (r_m^{\gamma(r)})^{s_m} (\alpha_j^{\gamma(r)})^{d - (s_1 + \cdots + s_m)}$$

and we can evaluate the polynomials $P_i^{j,r}$ $(1 \le i \le s+m)$ without divisions. Using Proposition 3.1.1 we put these polynomials into dense form with respect to the indeterminate Z (note that their degrees in this variable is bounded by $d^{O(m)}$) and their coefficients will be the entries of the former matrices.

Hence, the linear system (3) has $d^{O(m)}$ equations and $s^{O(1)}.d^{O(m)}$ indeterminates and its coefficients are polynomials in $k[X_1,\ldots,X_{n-m},T_{ij},T_i]_{1\le i,j\le m}$ with degrees bounded by $d'.d^{O(m)}$ given by a straight line program of length $L+s^{O(1)}.d^{O(m)}$ evaluated in $(x_1,\ldots,x_{n-m})$.

Let B be the matrix of the non-homogeneous system and B' be the matrix obtained by adding a column of zeros to the matrix associated to the homogeneous system (we do this in order to have two matrices of the same size). Using the techniques of [4, 33], we introduce two new indeterminates Z_1 and Z_2 and we compute the characteristic polynomials of the square matrices A and A' obtained from B and B' (both polynomials have the same degree because B and B' have the same size):

$$\mathscr{X}_A = G_0^{j,r} + G_1^{j,r}\lambda + \cdots + G_t^{j,r}\lambda^t + \lambda^{t+1},$$

$$\mathscr{X}_{A'} = H_0^{j,r} + H_1^{j,r}\lambda + \cdots + H_t^{j,r}\lambda^t + \lambda^{t+1},$$

where $G_u^{j,r}$, $H_u^{j,r}$ $(0\le u\le t)$ are polynomials in $k[X_1,\ldots,X_{n-m},T_{ik},T_i,Z_1,Z_2]_{1\le i,k\le m}$, with degrees bounded by $d'.d^{O(m)}$, given by a straight line program of length $L+s^{O(1)}.d^{O(m)}$, evaluated in $(x_1,\ldots,x_{n-m})$ and $t\le d^{O(m)}$.

Therefore, $\gcd(P_1^{j,r},\ldots,P_{s+m}^{j,r})=1$ in $R'[Z]$ if and only if the multiplicity of zero is the same in both characteristic polynomials and this is equivalent to the condition:

$$D_j^{(r)}\colon (G_0^{j,r}\neq 0\wedge H_0^{j,r}\neq 0)\vee(G_0^{j,r}=0\wedge H_0^{j,r}=0\wedge G_1^{j,r}\neq 0\wedge H_1^{j,r}\neq 0)$$
$$\vee\cdots\vee(G_0^{j,r}=0\wedge H_0^{j,r}=0\wedge\cdots\wedge G_{t-1}^{j,r}=0\wedge H_{t-1}^{j,r}=0\wedge G_t^{j,r}\neq 0\wedge H_t^{j,r}\neq 0).$$

So, if we have a set W like in (2) and either $W=\emptyset$ or every $x\in W$ is an isolated point, then:

$$W=\emptyset \Leftrightarrow \bigvee_{\substack{1\le j\le b\\ 1\le r\le c}} (B_j^{(r)}\wedge D_j^{(r)}).$$

Note that the polynomials appearing in conditions $B_j^{(r)}$ and $D_j^{(r)}$ are elements of $k[x_1,\ldots,x_{n-m},T_{ik},T_i,Z_1,Z_2]_{1\le i,k\le m}$ and we want to have equivalent conditions only involving elements of $k[x_1,\ldots,x_{n-m}]$. As the polynomials that appear in $B_j^{(r)}$ and $D_j^{(r)}$ are given by a straight line program of length $v=s^{O(1)}.d^{O(m)}$ (because the polynomials $F_1,\ldots,F_s$ are given in dense form in the indeterminates $X_{n-m+1},\ldots,X_n$) and have degrees bounded by $D\le d^{O(m)}$ in the m^2+m+2 indeterminates T_{ik},T_i,Z_1,Z_2 $(1\le i,k\le m)$, we will use [24] to obtain the desired conditions.

Let $\Delta\subseteq k^{m^2+m+2}$ be a set of cardinal $6(v+m^2+m+2).(v+m^2+m+3)\le s^{O(1)}.d^{O(m)}$ such that

$$P=0 \Leftrightarrow P(\delta)=0 \quad \forall\delta\in\Delta$$

for every $P\in k[x_1,\ldots,x_{n-m}]$ $[T_{ik},T_i,Z_1,Z_2]_{1\le i,k\le m}$ with degree bounded by D and given by a straight line program of length bounded by v.

In this way, every condition

$$P(x_1,\ldots,x_{n-m},T_{ik},T_i,Z_1,Z_2)_{1\le i,k\le m}=0$$

is equivalent to

$$\bigwedge_{\delta\in\Delta} P(x_1,\ldots,x_{n-m},\delta)=0$$

and, in the same way,

$$P(x_1,\ldots,x_{n-m},T_{ik},T_i,Z_1,Z_2)_{1\le i,k\le m}\neq 0$$

is equivalent to

$$\bigvee_{\delta\in\Delta} P(x_1,\ldots,x_{n-m},\delta)\neq 0.$$

Then, when we apply the algorithm of sequential time complexity $L+s^{O(1)}.d^{O(m)}$ we have constructed to a set W like in (2), we obtain as an output a quantifier-free formula ψ_W which is a boolean combination of atomic formulae of the type

$$g_1(x_1,\ldots,x_{n-m})=0\wedge\cdots\wedge g_{h'}(x_1,\ldots,x_{n-m})$$
$$=0\wedge g_{h'+1}(x_1,\ldots,x_{n-m})\neq 0\wedge\cdots\wedge g_h(x_1,\ldots,x_{n-m})\neq 0$$

such that

(a) $|\psi_W|\le s^{O(1)}.d^{O(m)}$;

(b) every g_i is a polynomial in $k[X_1,\ldots,X_{n-m}]$ with degree bounded by $d'.d^{O(m)}$;

(c) the polynomials g_i are given by a straight line program of length $L+s^{O(1)}.d^{O(m)}$.

Moreover, if either $W=\emptyset$ or every $x\in W$ is an isolated point,

$$W=\emptyset \quad \text{if and only if } \psi_W.$$

Applying this algorithm to $W_0,\ldots,W_m$ (the sets defined in (1) which satisfy $V=\emptyset\Leftrightarrow W_r=\emptyset\ \forall r,\ 0\le r\le m$), we obtain as an output the formulae $\psi_0,\ldots,\psi_m$ (where $\psi_r=\psi_{W_r},\ 0\le r\le m$).

Statement: $V=\emptyset\Leftrightarrow\bigwedge_{0\le r\le m}\psi_r$.

Proof. If $V=\emptyset$, then $W_r=\emptyset\ \forall r,\ 0\le r\le m$. Then $\psi_r\ \forall r,\ 0\le r\le m$.

On the other hand, if $\bigwedge_{0\le r\le m}\psi_r$, then ψ_m. As either $W_m=\emptyset$ or every $x\in W_m$ is an isolated point (note that $\lambda_1,\ldots\lambda_m$ are generic linear forms), then $W_m=\emptyset$. So (using genericity again), either $W_{m-1}=\emptyset$ or every $x\in W_{m-1}$ is an isolated point and, as ψ_{m-1}, $W_{m-1}=\emptyset$. Iterating this, we see that $W_r=\emptyset$ for every r, $0\le r\le m$. Hence $V=\emptyset$ and this concludes the proof of the statement.

Then

$$V\neq\emptyset\ \Leftrightarrow\ \bigvee_{0\le r\le m}\neg\psi_r=\psi(x_1,\ldots,x_{n-m}).$$

From the construction of the algorithm, it is clear that the polynomials appearing in ψ do not depend on the fixed point $(x_1,\ldots,x_{n-m})$. So, $\psi(X_1,\ldots,X_{n-m})$ is a quantifier-free formula satisfying:

$$\{(x_1,\ldots,x_{n-m})\in\overline{k}^{n-m}/\exists(x_{n-m+1},\ldots,x_n)\in\overline{k}^m:$$

$$F_1(x_1,\ldots,x_n)=0\wedge\cdots\wedge F_s(x_1,\ldots,x_n)=0\}$$

$$=\{(x_1,\ldots,x_{n-m})\in\overline{k}^{n-m}/\psi(x_1,\ldots,x_{n-m})\}. \qquad \square$$

Remark 3.2.2. In case the polynomials $F_1,\ldots,F_s$ are given by a straight line program of length $\mathscr{L}$ in the indeterminates $X_1,\ldots,X_n$, putting them into dense form in the indeterminates $X_{n-m+1},\ldots,X_n$ with coefficients in $k[X_1,\ldots,X_{n-m}]$ given by a straight line program (see Proposition 3.1.1) we can apply the algorithm in Theorem 3.2.1 for $L=\mathscr{L}.d^{O(m)}$. Moreover, if the polynomials $F_1,\ldots,F_s$ are given in dense form in the indeterminates $X_1,\ldots,X_n$, they can obviously be encoded in dense form with respect to the indeterminates $X_{n-m+1},\ldots,X_n$ with their coefficients in $k[X_1,\ldots,X_{n-m}]$ given by a straight line program of length $L=s.d'^{O(n-m)}.d^{O(m)}$ and, again, we can apply the algorithm in Theorem 3.2.1.

3.3. Example

Example 3.3.1. Let d and r be positive integers such that $d\geq r$ and let φ be the formula:

$$\exists X_1\exists X_2\ldots\exists X_{r-1}: X_1^d.Y_1-1=0\wedge X_2^d.Y_2-X_1=0\wedge$$

$$\wedge X_3^d.Y_3-X_2=0\wedge\cdots\wedge X_{r-1}^d.Y_{r-1}-X_{r-2}=0\wedge Y_r^d-X_{r-1}=0.$$

Applying the algorithm described in Theorem 3.2.1 we obtain, in time $d^{O(r)}$, an equivalent quantifier-free formula. Following the ideas in [16], we will show now that the sequential complexity bound of any algorithm that eliminates quantifiers using only dense representation of polynomials must be, in this case, at least d^{r^2}.

Let P be the polynomial $P=Y_r^{d^r}.Y_{r-1}^{d^{r-2}}.Y_{r-2}^{d^{r-3}}\ldots Y_2^d.Y_1-1$. Obviously, φ is equivalent to the quantifier-free formula $P=0$. Let ψ be a quantifier-free formula equivalent to φ. Then $V=\{(y_1,\ldots,y_r)\in\overline{k}^r/\psi(y_1,\ldots,y_r)\}$ is the set of all the zeros of P. Note that, as P is irreducible, V is an irreducible closed set of dimension $r-1$.

Let $G_1,\ldots,G_t$ be the polynomials involved in ψ. Then V can be described in the following way:

$$V=\bigcup\{(y_1,\ldots,y_r)\in\overline{k}^r/G_{i_1}(y_1,\ldots,y_r)=0\wedge G_{i_2}(y_1,\ldots,y_r)=0\wedge\cdots\wedge$$

$$\wedge G_{i_k}(y_1,\ldots,y_r)=0\wedge G_{i_{k+1}}(y_1,\ldots,y_r)\neq 0\wedge\cdots\wedge G_{i_t}(y_1,\ldots,y_r)\neq 0\}.$$

As V is closed,

$$V = \bigcup \overline{\{(y_1,\ldots,y_r)\in \bar{k}^r / G_{i_1}(y_1,\ldots,y_r)=0 \wedge G_{i_2}(y_1,\ldots,y_r)=0 \wedge \cdots \wedge}$$

$$\overline{\wedge\, G_{i_k}(y_1,\ldots,y_r)=0 \wedge G_{i_{k+1}}(y_1,\ldots,y_r)\neq 0 \wedge \cdots \wedge G_{i_t}(y_1,\ldots,y_r)\neq 0\}}.$$

V is irreducible, so it is one of the closed sets

$$\overline{\{(y_1,\ldots,y_r)\in \bar{k}^r / G_{i_1}(y_1,\ldots,y_r)=0 \wedge G_{i_2}(y_1,\ldots,y_r)=0 \wedge \cdots \wedge}$$

$$\overline{\wedge\, G_{i_k}(y_1,\ldots,y_r)=0 \wedge G_{i_{k+1}}(y_1,\ldots,y_r)\neq 0 \wedge \cdots \wedge G_{i_t}(y_1,\ldots,y_r)\neq 0\}}$$

and, as $\dim V = r - 1$, this closed set cannot be

$$\overline{\{(y_1,\ldots,y_r)\in \bar{k}^r / G_1(y_1,\ldots,y_r)\neq 0 \wedge \cdots \wedge G_t(y_1,\ldots,y_r)\neq 0\}}.$$

Then, there exists i, $(1 \le i \le t)$ such that

$$V \subseteq \{(y_1,\ldots,y_r)\in \bar{k}^r / G_i(y_1,\ldots,y_r)=0\}.$$

Then P divides G_i and, therefore, $\deg G_i \ge d^r$. Hence, if we encode G_i in dense form, the algorithm will have a sequential complexity not less than d^{r^2} (note that, a polynomial of degree d in r variables has $\binom{d+r}{r} = d^{O(r)}$ coefficients).

From this example, one may think that this bound could be improved by means of sparse encoding (i.e. not counting zero coefficients). Nevertheless, a simple change of variables, for example, may enlarge the sparse codification of polynomials. To show this, we can change every variable X_i by $X_i + 1$ and every Y_i by $Y_i + 1$ in the example above. The new bounds would be the same but none of the polynomials appearing will have a short sparse form of encoding.

3.4. *Formulae with only one block of existential quantifiers containing inequalities*

Let $X_1,\ldots,X_n$ be indeterminates over k and let m be an integer such that $1 \le m \le n$. Let $F_1,\ldots,F_s$ be polynomials in $k[X_1,\ldots,X_n]$. Let $d \ge m$ and d' be integers such that $\deg_{X_{n-m+1},\ldots,X_n}(F_i) \le d$ and $\deg_{X_1,\ldots,X_{n-m}}(F_i) \le d'$ for every i $(1 \le i \le s)$.

Let $G_1,\ldots,G_{s'}$ be polynomials in $k[X_1,\ldots,X_n]$. Let δ and δ' be integers such that $\deg_{X_{n-m+1},\ldots,X_n}(G_i) \le \delta$ and $\deg_{X_1,\ldots,X_{n-m}}(G_i) \le \delta'$ for every i $(1 \le i \le s')$.

We will assume that $F_1,\ldots,F_s,G_1,\ldots,G_{s'}$ are given by a straight line program of length L. Let $\mathscr{P} \subseteq \mathbf{A}^{n-m}(\bar{k})$ be the set:

$$\mathscr{P} = \{(x_1,\ldots,x_{n-m})\in \bar{k}^{n-m} / \exists (x_{n-m+1},\ldots,x_n)\in \bar{k}^m:$$

$$F_1(x_1,\ldots,x_n)=0 \wedge \cdots \wedge F_s(x_1,\ldots,x_n)=0 \wedge$$

$$G_1(x_1,\ldots,x_n)\neq 0 \wedge \cdots \wedge G_{s'}(x_1,\ldots,x_n)\neq 0\}$$

Let Y be a new indeterminate and let $G = 1 - Y.\prod_{1 \leq i \leq s'} G_i$. Using the ideas of Rabinowitz, $\mathscr{P}$ can be described in the following way:

$$\mathscr{P} = \{(x_1, \ldots, x_{n-m}) \in \overline{k}^{n-m} / \exists (x_{n-m+1}, \ldots, x_n, y) \in \overline{k}^{m+1}:$$
$$F_1(x_1, \ldots, x_n) = 0 \wedge \cdots \wedge F_s(x_1, \ldots, x_n) = 0 \wedge G(x_1, \ldots, x_n, y) = 0\}.$$

Hence, we have $s+1$ polynomials given by a straight line program of length $L + s' + 1$ with degrees in the indeterminates $X_{n-m+1}, \ldots, X_n, Y$ bounded by $D = \max\{d, s'.\delta + 1\}$ and degrees in the indeterminates $X_1, \ldots, X_{n-m}$ bounded by $D' = \max\{d', s'.\delta'\}$.

If we apply Theorem 3.2.1 (taking into account Remark 3.2.2), we obtain an algorithm with sequential time complexity $(L + s').D^{O(m)} + s^{O(1)}.D^{O(m)}$ that describes $\mathscr{P}$ by means of a quantifier-free formula ψ of length $(L + s').D^{O(m)} + s^{O(1)}.D^{O(m)}$. The polynomials appearing in the formula ψ are given by a straight line program of length $(L + s').D^{O(m)} + s^{O(1)}.D^{O(m)}$ and their degrees are bounded by $D'.D^{O(m)}$.

In this case, the bounds obtained depend on $s'^{O(m)}$ and $\delta^{O(m)}$. We will show that this dependence is not intrinsic of the problem.

Keeping the above notations and hypotheses, we have the following:

Theorem 3.4.1. *There exists a well-parallelizable algorithm without divisions with sequential time complexity bounded by* $L^2.(s.s'.\delta)^{O(1)}.d^{O(m)}$, *which describes the set* $\mathscr{P}$ *in the following way*:

$$\mathscr{P} = \{(x_1, \ldots, x_{n-m}) \in \overline{k}^{n-m} / \psi(x_1, \ldots, x_{n-m})\},$$

where ψ *is a quantifier-free formula, i.e. a boolean combination of atomic formulae of the type*:

$$H_1(x_1, \ldots, x_{n-m}) = 0 \wedge \cdots \wedge H_\lambda(x_1, \ldots, x_{n-m})$$
$$= 0 \wedge H_{\lambda+1}(x_1, \ldots, x_{n-m}) \neq 0 \wedge \cdots \wedge H_\mu(x_1, \ldots, x_{n-m}) \neq 0,$$

where $H_1, \ldots, H_\mu \in k[X_1, \ldots, X_{n-m}]$ *have degrees bounded by* $d'.\delta'.(s'.\delta)^{O(1)}.d^{O(m)}$. *Moreover, all the polynomials in the formula* ψ *are given by a straight line program of length* $L^2.(s.s'.\delta)^{O(1)}.d^{O(m)}$ *and the length of* ψ *is bounded by* $L^2.(s.s'.\delta)^{O(1)}.d^{O(m)}$.

Proof. The idea is to modify the algorithm of Theorem 3.2.1. First we put the polynomials $F_1, \ldots, F_s$ into dense form in the indeterminates $X_{n-m+1}, \ldots, X_n$ with their coefficients in $k[X_1, \ldots, X_{n-m}]$ given by a straight line program (see Proposition 3.1.1).

Again, let $(x_1, \ldots, x_{n-m}) \in \overline{k}^{n-m}$ be a fixed point and let $K = k[x_1, \ldots, x_{n-m}]$. As in Theorem 3.2.1, let K' be the quotient field of K and let $\overline{K}$ be an algebraic closure of K'.

For every $i (1 \leq i \leq s)$ let $f_i \in K[X_{n-m+1}, \ldots, X_n]$ be the polynomial

$$f_i = F_i(x_1, \ldots, x_{n-m}, X_{n-m+1}, \ldots, X_n)$$

and for every $j(1 \leq j \leq s')$ let $g_j \in K[X_{n-m+1}, \ldots, X_n]$ be the polynomial

$$g_j = G_j(x_1, \ldots, x_{n-m}, X_{n-m+1}, \ldots, X_n).$$

Let V be the closed subset of $\mathbf{A}^m(\overline{K})$

$$V = \{(x_{n-m+1}, \ldots, x_n) \in \overline{K}^m / f_1(x_{n-m+1}, \ldots, x_n) = 0 \wedge \cdots \wedge f_s(x_{n-m+1}, \ldots, x_n) = 0\}$$

and let U be the open subset of $\mathbf{A}^m(\overline{K})$

$$U = \{(x_{n-m+1}, \ldots, x_n) \in \overline{K}^m / g_1(x_{n-m+1}, \ldots, x_n) \neq 0 \wedge \cdots \wedge g_{s'}(x_{n-m+1}, \ldots, x_n) \neq 0\}.$$

As we did before, we introduce $m^2 + m$ new indeterminates $T_{rj}, T_r \quad (1 \leq r, j \leq m)$. Let $R = K[T_{rj}, T_r]_{1 \leq r,j \leq m}$, let R' be the quotient field of R and let $\overline{R}$ be an algebraic closure of R'.

For every r $(1 \leq r \leq m)$, let $\lambda_r \in R[X_{n-m+1}, \ldots, X_n]$ be the linear form:

$$\lambda_r = T_{r1} X_{n-m+1} + \cdots + T_{rm} X_n + T_r$$

and let W_r be the closed subset of $\mathbf{A}^m(\overline{R})$

$$W_r = \{(x_{n-m+1}, \ldots, x_n) \in \overline{R}^m / f_1(x_{n-m+1}, \ldots, x_n) = 0 \wedge \cdots \wedge f_s(x_{n-m+1}, \ldots, x_n) = 0 \wedge \lambda_1(x_{n-m+1}, \ldots, x_n) = 0 \wedge \cdots \wedge \lambda_r(x_{n-m+1}, \ldots, x_n) = 0\} \tag{4}$$

and let U' be the open subset of $\mathbf{A}^m(\overline{R})$

$$U' = \{(x_{n-m+1}, \ldots, x_n) \in \overline{R}^m / g_1(x_{n-m+1}, \ldots, x_n) \neq 0 \wedge \cdots \wedge g_{s'}(x_{n-m+1}, \ldots, x_n) \neq 0\}$$

Remark. $V \cap U = \emptyset$ if and only if $W_r \cap U' = \emptyset \quad \forall r,\ 0 \leq r \leq m$.

Proof. If $V \cap U = \emptyset$, introducing a new indeterminate Y over $\overline{K}$ and using the idea of Rabinowitz, we have that

$$\left\{ (x_{n-m+1}, \ldots, x_n, y) \in \overline{K}^{m+1} / f_1(x_{n-m+1}, \ldots, x_n) = 0 \wedge \cdots \wedge f_s(x_{n-m+1}, \ldots, x_n) = 0 \wedge 1 - y \cdot \prod_{1 \leq i \leq s'} g_i(x_{n-m+1}, \ldots, x_n) = 0 \right\}$$

is empty. Then, there exist polynomials $P_1,\ldots,P_{s+1}\in K'[X_{n-m+1},\ldots,X_n,Y]$ such that

$$1=\sum_{i=1}^{s}P_i.f_i+P_{s+1}.\left(1-Y.\prod_{1\le i\le s'}g_i\right)$$

(Hilbert Nullstellensatz). Then, for every r, $0\le r\le m$,

$$1=\sum_{i=1}^{s}P_i.f_i+\sum_{j=1}^{r}0.\lambda_j+P_{s+1}.\left(1-Y.\prod_{1\le i\le s'}g_i\right)$$

in $R'[X_{n-m+1},\ldots,X_n,Y]$. Therefore, $W_r\cap U'=\emptyset$.

On the other hand, if $W_r\cap U'=\emptyset$ for every r $(0\le r\le m)$, as $V\cap U\subset W_0\cap U'$, $V\cap U=\emptyset$ and this concludes the proof of the remark. □

Let Γ and c be like in Theorem 3.2.1 and let W be a closed set of the following type:

$$W=\{(x_{n-m+1},\ldots,x_n)\in\overline{R}^m/h_1(x_{n-m+1},\ldots,x_n)=0 \wedge\cdots\wedge h_{s+m}(x_{n-m+1},\ldots,x_n)=0\},\tag{5}$$

where $h_1,\ldots,h_{s+m}$ are polynomials in $k[X_1,\ldots,X_n,T_{ij},T_i]_{1\le i,j\le m}$ given in dense form with degrees bounded by d in the indeterminates $X_{n-m+1},\ldots,X_n$ and their coefficients are polynomials in $k[X_1,\ldots,X_{n-m},T_{ij},T_i]_{1\le i,j\le m}$ with degrees bounded by d' and given by a straight line program of length $L+m^2+m$ evaluated in the fixed point $(x_1,\ldots,x_{n-m})$.

We apply the first part of the algorithm described in Theorem 3.2.1 to a set W like in (5) to get an output of the type:

$$\left\{B_j^{(r)};\ \alpha_j^{\gamma^{(r)}};\ r_1^{\gamma^{(r)}},\ldots,r_m^{\gamma^{(r)}}\right\}_{\substack{1\le j\le b\\ 1\le r\le c}}$$

For every j,r we compute, like before, the polynomials $P_i^{j,r}\in R[Z](1\le i\le s+m)$ satisfying (•) and (••) when $j=j_0$ and $r=r_0$ (see Proof of Theorem 3.2.1).

Let $t\le d^{O(m)}$ be a bound for $\deg_Z(P_i^{j,r})(1\le i\le s+m;\ 1\le j\le b;\ 1\le r\le c)$.

For every j,r, let $G^{j,r}\in R[Z]$ be the polynomial

$$G^{j,r}=\left(\prod_{1\le i\le s'}\left(\alpha_j^{\gamma^{(r)}}\right)^{\delta}.g_i\left(\frac{r_1^{\gamma^{(r)}}}{\alpha_j^{\gamma^{(r)}}},\ldots,\frac{r_m^{\gamma^{(r)}}}{\alpha_j^{\gamma^{(r)}}}\right)\right)^t.$$

Statement: When $j=j_0$ and $r=r_0$, if either $W\cap U'=\emptyset$ or every $x\in W\cap U'$ is an isolated point of W, then $G^{j,r}$ satisfies:

$$W\cap U'=\emptyset\ \Leftrightarrow\ \gcd_Z(P_1^{j,r},\ldots,P_{s+m}^{j,r})\mid G^{j,r}.$$

Proof. If $\gcd_Z(P_1^{j,r},\ldots,P_{s+m}^{j,r}) \not| G^{j,r}$, then there exists $z\in\overline{R}$ such that $G^{j,r}(z)\neq 0$ and $\gcd_Z(P_1^{j,r},\ldots,P_{s+m}^{j,r})(z)=0$. (Note that the multiplicity of every root of $G^{j,r}$ is at least t and $\deg_Z(\gcd_Z(P_1^{j,r},\ldots,P_{s+m}^{j,r}))\leq t$).

Then, $P_i^{j,r}(z)=0 \quad \forall i\ (1\leq i\leq s+m)$ and $G^{j,r}(z)\neq 0$. Therefore,

$$\left(\frac{r_1^{\gamma^{(r)}}(z)}{\alpha_j^{\gamma^{(r)}}},\ldots,\frac{r_m^{\gamma^{(r)}}(z)}{\alpha_j^{\gamma^{(r)}}}\right)\in W\cap U'.$$

On the other hand, if $W\cap U'\neq\emptyset$, let $x\in W\cap U'$. Then, as x is an isolated point of W, it satisfies

$$x=\left(\frac{r_1^{\gamma^{(r)}}(y(x))}{\alpha_j^{\gamma^{(r)}}},\ldots,\frac{r_m^{\gamma^{(r)}}(y(x))}{\alpha_j^{\gamma^{(r)}}}\right)$$

(see Theorem 3.2.1). Then $P_i^{j,r}(y(x))=0 \quad \forall i\ (1\leq i\leq s+m)$ and $G^{j,r}(y(x))\neq 0$. Hence, $y(x)\in\overline{R}$ is a zero of $\gcd_Z(P_1^{j,r},\ldots,P_{s+m}^{j,r})$ but it is not a zero of $G^{j,r}$ and so $\gcd_Z(P_1^{j,r},\cdots,P_{s+m}^{j,r}) \not| G^{j,r}$. This concludes the proof of the statement. □

As we cannot decide which is j_0 and which is r_0, we will continue with the algorithm for every j and every r.

Now, $\gcd_Z(P_1^{j,r},\ldots,P_{s+m}^{j,r}) \mid G^{j,r}$ if and only if $\exists\, Q_1^{j,r},\ldots,Q_{s+m}^{j,r}\in R'[Z]$ such that

$$G^{j,r}=\sum_{i=1}^{s+m} Q_i^{j,r}P_i^{j,r} \tag{6}$$

and $\deg_Z(Q_i^{j,r})\leq d^{O(m)}+\deg_Z(G^{j,r})\leq \delta.s'.d^{O(m)}$. That is to say, the non-homogeneous linear system with coefficients in R given by (6) is solvable in R'. With the same methods we used before, this can be written as a polynomial condition $D_j^{(r)}$. The linear system has $\delta.s'.d^{O(m)}$ equations and $\sum_{i=1}^{s+m}(1+\deg_Z(Q_i^{j,r}))\leq \delta.s.s'.d^{O(m)}$ indeterminates.

As the coefficients of the matrices involved are the coefficients of $P_1^{j,r},\ldots,P_{s+m}^{j,r},G^{j,r}$, we put $P_1^{j,r},\ldots,P_{s+m}^{j,r}$ into dense form with respect to Z as we did in Theorem 3.2.1. In order to do the same with $G^{j,r}$, first we will find a straight line program avoiding divisions by $\alpha_j^{\gamma^{(r)}}$. Let Y be a new indeterminate and, for every i, $1\leq i\leq s'$, let

$$\overline{g}_i=g_i\left(Y.r_1^{\gamma^{(r)}},\ldots,Y.r_m^{\gamma^{(r)}}\right)\in k[x_1,\ldots,x_{n-m},T_{uv},T_u,Z,Y]_{1\leq u,v\leq m}.$$

We put $\overline{g}_i \quad (1\leq i\leq s')$ into dense form with respect to Y and, if a_{ik} is the coefficient of Y^k in $\overline{g}_i$, then

$$G^{j,r}=\left(\prod_{1\leq i\leq s'}\left(\sum_{0\leq k\leq\delta}a_{ik}\left(\alpha_j^{\gamma^{(r)}}\right)^{\delta-k}\right)\right)^t$$

In this way we obtain a straight line program for $G^{j,r}$ of length $L.s'.(s.\delta)^{O(1)}.d^{O(m)}$. Now, we put $G^{j,r}$ into dense form with respect to Z (see Proposition 3.1.1). Then, the

coefficients of the matrices involved are polynomials in $k[x_1,\ldots,x_{n-m},T_{uv},T_u]_{1\le u,v\le m}$ with total degrees bounded by $\delta.\delta'.s'.d'.d^{O(m)}$ given by a straight line program of length $L.(s'.s.\delta)^{O(1)}.d^{O(m)}$ evaluated in $(x_1,\ldots,x_{n-m})$.

Hence, there are at most $\delta.s'.d^{O(m)}$ polynomials in $D_j^{(r)}$ with degrees bounded by $\delta'.d'.(\delta.s')^{O(1)}.d^{O(m)}$ given by a straight line program of length $L.(s'.s.\delta)^{O(1)}.d^{O(m)}$.

Then, if W is like in (5) and either $W\cap U'$ is empty or every $x\in W\cap U'$ is an isolated point of W,

$$W\cap U'=\emptyset \Leftrightarrow \bigvee_{\substack{1\le j\le b\\ 1\le r\le c}} (B_j^{(r)}\wedge D_j^{(r)}).$$

Note that the polynomials appearing in conditions $B_j^{(r)}$ and $D_j^{(r)}$ are elements of $k[x_1,\ldots,x_{n-m},T_{uv},T_u,Z_1,Z_2]_{1\le u,v\le m}$, with degrees in the m^2+m+2 indeterminates T_{uv},T_u,Z_1,Z_2 $(1\le u,v\le m)$ bounded by $s'.\delta.d^{O(m)}$ and given by a straight line program of length $L.(s'.s.\delta)^{O(1)}.d^{O(m)}$. To have equivalent conditions only involving elements of $k[x_1,\ldots,x_{n-m}]$ we use [24] as we did in Theorem 3.2.1.

When we apply the algorithm of sequential time complexity $L^2.(s'.s.\delta)^{O(1)}.d^{O(m)}$ we have constructed to a set W like in (5) we obtain as an output a quantifier-free formula ψ_W which is a boolean combination of atomic formulae of the type

$$h_1(x_1,\ldots,x_{n-m})=0\wedge\cdots\wedge h_{k'}(x_1,\ldots,x_{n-m})=0$$
$$\wedge h_{k'+1}(x_1,\ldots,x_{n-m})\neq 0\wedge\cdots\wedge h_k(x_1,\ldots,x_{n-m})\neq 0$$

such that

(a) $|\psi_W|\le L^2.(s.s'.\delta)^{O(1)}.d^{O(m)}$;

(b) every h_i in $k[X_1,\ldots,X_{n-m}]$ has degree bounded by $\delta'.d'.(s'.\delta)^{O(1)}.d^{O(m)}$;

(c) the polynomials h_i are given by a straight line program of length $L^2.(s.s'.\delta)^{O(1)}.d^{O(m)}$.

Moreover, if either $W\cap U'=\emptyset$ or every $x\in W\cap U'$ is an isolated point of W, then

$W\cap U'=\emptyset$ if and only if ψ_W.

Applying this algorithm to $W_0,\ldots,W_m$ (the sets defined in (4)), we obtain as an output the formulae $\psi_0,\ldots,\psi_m$ (where $\psi_r=\psi_{W_r}$, $0\le r\le m$).

Statement: $V\cap U=\emptyset\Leftrightarrow\bigwedge_{0\le r\le m}\psi_r$.

Proof. If $V\cap U=\emptyset$, then $W_r\cap U'=\emptyset$ $\forall r\ (0\le r\le m)$. Then ψ_r $\forall r\ (0\le r\le m)$.

On the other hand, if $\bigwedge_{0\le r\le m}\psi_r$, then ψ_m. As either $W_m=\emptyset$ or every $x\in W_m$ is an isolated point, then either $W_m\cap U'=\emptyset$ or every $x\in W_m\cap U'$ is an isolated point of W_m and, therefore, ψ_m implies $W_m\cap U'=\emptyset$. As λ_m is a generic linear form, $W_{m-1}\cap\{\lambda_m=0\}=W_m$ and U' is an open set, using the dimension theorem we have that either $W_{m-1}\cap U'=\emptyset$ or every $x\in W_{m-1}\cap U'$ is an isolated point of W_{m-1}. Then ψ_{m-1} implies $W_{m-1}\cap U'=\emptyset$.

Iterating this, we see that $W_r\cap U'=\emptyset$ for every r $(0\le r\le m)$. Hence $V\cap U=\emptyset$ and this ends the proof of the statement.

Then

$$V \cap U \neq \emptyset \Leftrightarrow \bigvee_{0 \leq r \leq m} \neg \psi_r = \psi(x_1, \ldots, x_{n-m})$$

Like in Theorem 3.2.1, the polynomials appearing in ψ do not depend on the fixed point $(x_1, \ldots, x_{n-m})$. So, $\psi(X_1, \ldots, X_{n-m})$ is a quantifier-free formula satisfying:

$$\begin{aligned}\{(x_1, \ldots, x_{n-m}) \in \overline{k}^{n-m} / \exists (x_{n-m+1}, \ldots, x_n) \in \overline{k}^m : \\ F_1(x_1, \ldots, x_n) = 0 \wedge \cdots \wedge F_s(x_1, \ldots, x_n) = 0 \\ \wedge G_1(x_1, \ldots, x_n) \neq 0 \wedge \cdots \wedge G_{s'}(x_1, \ldots, x_n) \neq 0\} \\ = \{(x_1, \ldots, x_{n-m}) \in \overline{k}^{n-m} / \psi(x_1, \ldots, x_{n-m})\}. \quad \square\end{aligned}$$

Example 3.4.2. Let d and r be positive integers such that $d \geq r \geq 3$ and let φ be the formula:

$$\begin{aligned}\exists X_1 \exists X_2 \ldots \exists X_{r-1} : X_1^d . Y_1 - 1 = 0 \wedge X_2^d . Y_2 - X_1 = 0 \wedge X_3^d . Y_3 - X_2 = 0 \\ \wedge \cdots \wedge X_{r-1}^d . Y_{r-1} - X_{r-2} = 0 \wedge Y_r^d - X_{r-1} = 0 \wedge Y_2 . Y_r - X_1 \neq 0.\end{aligned}$$

It is clear that φ is equivalent to the quantifier-free formula:

$$Y_r^{d^r} . Y_{r-1}^{d^{r-2}} . Y_{r-2}^{d^{r-3}} \ldots Y_2^d . Y_1 - 1 = 0 \wedge Y_1 . Y_2^d . Y_r^d - 1 \neq 0.$$

Let $f, g \in k[Y_1, \ldots, Y_r]$ be the polynomials $f = Y_r^{d^r} . Y_{r-1}^{d^{r-2}} . Y_{r-2}^{d^{r-3}} \ldots Y_2^d . Y_1 - 1$ and $g = Y_1 . Y_2^d . Y_r^d - 1$.

As

$$\begin{aligned}\{(y_1, \ldots, y_r) \in \overline{k}^r / f(y_1, \ldots, y_r) = 0\} \\ = \{(y_1, \ldots, y_r) \in \overline{k}^r / f(y_1, \ldots, y_r) = 0 \wedge g(y_1, \ldots, y_r) = 0\} \\ \cup \{(y_1, \ldots, y_r) \in \overline{k}^r / f(y_1, \ldots, y_r) = 0 \wedge g(y_1, \ldots, y_r) \neq 0\}\end{aligned}$$

then

$$\begin{aligned}\{(y_1, \ldots, y_r) \in \overline{k}^r / f(y_1, \ldots, y_r) = 0\} \\ = \{(y_1, \ldots, y_r) \in \overline{k}^r / f(y_1, \ldots, y_r) = 0 \wedge g(y_1, \ldots, y_r) = 0\} \\ \cup \overline{\{(y_1, \ldots, y_r) \in \overline{k}^r / f(y_1, \ldots, y_r) = 0 \wedge g(y_1, \ldots, y_r) \neq 0\}}.\end{aligned}$$

Being $\{(y_1, \ldots, y_r) \in \overline{k}^r / f(y_1, \ldots, y_r) = 0\}$ irreducible (f is irreducible) and not equal to $\{(y_1, \ldots, y_r) \in \overline{k}^r / f(y_1, \ldots, y_r) = 0 \wedge g(y_1, \ldots, y_r) = 0\}$ (f does not divide g), it must be

$$\begin{aligned}\{(y_1, \ldots, y_r) \in \overline{k}^r / f(y_1, \ldots, y_r) = 0\} \\ = \overline{\{(y_1, \ldots, y_r) \in \overline{k}^r / f(y_1, \ldots, y_r) = 0 \wedge g(y_1, \ldots, y_r) \neq 0\}}.\end{aligned}$$

Following the ideas used in Section 3.3, it can easily be seen that the sequential complexity bound of any algorithm that eliminates quantifiers using only dense representation of polynomials must be, in this case, at least d^{r^2}.

However, applying the algorithm described in Theorem 3.4.1 we obtain a quantifier-free formula equivalent to φ in time $d^{O(r)}$. This shows that, in the case that the formula contains inequalities, the algorithm given in Theorem 3.4.1 is better than any possible algorithm which only uses dense representation of polynomials.

Remark 3.4.3. Let $X_1,\ldots,X_n$ be indeterminates over k, let $F_1,\ldots,F_s \in k[X_1,\ldots,X_n]$ be polynomials with total degrees bounded by an integer d such that $d \geq n$ and let $G_1,\ldots,G_{s'} \in k[X_1,\ldots,X_n]$ be polynomials with total degrees bounded by an integer δ. We assume that $F_1,\ldots,F_s,G_1,\ldots,G_{s'}$ are given by a straight line program of length L. Let $\mathscr{P} \subseteq \mathbf{A}^n(\bar{k})$ be the set

$$\mathscr{P} = \{x \in \bar{k}^n / F_1(x) = 0 \wedge \cdots \wedge F_s(x) = 0 \wedge G_1(x) \neq 0 \wedge \cdots \wedge G_{s'}(x) \neq 0\}.$$

Note that the condition "$\mathscr{P}$ is non-empty" can be described by means of the formula:

$$\exists x_1 \ldots \exists x_n : F_1(x_1,\ldots,x_n) = 0 \wedge \cdots \wedge F_s(x_1,\ldots,x_n) = 0$$
$$\wedge\, G_1(x_1,\ldots,x_n) \neq 0 \wedge \cdots \wedge G_{s'}(x_1,\ldots,x_n) \neq 0.$$

Then, using the algorithm given in Theorem 3.4.1, we can decide whether the set $\mathscr{P}$ is empty in sequential time $L^2.(s.s'.\delta)^{O(1)}.d^{O(n)}$

4. The general case

In this section we will construct an algorithm that eliminates quantifiers in any formula $\varphi \in \mathscr{L}(k)$. As every arbitrary formula φ can be transformed, using logic tools like concatenating words, interchanging or inserting symbols, into an equivalent prenex formula by means of a well-parallelizable algorithm of sequential complexity $O(|\varphi|)$ and this process does not modify $|\varphi|$ or the degrees of the polynomials involved or the number of indeterminates (see [16, 30]) we will assume without loss of generality that φ is prenex.

Let φ be a first order prenex formula of length $|\varphi|$ containing r blocks of quantifiers and let $F_1,\ldots,F_s \in k[X_1,\ldots,X_n]$ be all the polynomials involved in φ. Following the ideas shown in [16] and replacing the effective Nullstellensatz used there by the algorithm described in Theorem 3.4.1 to decide whether a closed set is empty or not (see Remark 3.4.3), we can find a disjunctive form for the formula φ. Then we can apply the algorithm described in Theorem 3.4.1 to eliminate the first block of quantifiers. Iterating this procedure, we obtain the following:

Theorem 4.1. *Let φ be a first order prenex formula of length $|\varphi|$ containing r blocks of quantifiers and let $F_1,\ldots,F_s \in k[X_1,\ldots,X_n]$ be all the polynomials involved in φ*

encoded in dense form. Let $D = \max\{1 + \sum_{i=1}^{s} \deg F_i, n, s\}$. *Then there exists a well-parallelizable algorithm without divisions with sequential time complexity bounded by* $O(|\varphi|) + D^{(O(n))^r}$ *which finds a quantifier-free formula* ψ *equivalent to* φ. *The length of* ψ *and the number of polynomials involved in it is bounded by* $D^{(O(n))^r}$. *Moreover, the output polynomials are given by a straight line program of length* $D^{(O(n))^r}$ *and their total degrees are bounded by* $D^{(O(n))^r}$.

The complexity of the algorithm of Theorem 4.1 is better than the complexities of the elimination algorithms known up to now (note that the best of these complexities is $D^{n^{c.r}}$ where $c \geq 2$ is a constant and our complexity is bounded by $D^{(c.n)^r}$ where c is a constant) and this shows the advantage of encoding the output in form of straight line programs.

A possibility to obtain an elimination algorithm with better bounds is to apply the results obtained in [20] which involve complexity bounds depending more intrinsecally on the geometry of the problem.

5. Computation of the Chow Form

Now, we will apply the algorithms given in Theorems 3.2.1 and 3.4.1 to obtain an algorithm which computes the Chow Form of an irreducible projective variety (see, e.g. [10]).

Let k be a field, let $\bar{k}$ be an algebraic closure of k and let $k[X_0,\ldots,X_n]$ be the ring of polynomials in the indeterminates $X_0,\ldots,X_n$ with coefficients in k. We assume that k is effective (when the characteristic p of k is positive we also assume that k is closed under the extraction of pth roots and that the extraction of these roots is done by an algorithm).

We will denote by $\mathbf{P}^n$ the n-dimensional projective space over $\bar{k}$.

Theorem 5.1. *Let* $F_1,\ldots,F_s \in k[X_0,\ldots,X_n]$ *be polynomials with total degrees bounded by an integer* d *satisfying* $d > n$. *Let* $V = \{x \in \mathbf{P}^n / F_1(x) = 0 \wedge \cdots \wedge F_s(x) = 0\}$ *be an irreducible projective variety and let* r *be its projective dimension. Then, there exists a well-parallelizable algorithm with sequential complexity bounded by* $s^{O(1)}.d^{O(nr)}$ *whose input is the set of polynomials* $\{F_1,\ldots,F_s\}$ *given in dense form and whose output is the Chow Form of* V, *given by a straight line program of length* $s^{O(1)}.d^{O(n)}$.

Proof. For every i, $0 \leq i \leq r$, let

$$L^{(i)} = Y_0^i X_0 + \cdots + Y_n^i X_n,$$

where Y_j^i $(0 \leq i \leq r, 0 \leq j \leq n)$ are new indeterminates over $k[X_0,\ldots,X_n]$.

Let $\Gamma \subseteq (\mathbf{P}^n)^{r+1} \times \mathbf{P}^n$ be the set

$$\Gamma = \{w \in (\mathbf{P}^n)^{r+1} \times \mathbf{P}^n / F_1(w) = 0 \wedge \cdots \wedge F_s(w) = 0 \wedge L^{(0)}(w) = 0 \wedge \cdots \wedge L^{(r)}(w) = 0\}.$$

Let $\pi:(\mathbf{P}^n)^{r+1}\times\mathbf{P}^n\to(\mathbf{P}^n)^{r+1}$ be the projection map. The set $\pi(\Gamma)$ is closed, irreducible and its codimension is 1 (see [34, Lemma 4]). Therefore, there exists an irreducible polynomial $F\in k[Y_j^i]_{0\le i\le r,0\le j\le n}$ such that

$$\pi(\Gamma)=\{y\in(\mathbf{P}^n)^{r+1}/F(y)=0\}.$$

The polynomial F is the Chow Form of the irreducible projective variety V.
Let $W=\{w\in\overline{k}^{(n+1)(r+1)}/F(w)=0\}\subseteq\mathbf{A}^{(n+1)(r+1)}(\overline{k})$. Then

$$W=\{(y_0^0,\dots,y_n^r)\in\overline{k}^{(n+1)(r+1)}/\varphi(y_0^0,\dots,y_n^r)\},$$

where φ is the formula:

$$\exists x_0\cdots\exists x_n : F_1(x_0,\dots,x_n)=0\wedge\cdots\wedge F_s(x_0,\dots,x_n)=0$$
$$\wedge L^{(0)}(x_0,\dots,x_n,y_0^0,\dots,y_n^0)=0\wedge\cdots\wedge L^{(r)}(x_0,\dots,x_n,y_0^r,\dots,y_n^r)=0.$$

Note that φ has only one block of existential quantifiers. Applying the algorithm described in Theorem 3.2.1 we obtain a quantifier-free formula ψ equivalent to φ. Then,

$$W=\{(y_0^0,\dots,y_n^r)\in\overline{k}^{(n+1)(r+1)}/\psi(y_0^0,\dots,y_n^r)\}.$$

Let $H_1,\dots,H_k$ be the polynomials involved in ψ.
Let $I=\{1,\dots,k\}$. For every $M\subseteq I$,

$$\left\{\bigwedge_{i\in M}H_i=0\wedge\bigwedge_{j\in I-M}H_j\neq0\right\}$$

will denote the algebraic set

$$\{w\in\overline{k}^{(r+1)(n+1)}/H_i(w)=0\,\forall i\in M\ \wedge\ H_j(w)\neq0\,\forall j\in I-M\}.$$

In the same way, if $G_1,\dots,G_h\in k[Y_j^i]_{0\le i\le r,0\le j\le n}$, for every i, $1\le i\le h$,

$$\{G_1=0\wedge\cdots\wedge G_i=0\wedge G_{i+1}\neq0\wedge\cdots\wedge G_h\neq0\}$$

will denote the algebraic set

$$\{w\in\overline{k}^{(r+1)(n+1)}/G_1(w)=0\wedge\cdots\wedge G_i(w)=0\wedge G_{i+1}(w)\neq0\wedge\cdots\wedge G_h(w)\neq0\}.$$

Let $C=\{M\subseteq I/M$ defines a $H_1,\dots,H_k$-cell$\}$ (see [16]) and let S be the subset of C such that

$$W=\bigcup_{M\in S}\left\{\bigwedge_{i\in M}H_i=0\wedge\bigwedge_{j\in I-M}H_j\neq0\right\}.$$

As $W=\{F=0\}$ is a closed set, then

$$W=\bigcup_{M\in S}\overline{\left\{\bigwedge_{i\in M}H_i=0\wedge\bigwedge_{j\in I-M}H_j\neq0\right\}}$$

and, as F is irreducible, then there exists $M_0 \in S$, $M_0 \neq \emptyset$ such that

$$W = \overline{\left\{ \bigwedge_{i \in M_0} H_i = 0 \wedge \bigwedge_{j \in I - M_0} H_j \neq 0 \right\}}.$$

If $u \in M_0$, then

$$W = \overline{\left\{ \bigwedge_{i \in M_0} H_i = 0 \wedge \bigwedge_{j \in I - M_0} H_j \neq 0 \right\}} \subseteq \{H_u = 0\}$$

and therefore $\{H_u \neq 0\} \cap W = \emptyset$.

On the other hand, let $u \in I$ be such that $\{H_u \neq 0\} \cap W = \emptyset$. If $u \notin M_0$, then

$$\left\{ \bigwedge_{i \in M_0} H_i = 0 \wedge \bigwedge_{j \in I - M_0} H_j \neq 0 \right\} \subseteq W \cap \{H_u \neq 0\} = \emptyset$$

and this is impossible because M_0 is a $H_1, \ldots, H_k$-cell.

Therefore, $M_0 = \{i \in I / \{H_i \neq 0\} \cap W = \emptyset\}$.

Let $P = \gcd(H_i,\ i \in M_0)$, let $G = rad(P)$ be the polynomial obtained by multiplying the irreducible polynomials which divide P (see [34]) and let $H = \prod_{j \in I - M_0} H_j$.

Statement: $F = G / \gcd(G, H)$

Proof. Given $i \in I$, $F \mid H_i$ if and only if $W \cap \{H_i \neq 0\} = \{F = 0\} \cap \{H_i \neq 0\} = \emptyset$ if and only if $i \in M_0$. Then $F \mid G$ and, as F is irreducible, F does not divide H. Therefore, F divides $G = [G / \gcd(G, H)].\gcd(G, H)$ and F does not divide $\gcd(G, H)$. Then F is an irreducible polynomial which divides $G / \gcd(G, H)$.

On the other hand, if $f \in k[Y_j^i]_{0 \leq i \leq r, 0 \leq j \leq n}$ is an irreducible polynomial which divides $G / \gcd(G, H)$, as G is square-free, then f divides G and f does not divide H. Therefore f divides H_i for every $i \in M_0$ and f does not divide H.

As

$$\{f = 0\} = \{f = 0 \wedge H \neq 0\} \cup \{f = 0 \wedge H = 0\}$$

then

$$\{f = 0\} = \overline{\{f = 0 \wedge H \neq 0\}} \cup \{f = 0 \wedge H = 0\}.$$

Then, as $\{f = 0\}$ is irreducible and $\{f = 0\} \neq \{f = 0 \wedge H = 0\}$ (because f does not divide H), it must be $\{f = 0\} = \overline{\{f = 0 \wedge H \neq 0\}}$.

Therefore

$$\{f = 0\} = \overline{\{f = 0 \wedge H \neq 0\}} \subseteq \overline{\left\{ \bigwedge_{i \in M_0} H_i = 0 \wedge \bigwedge_{j \in I - M_0} H_j \neq 0 \right\}} = W = \{F = 0\}$$

and this implies that $f \mid F$. Then it must be $f = F$.

This proves that F is the only irreducible polynomial which divides $G/\gcd(G,H)$. As $G/\gcd(G,H)$ is square-free (because G is square-free) then $G/\gcd(G,H)=F$. This concludes the proof of the statement. □

As the polynomials $H_1,\ldots,H_k$ were obtained applying the algorithm of sequential complexity $s^{O(1)}.d^{O(n)}$ described in Theorem 3.2.1 to the formula

$$\exists x_0\cdots\exists x_n : F_1(x_0,\ldots,x_n)=0 \wedge\cdots\wedge F_s(x_0,\ldots,x_n)=0 \wedge$$
$$\wedge L^{(0)}(x_0,\ldots,x_n,y_0^0,\ldots,y_n^0)=0 \wedge\cdots\wedge L^{(r)}(x_0,\ldots,x_n,y_0^r,\ldots,y_n^r)=0$$

then $k\le s^{O(1)}.d^{O(n)}$ and $H_1,\ldots,H_k$ are polynomials in $k[Y_j^i]_{0\le i\le r,0\le j\le n}$, with degrees bounded by $d^{O(n)}$, given by a straight line program of length $s^{O(1)}.d^{O(n)}$.

In order to find the set $M_0=\{i\in I/\{H_i\neq 0\}\cap W=\emptyset\}$, for every i, $1\le i\le k$, we decide whether $\{H_i\neq 0\}\cap W=\emptyset$ applying the algorithm described in Theorem 3.4.1 to the formula

$$\exists x_0\cdots\exists x_n\exists y_0^0\cdots\exists y_n^r : F_1(x_0,\ldots,x_n)=0 \wedge\cdots\wedge F_s(x_0,\ldots,x_n)=0 \wedge$$
$$\wedge L^{(0)}(x_0,\ldots,x_n,y_0^0,\ldots,y_n^0)=0 \wedge\cdots\wedge$$
$$\wedge L^{(r)}(x_0,\ldots,x_n,y_0^r,\ldots,y_n^r)=0 \wedge H_i(y_0^0,\ldots,y_n^r)\neq 0.$$

The sequential complexity of this step is $s^{O(1)}.d^{O(nr)}$. (Note that, if we had used the algorithm described in Theorem 3.2.1 and Rabinowitz' trick, the complexity bound would have been $s^{O(1)}.d^{O(n^2r)}$.)

Once we have found the set M_0, we compute $F=G/\gcd(G,H)$ using the techniques in [28] (i.e. making generic transformations to obtain polynomials which are monic in one of the variables and then applying an algorithm that uses linear algebra to compute the greatest common divisor for polynomials in only one variable). These techniques do not modify the order of complexity. So, the sequential complexity of the algorithm we have constructed is $s^{O(1)}.d^{O(nr)}$. □

Note that the bound obtained to compute the Chow Form of V is, in some sense, intrinsic as it depends on the projective dimension r of V. This dimension can be computed using the algorithm in [18] in sequential time $s^{O(1)}.d^{O(n)}$.

Acknowledgements

We thank Joos Heintz for suggesting us this problem and for his helpful remarks. We also wish to thank the referee for his advices.

References

[1] F. Amoroso, Tests d'appartenance d'après un théorème de Kollár, Acad. Sci. Paris, Serie I Math. 309 (1989) 691–694.

[2] J.L. Balcázar, J. Díaz, J. Gabarró, Structural Complexity I, EATCS Monographs on Theoretical Computer Science, vol. 11, Springer, Berlin, 1988.

[3] C. Berenstein, A. Yger, Effective Bezout identities in $\mathbf{Q}[X_1,\ldots,X_n]$, Acta Math. 166 (1991) 69–120.

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

[5] A. Borodin, On relating time and space to size and depth, SIAM J. Comput. 6 (1977) 733–744.

[6] A. Borodin, J. von zur Gathen, J. Hopcroft, Fast parallel matrix and gcd computations, Proc. 23rd Annual Symp. FOCS, 1982, pp. 65–71.

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

[8] D. Brownawell, A prime power version of Nullstellensatz, Manuscript, 1989.

[9] L. Caniglia, Complejidad de algoritmos en geometría computacional, Thesis, Universidad de Buenos Aires, 1989.

[10] L. Caniglia, How to compute the Chow Form of an unmixed polynomial ideal in single exponential time, AAECC, Springer, Berlin, 1990.

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

[12] L. Caniglia, A. Galligo, J. Heintz, Some new effective bounds in computational geometry, Proc. 6th Internat. Conf. Applied Algebra, Algebraic Algorithms and Error Correcting Codes, Rome 1988, Lecture Notes in Computer Science, vol. 357, Springer, Berlin, 1989, pp. 131–151.

[13] L. Caniglia, J. Guccione, J.J. Guccione, Local membership problems for polynomial ideals, in: T. Mora, C. Traverso (Eds.), Proc. Internat. Conf. Effective Methods in Algebraic Geometry MEGA 90, Castiglioncello 1990, Progress in Mathematics, vol. 94, Birkhäuser, Basel, 1991, pp. 31–45.

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

[15] A.L. Chistov, D. Yu. Grigor'ev, Complexity of quantifier elimination in the theory of algebraically closed fields, Proc. 11th Symp. MFCS 1984, Lecture Notes in Computer Science, vol. 176, Springer, Berlin, 1984, pp. 17–31.

[16] N. Fitchas, A. Galligo, J. Morgenstern, Precise sequential and parallel complexity bounds for quantifier elimination over algebraically closed fields, J. Pure Appl. Algebra 67 (1990) 1–14.

[17] N. Fitchas, M. Giusti, F. Smietanski, Sur le complexité du théorème des zéros, in: B. Brosowski, F. Deutsch, J. Gudatt (Eds.), Approximation and Optimization in the Caribbean II, Proc. 2nd Internat. Conf. on Approximation and Optimization, La Habana, 1993, Peter Lang, 1995, pp. 274–329.

[18] M. Giusti, J. Heintz, La détermination des points isolés et de la dimension d'une variété algébrique peut se faire en temps polynomial, in: D. Eiseinbud, L. Robbiano (Eds.), Proc. Cortona Conf. on Computational Algebraic Geometry and Commutative Algebra, Symposia Matematica, Vol. XXXIV, Istituto Nazionale di Alta Matematica, Cambridge University Press, Cambridge, 1993.

[19] M. Giusti, J. Heintz, J. Sabia, On the efficiency of effective Nullstellensätze, Comput. Complexity 3 (1993) 56–95.

[20] M. Giusti, J. Heintz, J.E. Morais, J. Morgenstern, L.M. Pardo, Straight line programs in geometric elimination theory, Manuscript, 1995.

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

[22] J. Heintz, Definability and fast quantifier elimination over algebraically closed fields, Theoret. Comput. Sci. 24 (1983) 239–277.

[23] J. Heintz, On the computational complexity of polynomials and bilinear mappings, a survey, in: L. Huguet, A. Poli (Eds.), Proc. 5th Internat. Conf. Applied Algebra, Algebraic Algorithms and Error Correcting Codes AAECC 5, Menorca, 1987, Lecture Notes in Computer Science, vol. 356, Springer, Berlin, 1989, pp. 269–300.

[24] J. Heintz, C.P. Schnorr, Testing polynomials which are easy to compute, Monographie de l'Enseignement Mathematique, vol. 30, Impr. Kundig, Geneve, 1982, pp. 237–254.

[25] J.Heintz, M. Sieveking, Absolute primality of polynomials is decidable in random polynomial time in the number of the variables, 8th Internat. Colloquium on Automata, Languages and Programming ICALP 81, Lecture Notes in Computer Science, vol. 115, Springer, Berlin, 1981.
[26] J. Heintz, R. Wüthrich, An efficient quantifier elimination algorithm for algebraically closed fields, SIGSAM Bull. 9 (1975) 11.
[27] D. Ierardi, Quantifier elimination in the theory of an algebraically-closed field, J. ACM (1989) 138–147.
[28] E. Kaltofen, Greatest common divisors of polynomials given by straight line programs, J. ACM 35 (1) (1988) 231–264.
[29] J. Kollar, Sharp effective Nullstellensatz, J. AMS 1 (1988) 963–975.
[30] T. Krick, Complejidad para problemas de geometría elemental, Thesis, Universidad de Buenos Aires, 1990.
[31] T. Krick, L.M. Pardo, A computational method for diofantine approximation, Proc. MEGA'94, Progress in Mathematics, Birkhäuser, Basel, 1994.
[32] G. Matera, J.M. Turull, The space complexity of elimination: upper bounds, Manuscript, 1995.
[33] K. Mulmuley, A fast parallel algorithm to compute the rank of a matrix over an arbitrary field, Proc. 18th ACM Symp. Theory of Computing, 1986, pp. 338–339.
[34] Yu. V. Nesterenko, Estimates for the orders of zeros of functions of a certain class and applications in the theory of trascendental numbers, Math. USSR Izvestija, 11 (2) (1977) Izvestia Akad. Nauk. SSR Set-Mat. Tom 41 (2) (1977).
[35] P. Philippon, Théorème des zéros effectif d' après J. Kollár, Seminaire I.H.P. (1988).
[36] J. Renegar, On the computational complexity and geometry of the first-order theory of the reals, J. Symbol. Comput. 13 (1992) 253–352.
[37] H.J. Stoss, On the representation of rational functions of bounded complexity, Theoret. Comput. Sci. 64 (1989) 1–13.
[38] V. Strassen, Berechnung und Programm I, Acta Inform. 1 (1972) 320–334.
[39] A. Tarski, A Decision Method for Elementary Algebra and Geometry, 2nd ed., Univ. of California Press, Berkeley, CA, 1951.
[40] J. von zur Gathen, Parallel arithmetic computations: a survey, Proc. 13th Symp. MFCS 1986, Lecture Notes in Computer Science, vol. 233, Springer, Berlin, 1986, pp. 93–112.
[41] R. Wüthrich, Ein Quantoreneliminationverfahren für die Theorie der algebraisch abgeschlossenen Körper, Ph.D. Thesis, University of Zurich, 1977.