An Algorithm for the Computation of the Radical of an Ideal

Santiago Laplagne
Departamento de Matemática, Facultad de Ciencias Exactas y Naturales, Universidad de Buenos Aires
Buenos Aires, Argentina
slaplagn@dm.uba.ar

ABSTRACT

We propose a new algorithm for the computation of the radical of an ideal in a polynomial ring. In recent years many algorithms have been proposed. A common technique used is to reduce the problem to the zero dimensional case. In the algorithm we present here, we use this reduction, but we avoid the redundant components that appeared in other algorithms . As a result, our algorithm is in some cases more efficient.

Categories and Subject Descriptors

I.1.2 [**Symbolic and Algebraic Manipulation**]: Algorithms—*algebraic algorithms*

General Terms

Algorithms, Performance

Keywords

radical, primary decomposition, polynomial ideal, algorithms, complexity

1. INTRODUCTION

Let k be a field, $k[\boldsymbol{x}] := k[x_1, \ldots, x_n]$ the ring of polynomials in n variables and $I \subset k[\boldsymbol{x}]$ an ideal. The radical of I is the ideal

$$\sqrt{I} = \{f \in k[\boldsymbol{x}] \ / \ f^m \in I \text{ for some } m \in \mathbb{N}\}.$$

The radical of an ideal plays an important role in commutative algebra, when we are concerned with the geometry aspects. This is due to the bijection existing between varieties and radical ideals.

In recent years some algorithms for the computation of the radical have been proposed. Among these, we mention [9], [8] and [15] for the general case, [14] for the zero-dimensional case and [18] for ideals over fields of positive characteristic.

ISSAC'06, July 9–12, 2006, Genova, Italy.

In this paper we present a new algorithm for the computation of the radical of a general ideal. The algorithm presented here is based on the ideas of [9] and [15].

Recall that given I, J ideals in $k[\boldsymbol{x}]$, the quotient $I : J$ is the ideal $\{f \in k[\boldsymbol{x}] \ / \ fJ \subset I\}$ and the saturation $I : J^\infty$ is the ideal $\{f \in k[\boldsymbol{x}] \ / \ fJ^m \subset I \text{ for some } m \in \mathbb{N}\}$. When J is generated by a single element h, we use the notation $I : h$ and $I : h^\infty$.

In [15], the authors use the splitting tool $\sqrt{I} = \sqrt{I : h} \cap \sqrt{\langle I, h\rangle}$ for an appropriate h. They find h such that $\sqrt{I : h}$ can be obtained by reduction to the zero-dimensional case and obtain $\sqrt{\langle I, h\rangle}$ by induction on the dimension.

When taking $\langle I, h\rangle$ there appear redundant components (that is, components that were not part of the original ideal) that slow down the algorithm performance.

In the algorithm of this paper, we avoid using $\langle I, h\rangle$ but instead we use repeatedly the saturation $I : h^\infty$ for appropriate h.

This leads in some cases to a more efficient algorithm.

2. PRELIMINARIES

The algorithms that we present make extensive use of Gröbner bases and its applications. For an introduction to the subject, see for example [2], [3] and [11].

Given a monomial order $<$, if $f = a_1\boldsymbol{x}^{A_1} + a_2\boldsymbol{x}^{A_2} + \cdots + a_r\boldsymbol{x}^{A_r} \in k[\boldsymbol{x}]$ with $\boldsymbol{x}^{A_1} > \boldsymbol{x}^{A_2} > \cdots > \boldsymbol{x}^{A_r}$, we write $\mathrm{lt}(f)$ for the leading term $a_1\boldsymbol{x}^{A_1}$ and $\mathrm{lc}(f)$ for the leading coefficient a_1.

We note $\mathbf{V}_k(I)$ for the vanishing set of I in k^n and $\bar{k}$ for the algebraic closure of k. An ideal is called zero dimensional if $\mathbf{V}_{\bar{k}}(I)$ has only a finite number of points. In [9] and [15] the computation of the radical of a general ideal is reduced to the zero dimensional case. For the computation of the radical of a zero dimensional ideal, the following algorithm is used.

PROPOSITION 1 (SEIDENBERG LEMMA, [19]). *Let $I \subset k[\boldsymbol{x}]$ (with k a perfect field) be a zero dimensional ideal and $I \cap k[x_i] = \langle f_i\rangle$ for $i = 1, \ldots, n$. Let $g_i = \sqrt{f_i} = f_i/\gcd(f_i, f_i')$, the square free part of f_i. Then*

$$\sqrt{I} = \langle I, g_1, \ldots, g_n\rangle.$$

We will need to compute the radical of zero-dimensional ideals over $k(\boldsymbol{u})$, with $\boldsymbol{u}$ a set of variables. When k has characteristic 0, $k(\boldsymbol{u})$ is still perfect, so we can use this lemma. However, if k does not have characteristic 0, $k(\boldsymbol{u})$ might not be perfect. In this case, more elaborated algorithms ([14],

[18]) can be used. We will restrict to the case of characteristic 0.
The general algorithm is based on the following well known properties (see for example [11], Chapters 3 and 4).

LEMMA 2. *Let $I = Q_1 \cap \cdots \cap Q_t \subset k[\boldsymbol{x}]$ be a primary decomposition of the ideal I, and $J \subset k[\boldsymbol{x}]$ another ideal. Then $I : J^\infty = \bigcap_{J \not\subset P_i} Q_i$, where $P_i = \sqrt{Q_i}$.*

We say that $\boldsymbol{u} \subset \boldsymbol{x}$ is *independent* (with respect to I) if $I \cap k[\boldsymbol{u}] = \{0\}$. We say that an independent set is maximal if it has $\dim(I)$ elements.
If $\boldsymbol{u} \subset \boldsymbol{x}$ is a maximal independent set of variables with respect to I then $Ik(\boldsymbol{u})[\boldsymbol{x} \setminus \boldsymbol{u}] \subset k(\boldsymbol{u})[\boldsymbol{x} \setminus \boldsymbol{u}]$ is a zero-dimensional ideal, $Ik(\boldsymbol{u})[\boldsymbol{x} \setminus \boldsymbol{u}] \cap k[\boldsymbol{x}] = Q_1 \cap \cdots \cap Q_s$, where the Q_i are the primary components of I such that $Q_i \cap k[\boldsymbol{u}] = \{0\}$ and $\sqrt{Ik(\boldsymbol{u})[\boldsymbol{x} \setminus \boldsymbol{u}]} \cap k[\boldsymbol{x}] = P_1 \cap \cdots \cap P_s$, with $P_i = \sqrt{Q_i}$.
To contract an ideal $J \subset k(\boldsymbol{u})[\boldsymbol{x} \setminus \boldsymbol{u}]$ to $k[\boldsymbol{x}]$, we take $\{g_1, \ldots, g_s\}$, a Gröbner basis of J in a monomial order $<$ in $k(\boldsymbol{u})[\boldsymbol{x} \setminus \boldsymbol{u}]$ consisting of polynomials in $k[\boldsymbol{x}]$, and $h := \operatorname{lcm}\{\operatorname{lc}(g_i), 1 \leq i \leq s\} \in k[\boldsymbol{u}]$, where $\operatorname{lc}(g_i)$ denotes the leading coefficient of g_i considered as a polynomial in $k[\boldsymbol{u}][\boldsymbol{x} \setminus \boldsymbol{u}]$. Then $J \cap k[\boldsymbol{x}] = \langle g_1, \ldots, g_s \rangle : h^\infty$, where $\langle g_1, \ldots, g_s \rangle$ is the ideal generated in $k[\boldsymbol{x}]$.

Remark 1. The ideal $\langle g_1, \ldots, g_s \rangle : h^\infty$ can be computed algorithmically by a Gröbner basis calculation, using that $I : h^\infty = \langle I, th - 1 \rangle \cap k[\boldsymbol{x}]$, where t is a new variable ([9], Corollary 3.2).

Remark 2. A Gröbner basis of $Jk(\boldsymbol{u})[\boldsymbol{x} \setminus \boldsymbol{u}]$ can be obtained by computations in $k[\boldsymbol{x}]$ taking $\{f_1, \ldots, f_s\}$, a set of generators of J consisting of polynomials in $k[\boldsymbol{x}]$, and computing a basis of $\langle f_1, \ldots, f_s \rangle k[\boldsymbol{x}]$ with respect to a lexicographical order with $\boldsymbol{x} \setminus \boldsymbol{u} > \boldsymbol{u}$.
Note that the generators of J in $k[\boldsymbol{x}]$ can be obtained from any set of generators of J by simply multiplying the polynomials by its denominators in $k[\boldsymbol{u}]$.

To get other components of $\sqrt{I}$ we will look for $g \in (\bigcap_{i \leq s} P_i) \setminus \sqrt{I}$. It is possible to know if a given polynomial g is in $\sqrt{I}$ without actually computing $\sqrt{I}$, by checking whether or not $I : g^\infty$ is the unit ideal.

3. THE ALGORITHM

We now describe the algorithm.

ALGORITHM 3. RADICAL1*(I)*
Input: $I \subset k[\boldsymbol{x}]$
Output: $\sqrt{I}$, *the radical of* I.

1. $\tilde{P} \leftarrow \langle 1 \rangle$.
2. *Repeat*
   (a) *Look for* $g \in \tilde{P} \setminus \sqrt{I}$. *To find it, search over the generators of* $\tilde{P}$ *and check if they are in* $\sqrt{I}$.
   (b) *If there does not exist such* g, *it means that* $\tilde{P} \subset \sqrt{I}$. *Since we always have* $\sqrt{I} \subset \tilde{P}$, *we conclude that* $\tilde{P} = \sqrt{I}$. *Exit the cycle.*
   (c) *If there exists* $g \in \tilde{P} \setminus \sqrt{I}$, *this means that there exists at least one minimal prime* P *associated to* I *such that* $g \notin P$.
   $J \leftarrow I : g^\infty$.
   (d) *Reduction to the zero-dimensional case:*
   Take a maximal independent set $\boldsymbol{u}$ *with respect to* J *and compute the radical of the zero-dimensional ideal* $Jk(\boldsymbol{u})[\boldsymbol{x} \setminus \boldsymbol{u}]$ *(Proposition 1).*
   (e) *Contract* $\sqrt{Jk(\boldsymbol{u})[\boldsymbol{x} \setminus \boldsymbol{u}]}$ *to* $k[\boldsymbol{x}]$.
   (f) $\tilde{P} \leftarrow \tilde{P} \cap (\sqrt{Jk(\boldsymbol{u})[\boldsymbol{x} \setminus \boldsymbol{u}]} \cap k[\boldsymbol{x}])$.
3. *output* $= \tilde{P}$, *the radical of* I.

The correctness of the algorithm is given by the following proposition.

PROPOSITION 4. *Let $I \subset k[\boldsymbol{x}]$ be a proper ideal, let $\mathcal{P}$ be a subset of the minimal primes of I and let $\tilde{P} := \bigcap_{P \in \mathcal{P}} P$ be the intersection of these minimal primes.*
We assume that there exists $g \in \tilde{P} \setminus \sqrt{I}$. If $I : g^\infty = \cap_{i=1}^s Q_i$ is an irredundant primary decomposition and $\boldsymbol{u}$ is a maximal independent set with respect to $I : g^\infty$ then, for all $1 \leq i \leq s$ such that $Q_i \cap k[\boldsymbol{u}] = \{0\}$, $\sqrt{Q_i}$ is a minimal prime of I, and moreover $\sqrt{Q_i} \notin \mathcal{P}$.

PROOF. Let Q_i be a primary component of $I : g^\infty$ such that $Q_i \cap k[\boldsymbol{u}] = \{0\}$. Clearly, Q_i is a primary component associated to I since it is a component of $I : g^\infty$ that satisfies that $g \notin \sqrt{Q_i} = P_i$. In particular $g \in \tilde{P} \setminus P_i$ implies that $P_i \notin \mathcal{P}$.
Since $\boldsymbol{u}$ is independent maximal and $Q_i \cap k[\boldsymbol{u}] = \{0\}$, P_i is a minimal prime of $I : g^\infty$.
If there exists a component Q of I with $\sqrt{Q} \subsetneq P_i$, we would have $g \notin \sqrt{Q}$ and therefore Q would appear in the primary decomposition of $I : g^\infty$, and P_i would not be minimal. Contradiction. □

Remark 3. The algorithm terminates because, in each iteration, we add to $\tilde{P}$ at least one new minimal prime ideal associated to I.

Remark 4. In this algorithm there is no redundancy. All the ideals that we intersect in $\tilde{P}$ are intersection of minimal prime ideals associated to I.

As an example, we apply the algorithm to the ideal

$$I = \langle y + z, x\, z^2 w, x^2 z^2 \rangle \subset \mathbb{Q}[x, y, z, w].$$

In the first iteration, we take $g := 1$ and $J := I : 1^\infty = I$. We find that $\boldsymbol{u} = \{x, w\}$ is a maximal independent set with respect to J. Making the reduction step, we obtain that $\sqrt{J(\boldsymbol{u})[\boldsymbol{x} \setminus \boldsymbol{u}]} \cap k[\boldsymbol{x}] = \langle y, z \rangle$. We take $\tilde{P} := \langle y, z \rangle$.
In the second iteration, we look for $g \in \tilde{P}$ such that $g \notin \sqrt{I}$. We obtain that $z \notin \sqrt{I}$ and compute $J = I : z^\infty = \langle y + z, xw, x^2 \rangle$. Now $\boldsymbol{u} = \{z, w\}$ is a maximal independent set with respect to J. We compute $\sqrt{Jk(\boldsymbol{u})[\boldsymbol{x} \setminus \boldsymbol{u}]} \cap k[\boldsymbol{x}] = \langle y + z, x \rangle$. We take $\tilde{P} := \langle y, z \rangle \cap \langle y + z, x \rangle = \langle y + z, xz \rangle$.
If we search for $g \in \tilde{P}$ such that $g \notin \sqrt{I}$, we obtain that $y + z$ and xz are both in $\sqrt{I}$. Therefore, the algorithm terminates. We obtain that $\sqrt{I} = \langle y + z, xz \rangle$.
We now apply Krick-Logar algorithm to the same ideal, to compare it with ours. We start with $I = \langle y + z, x\, z^2 w, x^2 z^2 \rangle$ and we take the independent set $\boldsymbol{u} = \{x, w\}$. Making the reduction step, we obtain that $\sqrt{I(\boldsymbol{u})[\boldsymbol{x} \setminus \boldsymbol{u}]} \cap k[\boldsymbol{x}] = \langle y, z \rangle$. Up to now, there is no difference with the algorithm we propose.

The next step is different. We look for h such that $\sqrt{I} = (\sqrt{I(\boldsymbol{u})[\boldsymbol{x} \setminus \boldsymbol{u}]} \cap k[\boldsymbol{x}]) \cap \langle I, h\rangle$. We can take $h = xz$. Now, $\sqrt{I} = \langle y, z\rangle \cap \sqrt{\langle I, xz\rangle}$. So it remains to be computed the radical of $\langle I, xz\rangle$. Carrying on the algorithm, we get $\sqrt{\langle I, xz\rangle} = \sqrt{\langle y+z, x\rangle} \cap \sqrt{\langle w, y+z, z^2\rangle} = \langle y+z, x\rangle \cap \langle w, y, z\rangle$.
The last component is redundant, it contains the component $\langle y, z\rangle$ that was already obtained. This redundant component is not an embedded component of I, it is a new component that appeared when we added xz to I.
This is a situation that repeats often in the examples. The polynomials that the algorithm adds to I makes it more and more complex. The polynomials added are usually large, since they are the product of coefficients of polynomials in a Gröbner basis and the size of the Gröbner basis of the new ideal can increase drastically.
This does not happen in our proposed algorithm. We compute instead the saturation with respect to polynomials that are usually simple, and this saturation does not increase the complexity of the ideal since it only takes some components away from it. No new components can appear.

4. COMPLEXITY OF THE ALGORITHM

We shall now compute the theoretical complexity of the algorithm. We remark that we will be analyzing the worst-case-complexity. In the applications, the bounds that we will get are usually not achieved and this is what gives the algorithm practical interest. The modifications to the algorithm that we will introduce in this section (such as random coordinate changes) are only for the purpose of improving the worst-case complexity but are not good in practice.
As presented in the section above, in each step of the algorithm we intersect with $\tilde{P}$ at least one new prime component of $\sqrt{I}$. Therefore, the number of iterations is bounded by the number of prime components of $\sqrt{I}$, which is in time bounded by the number of Bézout, d^n ([13]). Since the degrees of the polynomials in a Gröbner basis is doubly exponential in the number of variables, if we carry out the complexity estimation with the previous algorithm, we would obtain an estimate triply exponential in the number of variables.
To get a better theoretical complexity, we introduce some modifications in the algorithm that will allow us to reduce the *dimension* of the ideal in each iteration and therefore perform at most n iterations. This will lead to a doubly exponential complexity bound. We insist that although this modifications improve the theoretical complexity, in practice they are not efficient, since they destroy the good properties, such as sparsity, that the ideal might have.

Definition 1. We say that an ideal $I \subset k[\boldsymbol{x}]$ of dimension e is in *Noether position* if the set $\boldsymbol{u} = x_1, \ldots, x_e$ is a maximal independent set with respect to I and for each i, $e+1 \leq i \leq n$, there exists a polynomial $p \in I$, in $k[x_1, \ldots, x_e, x_i]$, monic as a polynomial in $k[x_1, \ldots, x_e][x_i]$.

If the ideal I is not in Noether position, we can put it in Noether position by a linear coordinate change. We can use a random coordinate change ([17], Proposition 4.5) or we can do it deterministically with complexity $s^5 d^{O(n^2)}$, where s is the number of polynomials of I and d the maximum degree of the polynomials ([6]).

When the ideal I is in Noether position, we have the following lemma.

LEMMA 5 ([15], LEMA 2.3). *Let I be an ideal of dimension e in Noether position, and*

$$I = (Q_{e_1 1} \cap \ldots Q_{e_1 a_1}) \cap \cdots \cap (Q_{e_t 1} \cap \cdots \cap Q_{e_t a_t})$$

the primary decomposition of I, where $Q_{e_i j}$ are primary ideals of dimension e_i and $0 \leq e_1 < \cdots < e_t = e$. Let $P_{e_i j}$ be the associate primes. Then

$$k[x_1, \ldots, x_e] \cap P_{e_t j} = (0), \quad j = 1, \ldots, a_t.$$

If we take $\boldsymbol{u} := \{x_1, \ldots, x_e\}$, we obtain that $Ik(\boldsymbol{u})[\boldsymbol{x} \setminus \boldsymbol{u}] \cap k[\boldsymbol{x}] = Q_{e_t 1} \cap \cdots \cap Q_{e_t a_t}$.
Therefore, in Step 2c, when we take $J = I : g^\infty$ with $g \in (P_{e_t 1} \cap \cdots \cap P_{e_t a_t}) \setminus \sqrt{I}$, all the primary components of I of dimension e are killed.
To get a good complexity bound we want to kill only the prime components of $\sqrt{I}$ of maximal dimension. We can use a random combination of the polynomials in $\tilde{P}$ as g or we can do it deterministically in the following way.

PROPOSITION 6. *Let I be an ideal of dimension e, as in Lemma 5. Let $J = Q_{e_t 1} \cap \cdots \cap Q_{e_t a_t}$. Then $I : J^\infty$ has dimension at most $e-1$ and $\sqrt{I} = \sqrt{J} \cap \sqrt{I : J^\infty}$.*

Therefore we can bound the number of iterations of the algorithm by e.

Remark 5. The ideal $I : J^\infty$ is not exactly $(Q_{e_1 1} \cap \cdots \cap Q_{e_1 a_1}) \cap \cdots \cap (Q_{e_{t-1} 1} \cap \cdots \cap Q_{e_{t-1} a_{t-1}})$, since some primary components corresponding to immerse primes can also be killed.

The ideal $I : J^\infty$ can be computed in the following way (see Proposition 1.2.6 of [20]):

PROPOSITION 7. *Let I, J be ideals in $k[\boldsymbol{x}]$, with J generated by $f_1, \ldots, f_r$. Let*

$$f := f_1 + t f_2 + \cdots + t^{r-1} f_r \in k[t, \boldsymbol{x}].$$

Then $I : J^\infty = (I : f^\infty) \cap k[\boldsymbol{x}]$.

PROOF. Let $I = Q_1 \cap \cdots \cap Q_s$ be a primary decomposition of I and $P_i = \sqrt{Q_i}$. By Proposition 2, $I : J^\infty = \bigcap_{J \not\subset P_i} Q_i$ and $(I : f^\infty) \cap k[\boldsymbol{x}] = (\bigcap_{f \notin P_i k[t, \boldsymbol{x}]} Q_i k[t, \boldsymbol{x}]) \cap k[\boldsymbol{x}]$. Therefore we need to prove that $J \subset P_i \iff f \in P_i k[t, \boldsymbol{x}]$. If $J \subset P_i$, clearly, $f \in P_i k[t, \boldsymbol{x}]$. For the converse, let $f = a_1 p_1 + \cdots + a_s p_s$, with $p_j \in P_i$ and $a_j \in k[t, \boldsymbol{x}]$. If we replace t by r different values, we obtain that $f_1 + t_j f_2 + \cdots + {t_j}^{r-1} f_r \in P_i$ for $t_1, \ldots, t_r \in k$. We deduce that $f_i \in P_i$ for $i = 1, \ldots, r$, and therefore $J \subset P_i$ as wanted. □

We get the following algorithm.

ALGORITHM 8. RADICAL2*(I)*
Input: $I \subset k[\boldsymbol{x}]$
Output: $\sqrt{I} = P$, *the radical of* I.

1. *Make a linear coordinate change of variables so that I is in Noether position.*
2. *Let $\boldsymbol{u} := \{x_1, \ldots, x_e\}$, with $e = \dim I$. Compute the radical of the zero-dimensional ideal $Ik(\boldsymbol{u})[\boldsymbol{x} \setminus \boldsymbol{u}]$ using Proposition 1.*

3. *Contract* $\sqrt{Ik(\boldsymbol{u})[\boldsymbol{x} \setminus \boldsymbol{u}]}$ *to* $k[\boldsymbol{x}]$.
   $J \leftarrow \sqrt{Ik(\boldsymbol{u})[\boldsymbol{x} \setminus \boldsymbol{u}]} \cap k[\boldsymbol{x}]$
4. *output* $= J \cap$ RADICAL2$(I : J^\infty)$.

Complexity
To estimate the complexity we work over $k = \mathbb{Q}$. We analyze the arithmetic complexity, that is, the number of operations performed in $\mathbb{Q}$. We use the notation $\mathrm{CG}(d, n, s)$, $\mathrm{DG}(d, n)$ and $\mathrm{NG}(d, n, s)$ for the complexity, maximum degree and number of polynomials in a Gröbner basis of an ideal in n variables over $\mathbb{Q}$, generated by s polynomials of maximum degree d. In [10], [16] and [7] they prove bounds for the complexity and the number of polynomials in the general case doubly exponential in the number of variables. The bounds are of order $s^{O(1)}d^{2^{O(n)}}$.
For the maximum degree, the following bound is given in [7]:

$$\deg(g) \leq 2\left(\frac{d^2}{2} + d\right)^{2^{n-1}}.$$

We approximate it by d^{2^n}.
We estimate the complexity of each step of the algorithm, without considering the intersection of the ideals in the last step. We assume that $I \subset \mathbb{Q}[\boldsymbol{x}]$ is an ideal generated by s polynomials of maximum degree d.

1. The Noether position can be achieved by a linear coordinate change. This does not affect the theoretical complexity.
2. To compute the radical $\sqrt{Ik(\boldsymbol{u})[\boldsymbol{x} \setminus \boldsymbol{u}]}$, following Proposition 1, we need to compute at most n Gröbner bases of I. This has complexity at most $ns^{O(1)}d^{2^{O(n)}}$. The n polynomials that appear have degree at most d^{2^n}.
3. The degree of the polynomial h used for the contraction can be bounded by the number of polynomials in the basis times the maximum degree of the polynomials:
   $$s^{O(1)}d^{2^{O(n)}}d^{2^n} = s^{O(1)}d^{2^{O(n)}},$$
   since the degree of the lcm is bounded by the degree of the product of all the polynomials.
   Now, the complexity of the contraction is the complexity of the computation of the Gröbner basis of $\langle I, th - 1\rangle$:
   $$\mathrm{CG}(s^{O(1)}d^{2^{O(n)}}, n+1, s^{O(1)}d^{2^{O(n)}}) =$$
   $$(s^{O(1)}d^{2^{O(n)}})^{O(1)}(s^{O(1)}d^{2^{O(n)}})^{2^{O(n)}} = (sd)^{2^{O(n)}}.$$
   The number of polynomials in J and their degrees can also be approximated by $(sd)^{2^{O(n)}}$.
4. To compute $I : J^\infty$, by Proposition 7 and Remark 1, we need to compute a Gröbner basis of $\langle I, tf - 1\rangle$. The degree of f is bounded by $(sd)^{2^{O(n)}} + d^{2^{O(n)}} = (sd)^{2^{O(n)}}$. This has complexity
   $$CG((sd)^{2^{O(n)}}, n+1, (sd)^{2^{O(n)}}) = (sd)^{2^{O(n)}}.$$
   The number of polynomials and the maximum degree can also be approximated by $(sd)^{2^{O(n)}}$.

We can estimate the complexity of the whole call by $(sd)^{2^{O(n)}} = (sd)^{2^{cn}}$ for some universal constant c.
In each call, the dimension of the ideal considered decreases. Therefore we need at most n calls, since the dimension cannot be greater than the number of variables.
In the second call we start with $(sd)^{2^{cn}}$ polynomials of degree $(sd)^{2^{cn}}$. The complexity of this call is

$$((sd)^{2^{cn}}, n, (sd)^{2^{cn}}) = ((sd)^{2\,2^{cn}})^{2^{cn}} = (sd)^{2^{2cn+1}}.$$

The same bounds are valid for the number of polynomials and their degrees.
Therefore, after n calls we get the bound

$$(sd)^{2^{n(cn)+n-1}} = (sd)^{2^{O(n^2)}},$$

for the complexity, the number of polynomials and their degrees in the last call.
Finally, to compute the intersection of the outputs in each call, we use that $I_1 \cap I_2 = \langle I_1 \cdot t, I_2 \cdot (1-t)\rangle \cap k[\boldsymbol{x}]$, which can be done by a Gröbner basis computation. This does not modify the estimates obtained.
We have shown that the theoretical complexity of the algorithm is doubly exponential in the number of variables.

5. PERFORMANCE EVALUATION

In this section, we apply the proposed algorithm to several examples given in [4] and [1] and evaluate its performance. (We only consider those ideals that are not zero dimensional.) We implemented the algorithm in Singular ([12]). Our routine uses the subroutine for the reduction to the zero dimensional case that is already implemented in the library `primdec` [5] for the computation of the radical by Krick-Logar-Kemper algorithm. We compare the times obtained by our algorithm with the algorithms implemented in `primdec`: Krick-Logar-Kemper ([15], [14]) and Eisenbud-Huneke-Vasconcelos ([8]).
The results are shown in Table 1. All the computations are done over $\mathbb{Q}$. The ordering of the monomials is always the degree reverse lexicographical ordering with the underlying ordering of the alphabet.
The codes for the examples in the first column are the ones given in [4] and [1]. The second column indicates the dimension of the ideal, the third column the total number of primary components and the fourth column the number of primary components corresponding to embedded primes. Timing is measured in hundredth of seconds. The entry * means that after one day of computations, the algorithm did not terminate.
In the implementation of KLK in Singular, the original ideal is first decomposed using factorizing Gröbner bases algorithm and then the radical of each component is computed. We do the same decomposition in our algorithm.
We see that for time consuming computations, our proposed algorithm is always faster. We explain briefly the differences that appear.
In example DGP-29, both KLK and our algorithm obtain the radical in the first step. Because of the structure of them, our algorithm stops after that step, but KLK algorithm goes on computing redundant components. In examples DGP-16, CCT-83 and CCT-C, after the first step, the saturations

Table 1: Timing results

Code	D.	Prim. comps	Emb. comps	EHV	KLK	this paper
DGP-1	3	4	0	*	104	90
DGP-2	3	16	1	*	86	158
DGP-3	2	11	7	240	8	13
DGP-4	6	4	1	53	23	21
DGP-5	3	9	2	*	4271	627
DGP-6	3	3	0	*	158	185
DGP-7	3	6	0	*	45	153
DGP-9	1	12	0	11	*	229
DGP-12	1	25	0	329	5597	247
DGP-14	1	8	6	5	7	10
DGP-16	8	4	0	*	3214	3402
DGP-20	4	2	1	589	74	38
DGP-21	9	9	8	4	39	13
DGP-22	2	9	2	*	63	84
DGP-23	2	18	6	*	111	157
DGP-24	8	6	1	*	14	29
DGP-25	5	7	2	*	225	273
DGP-27	4	3	0	199	5	9
DGP-28	7	2	0	2380	46	56
DGP-29	2	12	11	*	61714	3598
DGP-30	1	14	0	*	132	163
DGP-31	1	1	0	1	6	8
DGP-32	2	17	9	25814	66	265
DGP-33	2	3	0	2	11	16
CCT-M	5	3	0	*	119	129
CCT-83	5	3	0	*	*	250
CCT-C	5	4	0	*	*	326
CCT-O	2	5	0	1	217	29

computed by our algorithm are simple and the algorithm terminates quickly, while in KLK algorithm, the polynomials added are large, and the resulting Gröbner bases are huge and impossible to handle.

6. ACKNOWLEDGEMENTS

The author thanks Teresa Krick for her guidance in this work, Gabriela Jeronimo for her valuable comments and corrections and the referees of the previous version of this paper for their extremely useful suggestions.

7. REFERENCES

[1] M. Caboara, P. Conti, and C. Traverso. Yet another algorithm for ideal decomposition. *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes*, (12):39–54, 1997.

[2] D. Cox, J. Little, and D. O'Shea. *Ideals, Varieties and Algorithms*. Springer, 1996.

[3] D. Cox, J. Little, and D. O'Shea. *Using Algebraic Geometry*. Springer, 1998.

[4] W. Decker, G.-M. Gruel, and G. Pfister. Primary decomposition: Algorithms and comparisons. *Algorithmic algebra and number theory, Springer Verlag, Heidelberg*, pages 187–220, 1998.

[5] W. Decker, G. Pfister, and H. Schoenemann. `primdec.lib`. A SINGULAR 3.0 library for computing primary decomposition and radical of ideals, 2005.

[6] A. Dickenstein, N. Fitchas, M. Giusti, and C. Sessa. The membership problem for unmixed polynomial ideals is solvable in single exponential time. *Discrete Applied Mathematics*, (33):73–94, 1991.

[7] T. W. Dube. The structure of polynomial ideals and grobner bases. *SIAM J. Comput.*, (19):750–773, 1990.

[8] D. Eisenbud, C. Huneke, and W. Vasconcelos. Direct methods for primary decomposition. *Invent. Math.*, (110):207–235, 1992.

[9] P. Gianni, B. Trager, and G. Zacharias. Bases and primary decomposition of ideals. *J. Symbolic Computation*, (6):149–167, 1988.

[10] M. Giusti. Some effective problems in polynomial ideal theory. *EUROSAM 84, Lecture Notes in Computer Science*, (174):159–171, 1984.

[11] G.-M. Greuel and G. Pfister. *A Singular Introduction to Commutative Algebra*. Springer, 2002.

[12] G.-M. Greuel, G. Pfister, and H. Schonemann. SINGULAR 3.0.1. A Computer Algebra System for Polynomial Computations, Centre for Computer Algebra, University of Kaiserslautern, 2005. `http://www.singular.uni-kl.de`.

[13] J. Heintz. Definability and fast quantifier elimination over algebraically closed fields. *Theor. Comp. Science*, (24):239–278, 1983.

[14] G. Kemper. The calculation of radical ideals in positive characteristic. *J. Symbolic Computation*, (34):229–238, 2002.

[15] T. Krick and A. Logar. An algorithm for the computation of the radical of an ideal in the ring of polynomials. *AAECC9, Springer LNCS*, (539):195–205, 1991.

[16] T. Krick and A. Logar. Membership problem, representation problem and the computation of the radical for one-dimensional ideals. *Progress in Mathematics*, (94):203–216, 1991.

[17] T. Krick, L. M. Pardo, and M. Sombra. Sharp estimates for the arithmetic nullstellensatz. *Duke Math J.*, (109):521–598, 2001.

[18] R. Matsumoto. Computing the radical of an ideal in positive characteristic. *J. Symbolic Computation*, (32):263–271, 2001.

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

[20] W. Vasconcelos. *Computational Methods in Commutative Algebra and Algebraic Geometry*. Springer-Verlag, 1998.