PARDI !

François Boulier, François Lemaire and Marc Moreno Maza

Université Lille 1, LIFL, 59655 Villeneuve d'Ascq France
{boulier, lemaire, moreno}@lifl.fr

Abstract

We propose a new algorithm for converting a characteristic set of a prime differential ideal from one ranking into another. It identifies the purely algebraic subproblems which arise during differential computations and solves them algebraically. There are improvements w.r.t. other approaches. Formerly unsolved problems are carried out. It is conceptually very simple. Different variants are implemented.

differential algebra. PDE. characteristic sets. change of rankings. gcd.

Introduction

In this paper, we propose an algorithm which solves the following problem: given a characteristic set C of a prime differential ideal $\mathfrak{p}$ w.r.t some ranking $\mathcal{R}$ and another ranking $\overline{\mathcal{R}} \neq \mathcal{R}$, compute a characteristic set $\overline{C}$ of $\mathfrak{p}$ w.r.t. $\overline{\mathcal{R}}$.

The algorithm that we present, called* PARDI applies for systems of partial differential polynomial equations. It specializes to systems of ordinary differential polynomial equations and is then called† PODI. It specializes to nondifferential polynomial equations where it is called‡ PALGIE.

Consider the following three partial differential polynomials. There are two differential indeterminates u and v (which can be viewed as two unknown functions of two independent variables x and y) and two derivations $\partial/\partial x$ and $\partial/\partial y$.

$$u_x^2 - 4u, \quad u_{xy}v_y - u + 1, \quad v_{xx} - u_x.$$

The differential ideal $\mathfrak{p}$ generated by these differential polynomials is prime. With

*PARDI is an acronym for Prime pARtial Differential Ideal. In French, "*pardi*" is an old-fashioned swearword such as, say, "*egad*" in English.

†PODI is an acronym for Prime Ordinary Differential Ideal.

‡PALGIE is an acronym for Prime ALGebraic IdEal. However, since "*algie*" means "suffering" in French, one might also understand PALGIE as "polynomial suffering" say.

respect to the following ordering (ranking) $\mathcal{R}$ on the derivatives of u and v

$$\cdots > v_{xx} > v_{xy} > v_{yy} > u_{xx} > u_{xy} > u_{yy} > v_x > v_y > u_x > u_y > v > u$$

the differential ideal $\mathfrak{p}$ admits the following set C for characteristic set

$$v_{xx} - u_x, \quad 4v_y u + u_x u_y - u_x u_y u, \quad u_x^2 - 4u, \quad u_y^2 - 2u.$$

With respect to the following elimination ranking $\overline{\mathcal{R}}$,

$$\cdots > u_x > u_y > u > \cdots > v_{xx} > v_{xy} > v_{yy} > v_x > v_y > v$$

it admits the following set $\overline{C}$ for characteristic set

$$v_{yy}^4 - 2v_{yy}^2 - 2v_y^2 + 1, \quad v_{xy}v_y - v_{yy}^3 + v_{yy}, \quad v_{xx} - 2v_{yy}, \quad u - v_{yy}^2.$$

The PARDI algorithm is able to compute $\overline{C}$ from C, $\mathcal{R}$ and $\overline{\mathcal{R}}$ or C from $\overline{C}$, $\overline{\mathcal{R}}$ and $\mathcal{R}$. The computations are immediate for this system and take 60 kilobytes in the C programming language on a SUN ULTRA 5.

Euler's equations for perfect fluids write

$$\vec{v} + (\vec{v} \cdot \vec{\nabla})\,\vec{v} + \vec{\nabla}p = \vec{0}, \quad \vec{\nabla}\vec{v} = 0.$$

In two dimensions, denoting $\vec{v} = (v^1, v^2)$ and $\vec{\nabla} = (\partial/\partial x, \partial/\partial y)$, we get three differential polynomial equations

$$v_t^1 + v^1 v_x^1 + v^2 v_y^1 + p_x = 0, \quad v_t^2 + v^1 v_x^2 + v^2 v_y^2 + p_y = 0, \quad v_x^1 + v_y^2 = 0.$$

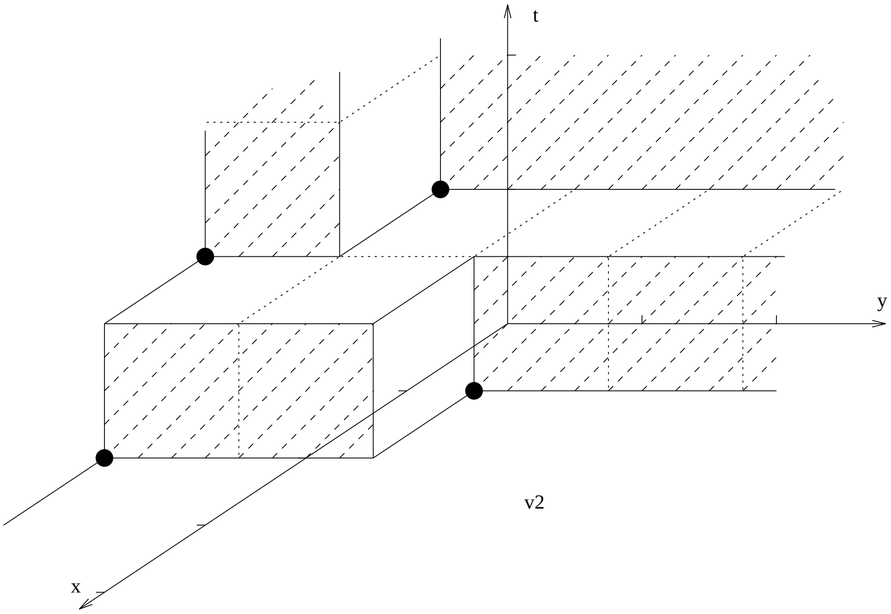
The differential polynomials which appear on the lefthand sides of the equations generate a prime differential ideal $\mathfrak{p}$. There are three differential indeterminates v^1, v^2 (components of the speed) and the pressure p. They depend on three independent variables x, y (space variables) and the time t. For some orderly ranking, the general simplifier Rosenfeld–Gröbner provides with nearly no computation the characteristic set C of $\mathfrak{p}$

$$p_{xx} + 2\,v_x^2\,v_y^1 + 2\,(v_y^2)^2 + p_{yy}, \quad v_t^1 + v^2\,v_y^1 + p_x - v_y^2\,v^1, \quad v_x^1 + v_y^2, \quad v_t^2 + v^1\,v_x^2 + v^2\,v_y^2 + p_y.$$

For some elimination ranking $(p, v^1) \gg \mathrm{degrevlex}(v^2)$ with $t > x > y$ the implementation of PARDI was able to compute a characteristic set $\overline{C}$ of $\mathfrak{p}$. This characteristic set cannot be written in this paper (the computer file is 600 kilobytes large). It is the first time that the computation of this characteristic set succeeds. There are 7 equations involving more than 50 different derivatives. Intermediate computations, performed with the C implementation, took a bit more than 100 megabytes and a quarter of an hour on a SUN ULTRA 5. We have

$$\mathrm{rank}\overline{C} = \{p_x,\, p_y,\, v^1,\, v_{xxxxt}^2,\, v_{xxxtt}^2,\, v_{xxytt}^2,\, v_{xxxyyt}^2\}.$$

The diagram of the differential indeterminate v^2 is[§]



As far as we know, Ollivier was the first to solve the problem addressed in this paper. Let's quote [Ollivier, 1990, page 95]: "one can [design] a method for constructing a characteristic set of a finitely generated prime differential ideal as soon as one can effectively test membership to this ideal". An algorithm is given in SCRATCHPAD in [Ollivier, 1990, page 97]. In most approaches, a known characteristic set provides the membership test algorithm. This functionality was afterwards implemented in the MAPLE *diffalg* package by the first author. The implemented algorithm handles differential ideals given by characteristic sets which do not need to be prime. Such a problem was also considered in Boulier [1999]. However, the algorithms presented in Boulier [1999] compute differential polynomials which are not necessarily part of the desired characteristic set but only help computing it. They are complementary to PARDI. The problem was also addressed by [Bouziane et al., 2001, section 3.2]. Their algorithm does not make use of the primality hypothesis. It computes a representation of the prime differential ideal as an intersection of differential ideals presented by characteristic sets. The desired characteristic set can then easily be picked from these latter (by a dimension argument). Their algorithm relies on a test of algebraic invertibility modulo triangular systems (so ours does) but they perform it by means of Gröbner bases computations.

The restriction to prime ideals is realistic. Indeed most differential systems coming from real problems generate differential prime ideals. Quite often, non-differential polynomial systems in positive dimension either generate prime ideals or can be decomposed into prime ideals.

Assuming that prime ideals are given by characteristic sets is realistic too. In the differential case, it happens quite often (e.g. dynamical systems in nonlinear

[§]The authors would like to thank Marc Giusti for his help.

control theory) that the input equations already form characteristic sets w.r.t. some rankings.

The algorithm we propose generalizes to ideals which are not necessarily prime. However, for the reasons explained above and the legibility of the paper, we prefer to restrict ourselves to the prime case.

Our algorithm easily extends to perform invertible changes of coordinates on the dependent and independent variables. Such maps realize ring isomorphisms between two differential polynomial rings $\phi : R \to \overline{R}$, and one–to–one correspondences between the differential ideals of R and the ones of $\overline{R}$. However the image $\overline{C}$ of a characteristic set C of $\mathfrak{p}$ is usually not a characteristic set of the ideal $\overline{\mathfrak{p}} = \phi\mathfrak{p}$ and there is usually no ranking w.r.t. which a characteristic set of $\overline{\mathfrak{p}}$ could be easily deduced from $\overline{C}$. The idea is then to apply PARDI over $\overline{C}$ but to test membership in $\overline{\mathfrak{p}}$ by performing the inverse changes of coordinates and testing membership in $\mathfrak{p}$ using C.

Our approach offers several advantages. It identifies the algebraic subproblems which occur in the differential computations and solves them by a purely algebraic method. This improves the control of the coefficients growth and avoids many useless computations only due to differential considerations. This very important advantage w.r.t. all other approaches permits us to handle some unsolved problems. The three variants were implemented: PARDI in MAPLE and C, PODI in C and PALGIE in MAPLE, C and Aldor. The application to the change of variables was implemented in MAPLE.

A last contribution (but not the least one) is the conceptual simplicity of our algorithm, which contrasts with the high technicity of its implementation. Everybody knows that the common roots of two univariate polynomials over a field are given by their gcd. Our algorithm applies this very simple idea and replaces any two univariate polynomials by one of their gcd over the fraction field of some quotient ring. This makes much more sense than speaking of full remainders as in the previous approaches. Some methods for computing triangular decompositions of arbitrary ideals (prime or not) are also explicitly formulated in terms of gcd: Kalkbrener [1993], Lazard [1991], Moreno Maza [2000]. The use of the gcd made by these methods is however more complicated than that made by PARDI. Indeed in these methods the ideal modulo which the gcd computations are performed has to change during the triangular decomposition, since it depends on the equations already processed. This is not the case in our particular context. Hence we wish that the simplicity of our approach helps in popularizing all triangular decomposition methods.

A preliminary version of this paper was published by Boulier et al. [2001a]. This paper contains two new results: a new subalgorithm called regalise which supersedes specialized_regCharacteristic but does not always apply for PDE systems and a new efficient criterion to avoid critical pairs in the completion process. The version of PARDI is also more efficient than that of Boulier et al. [2001a] since the set A of the processed equations is maintained as a regular chain. Last, proofs are given which were omitted by Boulier et al. [2001a].

1. General definitions and notations

1.1. Computer science

Definition: A while loop invariant is a property which holds each time the loop condition is evaluated.

Loop invariants are very important for they permit to prove the correctness of algorithms: they hold in particular when the loop condition evaluates to false i.e. when the loop terminates. Combined to the negation of the loop condition, they give the properties of the datas computed by the loop.

1.2. Commutative algebra

Let X be an ordered alphabet (possibly infinite).

Let $R = K[X]$ be a polynomial ring where K is a field. Let $p \in R \setminus K$ be a polynomial. If $x \in X$ is any indeterminate then the *leading coefficient* of p viewed as a univariate polynomial in x (with coefficients in the ring $K[X \setminus \{x\}]$) is denoted $\mathsf{lcoeff}(p,\, x)$. If $\deg(p,\, x) = 0$ then $\mathsf{lcoeff}(p,\, x) = p$. The *leader* of p, denoted $\operatorname{ld} p$, is the greatest indeterminate x which occurs in p. The polynomial p can be written as $p = a_d\, x^d + \cdots + a_1\, x + a_0$ where $d = \deg(p, x)$ and the polynomials a_i are free of x. The polynomial $i_p = a_d$ is the *initial* of p (the initial of p is the leading coefficient of p w.r.t. its leader). The *rank* of p is the monomial x^d. The *reductum* of p is the polynomial $p - i_p\, x^d$. If x^d and y^e are two ranks then $x^d < y^e$ if $x < y$ or $x = y$ and $d < e$. The *separant* of p is the polynomial $s_p = \partial p / \partial x$.

Let $A \subset R \setminus K$ be a set of polynomials. Then I_A (resp. S_A) denotes the set of the initials (resp. the separants) of its elements. We denote $H_A = I_A \cup S_A$. The set A is said to be *triangular* if its elements have distinct leaders.

Let q be a polynomial. We denote $\mathsf{pquo}(q, p, x)$ and $\mathsf{prem}(q, p, x)$ the pseudoquotient and the pseudoremainder [Knuth, 1966, volume 2, page 407] of q by p, viewed as univariate polynomials in x. If x is omitted, both polynomials are viewed as univariate polynomials in the leader of p. We denote $\mathsf{prem}(q, A)$ "*the*" pseudoremainder r of q by all the elements of A i.e. any polynomial r obtained from q and the elements of A by performing successive pseudoreductions and such that $\mathsf{prem}(r, p) = r$ for every $p \in A$. Without further precisions, r is not uniquely defined. Fix any precise algorithm. By convention, we define $\mathsf{prem}(q, \emptyset) = q$.

If A is a subset of a ring R then (A) denotes the ideal generated by A. By convention, we define $(A) = (0)$ when A is empty. Let $\mathfrak{a}$ be an ideal of R. If $S = \{s_1, \ldots, s_t\}$ then the *saturation* $\mathfrak{a} : S^\infty$ of $\mathfrak{a}$ by S is the ideal $\mathfrak{a} : S^\infty = \{p \in R \mid \exists a_1, \ldots, a_t \in \mathbb{N} \text{ such that } s_1^{a_1} \cdots s_t^{a_t}\, p \in \mathfrak{a}\}$. By convention, we define $\mathfrak{a} : S^\infty = \mathfrak{a}$ if S is empty.

1.3. Regular chains

A *regular* element of a ring R_0 is by definition a non zerodivisor. An element $a \in R_0$ is said to be *invertible* if there exists some $\overline{a} \in R_0$ such that $a\,\overline{a} = 1$. Invertible implies regular.

In this section, we consider a triangular set $A = \{p_1, \ldots, p_n\}$ of a polynomial ring R. Renaming the indeterminates if needed, we may assume that $R = K[t_1, \ldots, t_m, x_1, \ldots, x_n]$ and that $\mathrm{ld}\, p_i = x_1$ for each $1 \leq i \leq n$.

Regular chains are defined in Aubry et al. [1999]. See also Kalkbrener [1993] and Lazard [1991]. We take for definition the following caracterisation [Boulier et al., 2001b, theorem 3]:

Definition: Let $1 \leq \ell \leq n$ be an index. Denoting i_k the initial of p_k we define

$$Z_\ell = K(t_1, \ldots, t_m)[x_1, \ldots, x_\ell] / \left((p_1, \ldots, p_\ell) : (i_1 \cdots i_\ell)^\infty\right).$$

Definition: The triangular set A is a *regular chain* if the initial i_ℓ of p_ℓ is invertible in $Z_{\ell-1}$ for each $2 \leq \ell \leq n$.

Definition: A regular chain A is *squarefree* if the separant s_ℓ of p_ℓ is invertible in Z_ℓ for $1 \leq \ell \leq n$.

A regular chain $A = \{p_1, \ldots, p_n\}$ is a characteristic set of the ideal $(A) : I_A^\infty$ in the rings $K(t_1, \ldots, t_m)[x_1, \ldots, x_n]$ and $K[t_1, \ldots, t_m, x_1, \ldots, x_n]$ by [Aubry et al., 1999, theorem 6.1] and, more precisely, by [Aubry, 1999, théorème 4.6.1]. A squarefree¶ regular chain A is a characteristic set of the ideal $(A) : H_A^\infty$ in the same rings by [Boulier et al., 2001b, theorem 4].

Observe that these properties still hold if we enlarge the t's with some extra indeterminates which do not occur in A. They even hold if the set of the t's is infinite.

1.3.1. Checking invertibility

PROPOSITION 1.1: *The following function* is_regular *takes two parameters: a nonzero element p of Z_n and a regular chain A. It returns a pair (b, g) where b is a boolean and $g \in Z_n$. If b is* true *then p is invertible in Z_n. If b is* false *then g satisfies the following properties:*

1. *$g \notin K(t_1, \ldots, t_m)$ hence, for some $1 \leq \ell \leq n$, g has x_ℓ for leader ;*
2. *the initial of g is invertible in $Z_{\ell-1}$;*
3. *g is a nontrivial divisor of p_ℓ in $Z_{\ell-1}[x_\ell]$.*

Observe that the component g of the pair does not matter in the case b is true. We write a dot for it in the code below.

¶there is a strong analogy with the traditional meaning of "*squarefree*". In particular, the ideal $(A) : I_A^\infty$ is radical, it has no multiple zeros in the algebraic closure of $K(t_1, \ldots, t_m)$ and is equal to $(A) : H_A^\infty$.

```
function is_regular(p, {p_1, ..., p_n})
begin
  if p ∈ K(t_1, ..., t_m) then
    return (true, ·)
  else
    let x_ℓ be the leader of p
    let i_p be the initial of p
    let k be the smallest index such that i_p ∈ K(t_1, ..., t_m)[x_1, ..., x_k]
    (b, g) := is_regular(i_p, {p_1, ..., p_k})
    if b then
      (b', g') := Euclidean_algorithm(p, p_ℓ, x_ℓ, {p_1, ..., p_{ℓ-1}})
      if b' and deg(g', x_ℓ) = 0 then
        return (true, ·)
      else
        return (false, g')
      fi
    else
      return (false, g)
    fi
  fi
end
```

The proof of the above proposition cannot be given independently of that of the following one.

Proposition 1.2: *The following function* Euclidean_algorithm *takes four parameters: two polynomials a, $b \in Z_n[x]$ with initials invertible in Z_n, their leaders x and a regular chain A. It returns a pair $(b,\, g)$ where b is a boolean and $g \in Z_n[x]$. If b is true then g satisfies the following properties:*

1. *$g \in (a,\, b)$ in $Z_n[x]$*
2. *g is a common divisor of a and b in $Z_n[x]$*
3. *the leading coefficient of g w.r.t. x is invertible in Z_n.*

If b is false then g satisfies the properties already stated in the above proposition:

1. *$g \notin K(t_1, \ldots, t_m)$ hence, for some $1 \le \ell \le n$, g has x_ℓ for leader ;*
2. *the initial of g is invertible in $Z_{\ell-1}$;*
3. *g is a nontrivial divisor of p_ℓ in $Z_{\ell-1}[x_\ell]$.*

```
function Euclidean_algorithm(a, b, x, {p_1, ..., p_n})
begin
  p := a
  q := b
  while q ≠ 0 do
```

```
    r := prem(p, q, x)
    while r ≠ 0 and lcoeff(r, x) = 0 in Z_n do
      r := reductum(r, x) (i.e. r − lcoeff(r, x) x^deg(r,x))
    od
    if r ≠ 0 then
      let k be the smallest index such that lcoeff(r, x) ∈ K(t_1, ..., t_m)[x_1, ..., x_k]
      (bool, h) := is_regular(lcoeff(r, x), {p_1, ..., p_k})
      if bool is false then
        return (bool, h)
      fi
    fi
    p := q
    q := r
  od
  return (true, p)
end
```

The two functions above make the definitions of regular chains and of squarefree regular chains algorithmic.

The function Euclidean_algorithm actually tries to compute a greatest common divisor of a and b as defined in Moreno Maza and Rioboo [1995].

A variant of the above functions closer to a true implementation was written by Moreno Maza [2000]. The pseudoremainder sequence algorithm of Ducos [2000] is used instead of the basic scheme presented above.

Proposition 1.3: *Functions* is_regular *and* Euclidean_algorithm *terminate.*

Proof: We prove the proposition by induction on n.

Basis of the induction: the case $n = 0$. The function is_regular terminates for $p \in K(t_1, \ldots, t_m)$. The function Euclidean_algorithm terminates for it degenerates to the usual Euclidean algorithm between univariate polynomials over a field (apart perhaps at the first turn, the degree of q strictly decreases).

The general case: the case $n > 0$. We assume inductively that is_regular and Euclidean_algorithm terminate. Each time is_regular calls itself or the function Euclidean_algorithm, the number of elements of the regular chain is decreasing. Each time Euclidean_algorithm calls is_regular, the number of elements of the regular chain is decreasing. Using the induction hypothesis, recursive calls terminate. The loop of Euclidean_algorithm then always terminates for, apart perhaps at the first turn, the degree of q in x strictly decreases. □

Proposition 1.4: *The given pseudocodes satisfy the specifications stated in proposition 1.1 and 1.2.*

Proof: We prove the proposition by induction on n.

Basis of the induction: the case $n = 0$. Every nonzero element of a field is invertible. Thus is_regular always succeeds and proposition 1.1 is satisfied. The function Euclidean_algorithm and its specifications degenerate to that of the usual Euclidean algorithm between polynomials over a field. Thus proposition 1.2 is satisfied.

The general case: $n > 0$. We assume inductively that results of recursive calls satisfy both propositions.

First consider is_regular. If the first call to is_regular returns a pair (false, g) then, using the induction hypothesis, nothing more needs to be done and the pair is returned.

If the call to Euclidean_algorithm returns a pair (false, g') then, using the induction hypothesis, nothing more needs to be done and the pair is returned.

If the call to function Euclidean_algorithm returns a pair (true, g') such that $\deg(g', x_\ell) > 0$ then a nontrivial factorisation of p_ℓ is exhibited and, using the induction hypothesis, the pair (false, g') is returned.

If the call to function Euclidean_algorithm returns a pair (true, g') such that $\deg(g', x_\ell) = 0$ then, using the induction hypothesis, g' is regular in $Z_{\ell-1}$ and there exists $\lambda,\, \mu \in Z_{\ell-1}[x_\ell]$ such that $\lambda\, p + \mu\, p_\ell = 1$ in $Z_{\ell-1}[x_\ell]$. Since $p_\ell = 0$ in Z_n we have $\lambda\, p = 1$ in Z_n and p is invertible in Z_n. The pair $(\mathit{true}, \cdot)$ may be returned.

Consider Euclidean_algorithm now. This function is the usual Euclidean algorithm between polynomials. The only difference is that coefficients do not lie in a field. Relying on is_regular, it explicitly contains code which ensures that the leading coefficient of the polynomial by which divisions are performed is nonzero and invertible. As soon as some leading coefficient cannot be proven invertible, the function gives up and returns the exhibited factorisation of some element of A.

Thus the correction proof is very close to that of the usual Euclidean algorithm. The core of it is the following classical argument: if $a,\, b,\, r,\, g \in Z_n[x]$ have leading coefficients w.r.t. x invertible in Z_n and are such such that $r = \mathsf{prem}(a,\, b,\, x)$ then g is a common divisor of a, b if and only if g is a common divisor of b, r. The fact that, in the case of a success, the polynomial p returned by the function lies in the ideal $(a,\, b)$ can be proven, very classically, by considering the extended version of the extended Euclidean algorithm. □

1.3.2. Saturating ideals

PROPOSITION 1.5: *Let A be a squarefree regular chain. Let p be a polynomial such that* is_regular (p, A) *returns* (false, g). *Denote x_ℓ the leader of g and $h = \mathsf{pquo}(p_\ell, g)$. Then the set A_g (resp. A_h) obtained from A by replacing p_ℓ by g (resp. by h) has the same sets of leaders as A, forms squarefree regular chains. These sets satisfy*

$$(A) : H_A^\infty = (A_g) : H_{A_g}^\infty \cap (A_h) : H_{A_h}^\infty.$$

Proof: See [Boulier et al., 2001b, theorem 5]. □

The above proposition provides an algorithm for saturating an ideal presented by a squarefree regular chain by the multiplicative family generated by a polynomial p.

PROPOSITION 1.6: *Let A be a squarefree regular chain. Assume that $(A):H_A^\infty \subset \mathfrak{p}$ where $\mathfrak{p}$ is some prime ideal and that membership testing in $\mathfrak{p}$ is algorithmic. Let $p \notin \mathfrak{p}$ be a polynomial. Then the following pseudocode computes a squarefree regular chain $\overline{A}$ having the same set of leaders as A and such that*

$$(A) : H_A^\infty \subset (A) : (H_A \cup \{p\})^\infty \subset (\overline{A}) : H_{\overline{A}}^\infty \subset \mathfrak{p}. \quad (1)$$

$\overline{A} := A$
$(b,\, g) :=$ is_regular$(p, \overline{A})$
while b is *false* do
  let x_ℓ be the leader of g
  if $g \notin \mathfrak{p}$ then
    Replace p_ℓ by pquo(p_ℓ, g) in $\overline{A}$
  else
    Replace p_ℓ by g in $\overline{A}$
  fi
  $(b,\, g) :=$ is_regular$(p, \overline{A})$
od

Proof: The pseudocode terminates for the degree of the ℓth element of $\overline{A}$ strictly decreases each time the loop body is performed.

We first claim that the relation (2) is a an invariant of the above loop.

$$(A) : H_A^\infty \subset (\overline{A}) : H_{\overline{A}}^\infty \subset \mathfrak{p}. \quad (2)$$

This relation is satisfied initially.

It is sufficient to prove that relation (2) holds after the first turn. We have $(A) : H_A^\infty \subset \mathfrak{p}$. Assume is_regular returns *false*. Denote h the pseudoquotient, we have by proposition 1.5

$$(A) : H_A^\infty = (A_g) : H_{A_g}^\infty \cap (A_h) : H_{A_h}^\infty \subset \mathfrak{p}.$$

Moreover, p_ℓ, g, h have the same leader x_ℓ and we have a relation $c\,p_\ell = g\,h \mod \mathfrak{p}$ where c is a power of the initial of g. The polynomial c and the initial of p_ℓ are regular modulo $\mathfrak{p}$ thus so are the initials of g and h.

Therefore if $g \in \mathfrak{p}$ then $\overline{A} = A_g$ is a squarefree regular chain and we have $(\overline{A}) : H_{\overline{A}}^\infty \subset \mathfrak{p}$.

AAAAAAAAAAAAAAAA

If $g \notin \mathfrak{p}$ we have $h \in \mathfrak{p}$ for $\mathfrak{p}$ is prime. In that case $\overline{A} = A_h$ is also a squarefree regular chain and we also have $(\overline{A}) : H_{\overline{A}}^\infty \subset \mathfrak{p}$. The claim is proven.

Putting the claim just proven in an inductive argument, we see that relation (2) holds after finitely many executions of the loop body.

The inclusion $(A){:}H_A^\infty \subset (A){:}(H_A \cup \{p\})^\infty$ is trivial. When the loop terminates, p is regular modulo $(\overline{A}) : H_{\overline{A}}^\infty$ thus $(\overline{A}) : H_{\overline{A}}^\infty = (\overline{A}) : (H_{\overline{A}} \cup \{p\})^\infty$. Relation (2) implies that $(A){:}(H_A \cup \{p\})^\infty \subset (\overline{A}){:}(H_{\overline{A}} \cup \{p\})^\infty$. Putting these three arguments together we see relation (1) holds after termination of the above pseudocode. The proposition is proven. $\square$

We are going to encounter the above pseudocode many different times. Observe that if g and p have the same leader then $g \notin \mathfrak{p}$ (quick membership test). Observe that we do not have in general $(A){:}(H_A \cup \{p\})^\infty = (\overline{A}){:}H_{\overline{A}}^\infty$ though this is quite often true since is_regular checks the regularity of polynomials different from p. Any of these tests may fail. For the same reason, is_regular may fail even if p is not a zero divisor.

1.3.3. Variants of **I2**

Invariant **I2** implies that the set of ranks of the elements of A is autoreduced. This is an old and weaker form of that invariant, used in Boulier et al. [2001a]. The following **I2'** is a stronger version of **I2** which presents some advantages.

I2' the set A is squarefree, autoreduced and strongly normalized ; its elements are primitive over $K[t_1, \ldots, t_m]$.

Strongly normalized means that the initials of the elements of A lie in the ring $K[t_1, \ldots, t_m]$ (whence A is a regular chain). See Boulier and Lemaire [2000].

Algorithmically, invariant **I2'** can be achieved with a variant of is_regular based on an extended variant of Euclidean_algorithm. All computations can be performed with no fractions. The use of such an algebraic inverse computation algorithm can be costly. Invariant **I2'** is algorithmically interesting anyway when one computes the content of a differential polynomial p w.r.t. its leader (say) x_ℓ (this operation is very important in practice). The content of a polynomial is the gcd of its coefficients. This gcd should be computed "modulo the equations". Since the word gcd is not really defined in that setting, let's say roughly that we would like to catch as many common factors of the coefficients (modulo the equations), as possible. Strongly normalizing a polynomial seems to have the effect of making more of these common factors become plain common factors (i.e. common factors but not modulo the equations). And there are much more algorithms to compute a plain multivariate gcd than to compute a gcd modulo a set of equations. A reference book for plain multivariate gcd algorithms is Geddes et al. [1992].

1.4. Differential algebra

Reference books for differential algebra are Ritt [1950] and Kolchin [1973]. We also refer to the MAPLE VR5 and following *diffalg* package and to Boulier et al. [1995, 1997], Petitot [1999], Hubert [2000], Boulier and Lemaire [2000].

A *derivation* over a ring R is a map $\delta : R \to R$ such that $\delta(a + b) = \delta a + \delta b$

and $\delta(a\,b) = (\delta a)b + a(\delta b)$ for every $a, b \in R$. A *differential ring* is a ring endowed with finitely many derivations which commute pairwise. The commutative monoid generated by the derivations is denoted by Θ. Its elements are the *derivation operators* $\theta = \delta_1^{a_1} \cdots \delta_m^{a_m}$ where the a_i are nonnegative integer numbers. The sum of the exponents a_i, called the *order* of the operator θ, is denoted by $\operatorname{ord}\theta$. The identity operator is the unique operator with order 0. The other ones are called *proper*. If $\phi = \delta_1^{b_1} \cdots \delta_m^{b_m}$ then $\theta\phi = \delta_1^{a_1+b_1} \cdots \delta_m^{a_m+b_m}$. If $a_i \geq b_i$ for each $1 \leq i \leq m$ then $\theta/\phi = \delta_1^{a_1-b_1} \cdots \delta_m^{a_m-b_m}$.

A *differential ideal* $\mathfrak{a}$ of R is an ideal of R closed under derivation i.e. such that $a \in \mathfrak{a} \Rightarrow \delta a \in \mathfrak{a}$. Let A be a nonempty subset of R. We denote $[A]$ the differential ideal generated by A which is the smallest differential ideal which contains A.

1.4.1. Differential polynomials

Let $U = \{u_1, \ldots, u_n\}$ be a set of *differential indeterminates*. Derivation operators apply over differential indeterminates giving *derivatives* θu. We denote ΘU the set of all the derivatives. Let K be a differential field. The differential ring of the differential polynomials built over the alphabet ΘU with coefficients in K is denoted $R = K\{U\}$.

A *ranking* is a total ordering over the set of the derivatives [Kolchin, 1973, page 75] satisfying the following axioms

1. $\delta v > v$ for each derivative v and derivation δ,
2. $v > w \Rightarrow \delta v > \delta w$ for all derivatives v, w and each derivation δ.

Let us fix a ranking. The infinite alphabet ΘU gets ordered. Consider a polynomial $p \in R \setminus K$. Then the leader, initial, ... of p are well defined. Axioms of rankings imply that the separant of p is the initial of every proper derivative of p.

Let $\operatorname{rank} p = v^d$. A differential polynomial q is said to be *partially reduced* w.r.t. p if no proper derivative of v occurs in q. It is said to be *reduced* w.r.t. p if it is partially reduced w.r.t. p and $\deg(q, v) < d$.

A set A of differential polynomials is said to be *differentially triangular* if it is triangular and if its elements are pairwise partially reduced. It is said to be *autoreduced* if its elements are pairwise reduced. It is said to be *partially auotoreduced* if its elements are pairwise partially reduced. Autoreduced implies differentially triangular.

If A is a set of differential polynomials and v is a derivative then $A_v = \{p \in \Theta A \mid \operatorname{ld} p \leq v\}$. Thus R_v denotes the set of the differential polynomials with leader less than or equal to v.

1.4.2. Ritt's reduction algorithms

One distinguishes the partial reduction algorithm, which is denoted partial_rem from the full reduction algorithm, denoted full_rem. See [Kolchin, 1973, page 77].

Let q and p be two differential polynomials. The *partial remainder* $\mathsf{partial\_rem}\,(q,p)$ is the pseudoremainder of q by the (infinite) set of all the *proper* derivatives of p. The *full remainder* $\mathsf{full\_rem}\,(q,p)$ is the pseudoremainder of q by the set of all the derivatives of p (including p). A precise algorithm is given in [Kolchin, 1973, chapter I, section 9]. Let A be a set of differential polynomials. We denote $\mathsf{partial\_rem}\,(q,A)$ and $\mathsf{full\_rem}\,(q,A)$ respectively the partial remainder and the full remainder of q by all the elements of A.

Let $v = \mathrm{ld}\,q$ and $\overline{A} = A \cap R_v$.

The partial remainder $\overline{q}$ of q by A is partially reduced w.r.t. all the elements of A and there exists a power product h of elements of $S_{\overline{A}}$ such that $h\,q \equiv \overline{q} \mod (\overline{A}_v)$.

The full remainder $\overline{q}$ of q by A is reduced w.r.t. all the elements of A and there exists a power product h of elements of $H_{\overline{A}}$ such that $h\,q \equiv \overline{q} \mod (\overline{A}_v)$.

1.4.3. Critical pairs

A pair $\{p_1, p_2\}$ of differential polynomials is said to be a *critical pair* if the leaders of p_1 and p_2 are derivatives of some same differential indeterminate u (say $\mathrm{ld}\,p_1 = \theta_1 u$ and $\mathrm{ld}\,p_2 = \theta_2 u$). Denote $\theta_{12} u = \mathrm{lcd}(\mathrm{ld}\,p_1, \mathrm{ld}\,p_2)$ the least common derivative of $\mathrm{ld}\,p_1$ and $\mathrm{ld}\,p_2$ defined by $\theta_{12} = \mathrm{lcm}(\theta_1, \theta_2)$.

One distinguishes the *triangular situation* which arises when $\theta_{12} \neq \theta_1$ and $\theta_{12} \neq \theta_2$ from the *nontriangular* one which arises when $\theta_{12} = \theta_2$ (say). In the last case, the critical pair is said to be a *reduction critical pair*. In this article, we do not need to consider the case $\theta_1 = \theta_2$. In the triangular situation, the Δ–polynomial $\Delta(p_1, p_2)$ is

$$\Delta(p_1, p_2) = s_2 \, \frac{\theta_{12}}{\theta_1} \, p_1 - s_1 \, \frac{\theta_{12}}{\theta_2} \, p_2.$$

In the nontriangular one,

$$\Delta(p_1, p_2) = \mathsf{prem}(p_2, \, \frac{\theta_2}{\theta_1} \, p_1).$$

Definition: If $\{p, p'\}$ is a reduction critical pair with (say) $\mathrm{ld}\,p > \mathrm{ld}\,p'$ then $\mathrm{hi}\,(\{p, p'\}) \underset{\mathrm{def}}{=} p$ and $\mathrm{lo}\,(\{p, p'\}) \underset{\mathrm{def}}{=} p'$. If D is a list of critical pairs then

$$\mathrm{hi}(D) \underset{\mathrm{def}}{=} \{\mathrm{hi}\,(\{p, p'\}) \mid \{p, p'\} \text{ is a reduction critical pair of } D\}\,.$$

Definition: A critical pair $\{p, p'\}$ is said to be *solved* by a system $F = 0,\ S \neq 0$ if there exists a derivative $v < \mathrm{lcd}(\mathrm{ld}\,p, \mathrm{ld}\,p')$ such that $\Delta(p, p') \in (F_v) : (S \cap R_v)^\infty$.

1.4.4. Characteristic sets

The traditional definition is due to Ritt: a subset C of a differential ideal $\mathfrak{a}$ is said to be a *characteristic set* of $\mathfrak{a}$ if C is autoreduced and $\mathfrak{a}$ contains no nonzero element reduced w.r.t. C.

We adopt in this paper a slightly more general definition, which relinquishes Ritt's autoreduction requirement and was given by Aubry et al. [1999]. Their definition, given in the purely algebraic setting readily lifts to the differential one.

Definition: A subset C of a differential ideal $\mathfrak{a}$ is said to be a *characteristic set* of $\mathfrak{a}$ if C is differentially triangular, the initials of the elements of C are not reduced to zero by C (by Ritt's full reduction algorithm) and $\mathfrak{a}$ contains no nonzero element reduced w.r.t. C.

Every characteristic set in the sense of Ritt is a characteristic set in the sense of Aubry et al. [1999]. Conversely, if C is a characteristic set in the sense of Aubry et al. [1999], it can be made autoreduced by pseudoreducing each of its elements by the other ones. This autoreduction process does not change the rank of C since it is required that the initials of the elements of C are not reduced to zero by C. Every theorem about Ritt's characteristic sets which only relies on rank considerations therefore applies to the more general definition. The following "well known" proposition provides a useful example.

Proposition 1.7: *If C is a characteristic set of $\mathfrak{a}$ and H_C contains no zero divisor in the quotient ring $R/\mathfrak{a}$ then $\mathfrak{a} = [C] : H_C^\infty$ and $p \in \mathfrak{a}$ if and only if* $\mathsf{full\_rem}\,(p, C) = 0$. *This is the case when $\mathfrak{a}$ is prime.*

Proof: Denote $r = \mathsf{full\_rem}\,(p, C)$. If $p \in \mathfrak{a}$ then $r \in \mathfrak{a}$ and is reduced w.r.t. C. It is thus zero. If r is zero then $p \in \mathfrak{a}$. This concludes the proof of the first claim. The elements of H_C are reduced w.r.t. C. Thus they do not lie in $\mathfrak{a}$. Since $\mathfrak{a}$ is prime they are non zero divisors in $R/\mathfrak{a}$. This concludes the proof of the second claim. □

1.5. Quadruples

Recall the problem addressed in this paper: given a known characteristic set C w.r.t. a ranking $\mathcal{R}$ of a prime differential ideal $\mathfrak{p}$ and a new ranking $\overline{\mathcal{R}}$, compute a characteristic set $\overline{C}$ of $\mathfrak{p}$ w.r.t. $\overline{\mathcal{R}}$.

The ranking implicitly used in this section is the target ranking $\overline{\mathcal{R}}$.

The main data structure handled by the PARDI algorithm is a quadruple $G = \langle A, D, P, S \rangle$. Roughly speaking, A is the set of the differential polynomial equations already processed (it will contain $\overline{C}$ at the end of the computations), D is the set of the critical pairs to be processed, P is the set of the differential polynomial equations to be processed, S is the set of the differential polynomial inequations ($\neq 0$).

Definition: Let $G = \langle A, D, P, S \rangle$ be a quadruple and $F = A \cup \mathrm{hi}(D) \cup P$. The system $F = 0,\ S \neq 0$ is called the system *associated* to G and $I(G) = [F] : S^\infty$ is called the differential ideal *associated* to G.

Definition: If v is any derivative and $F = 0,\ S \neq 0$ is a system then $I^v(F,S)$ denotes the algebraic ideal $(F_v) : (S \cap R_v)^\infty$. If $G = \langle A, D, P, S\rangle$ is a quadruple then $I^v(G) \underset{\text{def}}{=} I^v(F,S)$ where $F = 0,\ S \neq 0$ is the system associated to G.

Definition: A critical pair is said to be *solved* by a quadruple G if it is solved by the system associated to G.

Definition: A critical pair $\{p, p'\}$ is said to be *nearly solved* by a quadruple G if it is solved by G or if it lies in D.

2. Statement of the algorithm

Given a known characteristic set C w.r.t. a ranking $\mathcal{R}$ of a prime differential ideal $\mathfrak{p}$ and a new ranking $\overline{\mathcal{R}}$, we want to compute a characteristic set $\overline{C}$ of $\mathfrak{p}$ w.r.t. $\overline{\mathcal{R}}$. The main data structure is a quadruple $G = \langle A, D, P, S\rangle$. At the end of the computations, the desired characteristic set will be found in A.

One of the main ideas of the algorithm consists in applying a “master–student” relationship between C and A. To decide whether a quantity is zero or not modulo $\mathfrak{p}$ we just need to decide whether this quantity is reduced to zero or not by the “master” C. If it is, we check if it is also reduced to zero by the “student” A and we store in P (the equations to be processed) every quantity reduced to zero by C but not by A.

2.1. Making sure of the rank of a polynomial

The following function provides an easy example of function which applies the “master–student” relationship. It takes as input a differential polynomial p, a quadruple G and the known characteristic set C. It simplifies p while its initial or its separant lies in $\mathfrak{p}$. It returns the simplified value of p together with an updated value of P.

```
function ensure_rank(p, G = ⟨A, D, P, S⟩, C)
begin
  r := p
  newP := P
  while r ∉ K and (i_r ∈ 𝔭 or s_r ∈ 𝔭) do
    if i_r ∈ 𝔭 then
      if prem(i_r, A) ≠ 0 then
        newP := newP ∪ {i_r}
      fi
      r := reductum(r) (i.e. r − i_r x^d) where x^d = rank r)
    else
      if prem(s_r, A) ≠ 0 then
        newP := newP ∪ {s_r}
      fi
```

```
      r := d r − s_r x where x^d = rank r
    fi
  od
  return (r, newP)
end
```

2.2. Invariant properties of quadruples

Throughout the execution of the PARDI algorithm, we will keep the following properties true. Recall $G = \langle A,\, D,\, P,\, S\rangle$ is the quadruple handled by the algorithm.

I1 $\mathfrak{p} = I(G)$;

I2 the set A is a partially autoreduced squarefree regular chain ;

I3 every critical pair made of elements of A is nearly solved by G ;

I4 the initials and separants of the elements of A and of the critical pairs of D belong to S ;

I5 let $\{p, p'\} \in D$ be a reduction pair such that $p' = \mathrm{lo}\,(\{p, p'\})$ and $\mathrm{ld}\, p' = v$; then $p' \in I^v(G)$.

2.3. Completion of a quadruple

One of the key steps of the PARDI algorithm consists in inserting a new differential polynomial p (picked or computed from one of the lists D and P) in the component A of a quadruple $G = \langle A,\, D,\, P,\, S\rangle$. This operation is performed by the complete subfunction below. The parameter C is the known characteristic set of $\mathfrak{p}$.

PROPOSITION 2.1: *The* complete *function takes three parameters: a quadruple $G = \langle A,\, D,\, P,\, S\rangle$ satisfying properties* **I1** *to* **I5***, the known characteristic set C of the differential ideal $\mathfrak{p}$ and a polynomial $p \in \mathfrak{p}$, partially reduced w.r.t. A, with a leader distinct from that of the elements of A and an initial and a separant which do not lie in $\mathfrak{p}$.*

It inserts p in A and returns a quadruple $G' = \langle A',\, D',\, P',\, S'\rangle$ which satisfies properties **I1** *to* **I5** *and such that $I^v(G) \subset I^v(G')$ for every derivative v.*

```
function complete(⟨A, D, P, S⟩, C, p)
begin
  A' := insert_and_rebuild(p, A, C)
  D' := D ∪ {{p_ℓ, p} | p_ℓ ∈ A, {p_ℓ, p} is a critical pair}
  P' := P
  S' := S ∪ {i_p, s_p}
  return ⟨A', D', P', S'⟩
end
```

The function insert_and_rebuild is a subfunction of complete. It inserts p in A giving a new set $\overline{A}$ in such a way that $\overline{A}$ is again a partially autoreduced square-free regular chain. It contains the pseudocode described in proposition 1.6.

```
function insert_and_rebuild(p, A, C)
begin
  Ā := {p} ∪ {f ∈ A | ld f is not a derivative of ld p}
  Denote Ā = {p_1, ..., p_n} (s.t. ld p_i < ld p_{i+1})
  Denote m the index of p in Ā
  k := m
  while k ≤ n do
    p̄_k := partial_rem (p_k, Ā)
    (b, g) := is_regular(i_{p̄_k}, Ā)
    if b then
      (b, g) := is_regular(s_{p̄_k}, Ā)
    fi
    if b is false then
      if g ∉ 𝔭 then
        Replace p_ℓ by pquo(p_ℓ, g) in Ā
      else
        Replace p_ℓ by g in Ā
      fi
    else
      k := k + 1
    fi
  od
  return Ā
end
```

Proposition 2.2: *The* complete *function terminates.*

Proof: The while loop of insert_and_rebuild implements the mechanism whose termination proof is given in proposition 1.6. □

Proposition 2.3: *The pseudocode of the function* complete *satisfies the properties stated in proposition 2.1.*

Proof: We first claim that $I^v(G) \subset I^v(G')$ for each derivative v.

We only need to focus on the operations performed by insert_and_rebuild.

This function withdraws from A' the polynomials of A whose leader is a derivative of the leader of p. These polynomials are stored in reduction critical pairs by complete and they belong to $\mathrm{hi}(D')$ which is part of the associated system of G'. Thus, after this operation, we still have $I^v(G) \subset I^v(G')$ for each derivative v.

This function also performs some algebraic operations on A'. These operations are described by proposition 1.6 which shows that $(A) : H_A^\infty \subset (A') : H_{A'}^\infty$. Thus, after this operation, we still have $I^v(G) \subset I^v(G')$ for each derivative v and the claim is proven.

Property **I1**. We have already proven that $I(G) \subset I(G')$.

We thus only need to prove $I(G') \subset I(G)$. G' is obtained from G by the following operations. The polynomial p is stored in A'. Since $p \in \mathfrak{p}$, after this operation, we still have $I(G') \subset I(G)$. The initial and separant of p are stored in S'. Since these polynomials do not lie in $\mathfrak{p}$ which is prime, after this operation, we have $I(G') \subset \mathfrak{p} : (i_p\, s_p)^\infty = \mathfrak{p}$. Some algebraic operations are performed by insert_and_rebuild on A'. Proposition 1.6, which describes them, shows that $(A') : H_{A'}^\infty \subset \mathfrak{p}$. Hence $I(G') \subset I(G)$ and G' satisfies **I1**.

Property **I2**. The fact that A' is a squarefree regular chain is proven in proposition 1.6. The fact that it is partially autoreduced comes from the fact that p is partially reduced w.r.t. A and that the polynomials of A whose leader is a derivative of the leader of p do not lie in A'.

Property **I3**. It is satisfied by G' since it is satisfied by G and all the critical pairs generated by p and any other element of A are stored in D' (observe A' is the union of $\{p\}$ and of a subset of A).

Property **I4**. It holds for it is satisfied by G and the initial and separant of the new polynomials p inserted in A' are stored in S'.

Property **I5**. It is satisfied by G hence, using the already proven fact that $I^v(G) \subset I^v(G')$ for each derivative v, it holds for reduction critical pairs of D' which are already in D. Reduction critical pairs which lie in D' but not in D are of the form $\{p,\, p_\ell\}$ with $p = \mathrm{lo}(\{p,\, p_\ell\})$. Since $p \in A'$ which is part of the set of equations of the associated system of G' we have $p \in I^v(G')$ where v denotes the leader of p. Hence property **I5** is satisfied by G'. □

2.3.1. Avoiding critical pairs: a new criterion

Not all new critical pairs between p and the elements of A need to be generated. Moreover, some of the critical pairs present in D can be simply removed (i.e. not kept in D').

One can implement an analogue of Buchberger's second criterion as described in Boulier et al. [1997] but the resulting algorithm is quite technical. The following new criterion is much easier to implement and turns out to be very efficient. It only tells us how to remove critical pairs in D but it removes more critical pairs that the analogue of Buchberger's second criterion.

Proposition 2.4: *Let $\{p,\, p'\} \in D$ be a critical pair. If $\{p,\, p'\}$ is not a reduction critical pair and $\{p,\, p'\} \not\subset A'$ then the critical pair does not need to be kept in D'.*

This criterion is proven in the (less interesting) context of Gröbner bases in Boulier [2001]. We are not going to prove it in this paper but the idea is very simple: properties on critical pairs are only useful for proving that the hypotheses of the so called Rosenfeld's lemma hold for the set A at the end of computations (the main loop of PARDI). Therefore critical pairs which contain at least one polynomial withdrawn from A are irrelevant. Now, one must take care not to remove reduction critical pairs for these ones contain generators of the ideal (elements of the set of equations of the associated system of the quadruple). It is surprising that this criterion was not discovered earlier (at least in the context of Gröbner bases, see Becker and Weispfenning [1991]). We believe that this is due to the fact that reduction critical pairs were not distinguished from the other ones while they play a very special role.

2.4. The gcd (lsr sorry) of two polynomials over a factor ring

In this section we consider two polynomials a and b with leader x and a quadruple G. We introduce the following notations:

1. $R^- = K[w \in \Theta U \mid w < x]$
2. $\mathfrak{p}^- = \mathfrak{p} \cap R^-$
3. $I^-(G) = (F \cap R^-) : (S \cap R^-)^\infty$ where $F = 0,\ S \neq 0$ denotes the system associated to G.

Observe that $\mathfrak{p}^-$ is prime, $R^-/\mathfrak{p}^-$ is a domain and $\mathrm{Fr}(R^-/\mathfrak{p}^-)$ is a field.

Proposition 2.5: *The* lsr *function takes five parameters. The two first ones are differential polynomials a, $b \in \mathfrak{p}$ with leader x, partially reduced w.r.t. A and with initials and separants outside $\mathfrak{p}$. The remaining ones are the derivative x, a quadruple G satisfying properties* **I1** *to* **I5** *and the known characteristic set C. It returns a triple (g, newP, newS) satisfying the following properties.*

1. *g is a gcd of a and b in the ring* $\mathrm{Fr}(R^-/\mathfrak{p}^-)[x]$
2. $\deg(g, x) > 0$ *and its initial and separant do not lie in* $\mathfrak{p}$
3. *$(a, b) \subset (g) : h^\infty$ in the ring $(R^-/I^-(G'))[x]$ where h is an element of the multiplicative family generated by newS and $G' = \langle A, D, newP, newS\rangle$.*

The sets *newP* and *newS* are updated version of P and S obtained by applying the "master–student" relationship idea described in section 2.

```
function lsr(a, b, x, G = ⟨A, D, P, S⟩, C)
begin
  p := a
  q := b
  newP := P
  newS := S
  while q ≠ 0 do
    r := prem(p, q, x)
    while r ≠ 0 and lcoeff(r, x) ∈ 𝔭 do
      if prem(lcoeff(r, x), A) ≠ 0 then
```

```
            newP := newP ∪ {lcoeff(r, x)}
        fi
        r := reductum(r, x)
    od
    if r ≠ 0 then
        newS := newS ∪ {lcoeff(r, x)}
        p := q
        q := r
    fi
od
g := p
return (g, newP, newS)
end
```

PROPOSITION 2.6: *The function* lsr *terminates.*

Proof: It is a variant of the Euclidean algorithm. Apart perhaps at the first turn, the degree of q in x strictly decreases at each turn. □

PROPOSITION 2.7: *The pseudocode of* lsr *satisfies its specifications*

Proof: Observe that the pseudocode of lsr is nothing but the Euclidean algorithm applied on a and b in $\mathrm{Fr}(R^-/\mathfrak{p}^-)[x]$ together with instructions which store in *newP* every leading coefficient which is zero in $R^-/\mathfrak{p}^-$ but not reduced to zero by A and stores in *newS* the "true" leading coefficients of the computed pseudoremainders (among the coefficients in R^-, the first one which is nonzero in $R^-/\mathfrak{p}^-$).

Therefore, the polynomial g returned is a gcd of a and b in $\mathrm{Fr}(R^-/\mathfrak{p}^-)[x]$ hence property (1) holds.

All the computed pseudoremainders belong to the ideal (a, b) of the ring $\mathrm{Fr}(R^-/\mathfrak{p}^-)[x]$. Since $a, b \in \mathfrak{p}$, all the computed pseudoremainders lie in $\mathfrak{p}$ thus the first pseudoremainder which does not depend on x is zero in $\mathrm{Fr}(R^-/\mathfrak{p}^-)[x]$. Therefore, the last nonzero pseudoremainder g satisfies $\deg(g, x) > 0$

For this reason, the leading coefficients w.r.t. x are equal to the initials of the computed pseudoremainders and the function explicitly tests that they do not lie in $\mathfrak{p}$.

Denote η a generic zero of $\mathfrak{p}$. It is a zero of a and b but not a zero of their separants s_a and s_b since these latter do not lie in $\mathfrak{p}$. Therefore η is a simple zero of a and b hence a simple zero of their gcd g. Thus η is not a zero of the separant s_g of g and $s_g \notin \mathfrak{p}$. This concludes the proof of property (2).

It is a well-known property of the Euclidean algorithm that $a, b \in (g) : h^\infty$ in $\mathrm{Fr}(R^-/\mathfrak{p}^-)[x]$ where h denotes the product of the leading coefficients of the computed pseudoremainders. On the one hand, these leading coefficients are stored in *newS* which is part of the set of inequations of the system associated to G'. On the other hand, the leading coefficients in R^- of the pseudoremainders which are zero in $R^-/\mathfrak{p}^-$ are either stored in *newP* (which is part of the set of equations of the system associated to G') or reduced to zero by A. Thus property (3) holds. $\square$

2.4.1. Performing exact quotient operations

In practical implementations, the returned gcd is actually the last nonzero subresultant of a and b and the computation is performed using a variant of a (good) pseudoremainder sequence algorithm (we chose the algorithm of Ducos [2000] but the Lombardi et al. [2000] algorithm would fit as well).

Such an algorithm actually computes a sequence of subresultants $p_1, \ldots, p_n$ of a and b in $(R^-/\mathfrak{p}^-)[x]$.

The only issue with such efficient algorithms consists in performing the exact quotient operations of the algorithm in $R^-/\mathfrak{p}^-$. Let's describe how we proceed.

At each step i we verify that the leading coefficient of the current subresultant p_i is nonzero in $R^-/\mathfrak{p}^-$. Assume this is the case. Then one continues the Ducos [2000] algorithm without normalizing p_i in any sense w.r.t. $\mathfrak{p}$. Assume the leading coefficients of all the encountered subresultants are nonzero in $R^-/\mathfrak{p}^-$. Then the algorithm behaves exactly as Ducos [2000] in $R^-[v]$ whence exact quotient operations just have to be done in R^-. Assume now that the leading coefficient of p_i is zero in $R^-/\mathfrak{p}^-$. Then one replaces p_i by its reductum (i.e. one removes this coefficient from p_i), possibly many times, giving a polynomial $\overline{p}_i$. Then one restarts lsr over p_{i-1} and $\overline{p}_i$.

This idea is very simple but very important. Elements of $R^-/\mathfrak{p}^-$ are residue classes. They can be computationally represented by any of their elements. For pseudoremainder sequences algorithms, the most convenient choice is to represent residue classes by representatives which make easy the exact quotient operations. This can be done by not normalizing coefficients at all. One just needs to make sure that leading coefficients are nonzero in the factor ring.

2.4.2. Identifying algebraic subproblems

The lsr algorithm is purely algebraic in the following sense:

1. it does not manipulate the separants of the polynomials p and q ;
2. it does not generate any critical pair.

It is going to be used by the PARDI algorithm when two differential polynomials having the same leader are encountered. This is a true major improvement w.r.t. the Rosenfeld–Gröbner algorithm of the MAPLE *diffalg* package as explained in Boulier et al. [2001a].

2.5. The main algorithm

function PARDI$(C, \mathcal{R}, \overline{\mathcal{R}})$
begin

```
  ⟨A, D, P, S⟩ := ⟨∅, ∅, C, H_C taken w.r.t. 𝓡⟩
  while D ≠ ∅ or P ≠ ∅ do
    Take and remove some p ∈ P or some critical pair {p1, p2} ∈ D.
        In the latter case let p = Δ(p1, p2).
    p̄ := partial_rem (p, A)
    (p̄, P) := ensure_rank(p̄, G = ⟨A, D, P, S⟩, C)
    if p̄ ≠ 0 then
      if ∃q ∈ A such that ld p̄ = ld q then
        (g, P, S) := lsr(p̄, q, ld q, ⟨A, D, P, S⟩, C)
        if g ≠ q then
          ⟨A, D, P, S⟩ := complete(⟨A \ {q}, D, P, S⟩, C, g)
        fi
      else
        ⟨A, D, P, S⟩ := complete(⟨A, D, P, S⟩, C, p̄)
      fi
    fi
  od
  S := partial_rem (S, A)
  return strip_charset(⟨A, D, P, S⟩, C, 𝓡, 𝓡̄)
end
```

The strip_charset function will be described later.

Proposition 2.8: *The main loop of the* PARDI *algorithm terminates.*

Proof: The rank of A decreases at each turn w.r.t. the classical ordering on autoreduced sets. This rank cannot strictly decrease at each turn by [Kolchin, 1973, proposition 3, page 81]. It is sufficient to establish that it cannot indefinitely keep the same value.

The rank of A does not change only if $g = q$ after a call to lsr or all the coefficients of the differential polynomial picked and removed from P or computed from a critical pair of D belong to $\mathfrak{p}$.

In the three cases, the algorithm does not generate any critical pair (provided that the case $g = q$ is handled separately after a call to lsr). Therefore it is impossible to extract infinitely many critical pairs from D and it is sufficient to consider the two first cases: In these two cases, one differential polynomial is picked from P and is replaced by finitely many differential polynomials with a lower leader. Rankings are well orderings [Kolchin, 1973, page 75]. By a classical argument of graph theory [König, 1950, Satz 6.6] (i.e. every infinite, locally finite tree involves a branch of infinite length) this cannot happen infinitely many times. Thus the algorithm terminates. □

Proposition 2.9: *Property* **I4** *is a loop invariant of* PARDI.

Proof: This property is satisfied initially. The only function which modifies A or D is the complete function. The proof is concluded by proposition 2.1. □

Before proving that the other properties **I1** to **I5** are loop invariants of PARDI, we establish a lemma which proves that the ideals $I^v(G)$ grow i.e. that if

$$v_1 < v_2 < v_3 < \cdots$$

is an increasing sequence of derivatives then

$$\begin{array}{ccccccc} I^{v_1}(G) & \subset & I^{v_2}(G) & \subset & I^{v_3}(G) & \subset & \cdots \\ \cap & & \cap & & \cap & & \\ I^{v_1}(G') & \subset & I^{v_2}(G') & \subset & I^{v_3}(G') & \subset & \cdots \end{array}$$

This lemma is very important for it proves that if a critical pair is solved at some loop iteration then it keeps being solved afterwards.

PROPOSITION 2.10: *Denote $G = \langle A,\, D,\, P,\, S\rangle$ the value of the quadruple at the beginning of the loop body and $G' = \langle A',\, D',\, P',\, S'\rangle$ its value after execution of the loop body. Assume G satisfies property* **I5**. *Then*

(α) *$I^v(G) \subset I^v(G')$ for every derivative v ;*

(β) *G' satisfies property* **I5**.

Proof: Denote $F = 0,\ S \neq 0$ the system associated to G and $F' = 0,\ S' \neq 0$ the system associated to G'.

We first consider the case p is picked from the set P.

Denote v the leader of p and $\overline{p}$ the partial remainder of p by A. Then, for some $h \in S \cap R_v$ we have $h\,p = \overline{p} \mod I^v(G)$. The call to ensure_rank may modify $\overline{p}$ but stores in P the initials and separants needed to keep this relation true.

Now, if $\overline{p} = 0$ then $p \in I^v(G)$ and (α) is proven. Since D is not modified, (β) is proven also. Assume then $\overline{p} \neq 0$.

If there is not any $q \in A$ having the same leader as $\overline{p}$ then complete is called and, using proposition 2.1, the proposition is proven. Observe though that, when complete is called, G does not satisfy **I1** to **I5** since p is already withdrawn from P. However, for the needs of the proof, we may assume that we have delayed the withdrawal of p from P: once $\overline{p}$ is inserted in A, the polynomial p is redundant.

If there does exist some $q \in A$ having the same leader as $\overline{p}$ then, by proposition 2.5, the call to lsr provides a gcd g of $\overline{p}$ and q which has leader v and satisfies $\overline{p},\, q \in (g) : h^\infty$ in the ring $\mathrm{Fr}(R^-/\mathfrak{p}^-)[v]$ where $h \in S \cap R_v$ (the value of S being the one updated by lsr). This gcd is inserted in G by complete hence, using proposition 2.1 as above, the proposition is proven. Here also, observe that when complete is called, G does not satisfy **I1** to **I5** since p is already withdrawn from P and q from A. However, for the needs of the proof, we may assume that we have delayed the withdrawal of those polynomials: once g is inserted in A, they are redundant.

This concludes the case p is picked from P.

We now consider the case a critical pair is picked from D. First observe we only need to focus on the case of a reduction critical pair since the other ones do not enter the definition of the associated systems of the quadruples.

To shorten the proof, we may also assume that Δ–polynomials are temporarily stored in P before being handled by the remaining instructions of the loop body. That way, relying on the analysis of the case p is picked from P, we only need to prove that (α) and (β) hold when a reduction critical pair is picked and removed from D and the corresponding Δ–polynomial is stored in P.

Both properties are proven by induction on v.

The basis (v is the lowest derivative). $F_v \neq F'_v$ only if the critical pair $\{p, p'\}$ is such that $\mathrm{ld}(\mathrm{hi}(\{p, p'\})) = v$. No such pair exists. Thus (α) and (β) hold.

The general case. We assume both properties hold for every derivative $v' < v$. We prove them for v. Assume the critical pair $\{p, p'\}$ is such that $\mathrm{ld}\, p > \mathrm{ld}\, p'$.

If $v = \mathrm{ld}\, p'$ then $F_v = F'_v$ and $S = S'$ and (α) holds. Property (β) follows from (α) and the fact that we have just removed one pair from D.

If $v = \mathrm{ld}\, p$ then it is sufficient to prove $p \in I^v(G')$. Since the critical pair is a reduction one we have $\Delta(p, p') = \mathsf{prem}(p, \phi p')$ where $\phi = \theta/\theta'$, denoting $v = \theta u$, and $v' = \mathrm{ld}\, p' = \theta' u$. We have $p' \in I^{v'}(G)$ and $I^{v'}(G) \subset I^{v'}(G')$ by induction hypothesis whence $\phi p' \in I^v(G')$ and $p \in I^v(G')$. Property (β) follows from (α) and the fact that we have just removed one pair from D. □

Proposition 2.11: *Properties* **I1** *to* **I5** *are loop invariants of* PARDI.

Proof: These properties are all satisfied initially. The case of **I4** is already solved.

Then proposition 2.10 proves that **I5** is also a loop invariant and that the ideals $I^v(G)$ grow.

Invariant **I1**. The inclusion $\mathfrak{p} \subset I(G)$ comes from proposition 2.10 (α). The converse inclusion is clear.

Invariant **I2** comes from proposition 2.1.

Invariant **I3**. Proposition 2.10 (α) proves that any critical pair solved by G is solved by G'. Consider a critical pair $\{p,\, p'\}$ removed from D. It is nearly solved by G since it lies in D. It is solved (hence nearly solved) by G' for the Δ–polynomial is stored in A' by complete and has a leader strictly less than the leader of $\mathrm{hi}(\{p,\, p'\})$. The case of the critical pairs generated by complete is considered in proposition 2.1. □

2.6. An example

We illustrate the behaviour of PARDI over the first example given in introduction and verify that some of the properties stated in section 2.2 hold at each step. The criterion stated in proposition 2.4 is applied (though it is not proven in this paper). The reader may check that the avoided corresponding Δ–polynomial are reduced to zero by the set A of the quadruple.

This implementation of PARDI was done in the C programming language. It maintains invariant **I2’** and performs full reductions instead of partial ones. The strategy consists in picking differential polynomials in P first and critical pairs in D only if P is empty. The lists P and D are sorted according to some heuristic criterion which does not need to be stated.

Each time a differential polynomial is written, its leader w.r.t. the ranking $\overline{\mathcal{R}}$ occurs at the leftmost place: it is the first derivative written.

When PARDI first enters the loop, A and D are empty, all elements of C belong to P and the initials and separants of C taken w.r.t. the ranking $\mathcal{R}$ lie in S i.e.

$$\begin{array}{rcl} A & = & [] \\ D & = & [] \\ P & = & [u_y^2 - 2u,\, -u_x + v_{xx},\, u_x^2 - 4u,\, -u_x u_y u + u_x u_y + 4uv_y] \\ S & = & [u,\, u_x,\, u_y] \end{array}$$

Since $F = P = C$ and $S = H_C$ invariant **I1** is clearly satisfied. The other invariants are initially trivial.

At the first turn, the differential polynomial $u_y^2 - 2u$ is picked and removed from P. The complete subfunction stores it in A'. It inserts also its initial and separant in S' to keep invariant **I4** but this operation has no effect for the initial lies in the base field and the separant is already present in S.

$$\begin{array}{rcl} A & = & [u_y^2 - 2u] \\ D & = & [] \\ P & = & [-u_x + v_{xx},\, u_x^2 - 4u,\, -u_x u_y u + u_x u_y + 4uv_y] \\ S & = & [u,\, u_x,\, u_y] \end{array}$$

At the second turn the differential polynomial $-u_x+v_{xx}$ is picked and removed from P. The complete subfunction inserts it in A' (after normalizing its sign) and stores in D' a critical pair generated by this differential polynomial and the one already present in A.

$$\begin{array}{rcl} A & = & [u_y^2 - 2u,\, u_x - v_{xx}] \\ D & = & [\{u_x - v_{xx},\, u_y^2 - 2u\}] \\ P & = & [u_x^2 - 4u,\, -u_x u_y u + u_x u_y + 4uv_y] \\ S & = & [u,\, u_x,\, u_y] \end{array}$$

At the third turn the differential polynomial $u_x^2 - 4u$ is picked and removed from P. After reduction by A it becomes $4u - v_{xx}^2$. The complete subfunction stores it in A' and stores in D' the two (reduction) critical pairs generated by this differential polynomial and the elements of A.

The former elements $u_y^2 - 2u$ and $u_x - v_{xx}$ of A are not kept in A'. These two differential polynomials belong however to $\mathrm{hi}(D')$ so that

$$F \underset{\mathrm{def}}{=} A \cup \mathrm{hi}(D) \cup P = F' \underset{\mathrm{def}}{=} A' \cup \mathrm{hi}(D') \cup P'.$$

Thus $I(G) = I(G')$ and invariant **I1** still holds. The old critical pair in D is not kept in D' by proposition 2.4.

$$\begin{array}{rcl} A &=& [4u - v_{xx}^2] \\ D &=& [\{4u - v_{xx}^2,\, u_y^2 - 2u\},\, \{4u - v_{xx}^2,\, u_x - v_{xx}\}] \\ P &=& [-u_x u_y u + u_x u_y + 4uv_y] \\ S &=& [u,\, u_x,\, u_y] \end{array}$$

At the fourth turn the differential polynomial $-u_x u_y u + u_x u_y + 4uv_y$ is picked and removed from P.

The differential polynomial obtained after the reduction by A has a nontrivial content. The implementation of PARDI verifies that this content does not lie in $\mathfrak{p}$ and removes it. The obtained differential polynomial is $v_{xxx}v_{xxy}v_{xx}^2 - 4v_{xxx}v_{xxy} - 16v_y$. It is stored in A' and its initial (splitted in two factors) is stored in S'.

$$\begin{array}{rcl} A &=& [v_{xxx}v_{xxy}v_{xx}^2 - 4v_{xxx}v_{xxy} - 16v_y,\, 4u - v_{xx}^2] \\ D &=& [\{4u - v_{xx}^2,\, u_y^2 - 2u\},\, \{4u - v_{xx}^2,\, u_x - v_{xx}\}] \\ P &=& [] \\ S &=& [v_{xxy},\, v_{xx}^2 - 4,\, u,\, u_x,\, u_y] \end{array}$$

At the fifth turn, the critical pair $\{4u - v_{xx}^2,\, u_y^2 - 2u\}$ is picked and removed from D. The Δ–polynomial is computed, reduced by A. The content of the result is removed. This provides the new differential polynomial $v_{xxy}^2 - 2$.

It is stored in A' and one critical pair is stored in D'. It has a trivial initial and its separant is already present in S.

The implementation of insert_and_rebuild, which maintains invariant **I2'**, multiplies the differential polynomial with leader v_{xxx} by the algebraic inverse $v_{xxy}/2$ of v_{xxy} modulo $(v_{xxy}^2 - 2)$ and simplifies using the new differential polynomial.

$$\begin{array}{rcl} A &=& [v_{xxy}^2 - 2,\, v_{xxx}v_{xx}^2 - 4v_{xxx} - 8v_{xxy}v_y,\, 4u - v_{xx}^2] \\ D &=& [\{4u - v_{xx}^2,\, u_x - v_{xx}\},\, \{v_{xxy}^2 - 2,\, v_{xxx}v_{xx}^2 - 4v_{xxx} - 8v_{xxy}v_y\}] \\ P &=& [] \\ S &=& [v_{xxy},\, v_{xx}^2 - 4,\, u,\, u_x,\, u_y] \end{array}$$

At the sixth turn, the critical pair $\{4u - v_{xx}^2,\, u_x - v_{xx}\}$ is picked and removed from D. The Δ–polynomial is computed and reduced by A. The content of the result is removed. This provides the new differential polynomial $p = 4v_{xxy}v_y - v_{xx}^2 + 4$. Since p has the same leader as the element $q = v_{xxy}^2 - 2 \in A$ the lsr function is called.

The lsr function is called in order to compute a gcd g of $q = v_{xxy}^2 - 2$ and $p = 4v_{xxy}v_y - v_{xx}^2 + 4$ in the ring $\mathrm{Fr}(R^-/\mathfrak{p}^-)[v_{xxy}]$. Here $R^- = K[w \in \Theta U \mid w < v_{xxy}]$.

The pseudoremainder $\mathsf{prem}(q, p, v_{xxy})$ is the polynomial $r = v_{xx}^4 - 8v_{xx}^2 - 32v_y^2 + 16$. It does not involve v_{xxy} thus it is equal to its leading coefficient w.r.t. this derivative. The characteristic set C of $\mathfrak{p}$ pseudoreduces r to zero (notice that proposition 2.5 proves this fact without having to perform the pseudoreductions). Thus the gcd $g = p$. Since r is not reduced to zero by A, it is stored in P. The initial $4\,v_y$ of g is stored in S.

The old critical pair in D is removed using proposition 2.4.

$$\begin{array}{rcl} A & = & [4v_{xxy}v_y - v_{xx}^2 + 4,\, v_{xxx} - 2,\, 4u - v_{xx}^2] \\ D & = & [\{4v_{xxy}v_y - v_{xx}^2 + 4,\, v_{xxx} - 2\}] \\ P & = & [v_{xx}^4 - 8v_{xx}^2 - 32v_y^2 + 16] \\ S & = & [v_y,\, v_{xxy},\, v_{xx}^2 - 4,\, u,\, u_x,\, u_y] \end{array}$$

At the seventh turn the differential polynomial $v_{xx}^4 - 8v_{xx}^2 - 32v_y^2 + 16$ is picked and removed from P. It is stored in A' and throws away the differential polynomials with leaders v_{xxy} and v_{xxx}. Its separant factors. Only the factor v_{xx} needs to be stored in S'. Two reduction critical pairs are stored in D' but the old critical pair can be removed using proposition 2.4.

$$\begin{array}{rcl} A & = & [v_{xx}^4 - 8v_{xx}^2 - 32v_y^2 + 16,\, 4u - v_{xx}^2] \\ D & = & [\{v_{xx}^4 - 8v_{xx}^2 - 32v_y^2 + 16,\, v_{xxx} - 2\}, \\ & & \{v_{xx}^4 - 8v_{xx}^2 - 32v_y^2 + 16,\, 4v_{xxy}v_y - v_{xx}^2 + 4\}] \\ P & = & [] \\ S & = & [v_{xx},\, v_y,\, v_{xxy},\, v_{xx}^2 - 4,\, u,\, u_x,\, u_y] \end{array}$$

At the eighth turn, the first critical pair is picked and removed from D. After reduction by A, it provides a differential polynomial $p = v_{xx}^3 - 4v_{xx} - 8v_{xy}v_y$.

The lsr function is called on p and the polynomial $q \in A$ with rank v_{xx}^4. It returns a gcd $g = v_{xx}v_{xy}^2 - 2v_{xx} - 4v_{xy}v_y$ (after removal of its content). It stores in P' a differential polynomial representing the resultant of p and q (after removal of its content) but which is not reduced to zero by A. The initial of g is stored in S'.

The gcd is stored in A' and replaces the differential polynomial with rank v_{xx}^4. The function $\mathsf{insert\_and\_rebuild}$ pseudoreduces the differential polynomial with leader u using it. This does not simplify this polynomial at first sight. No critical pair is generated.

$$\begin{array}{rcl} A & = & [v_{xx}v_{xy}^2 - 2v_{xx} - 4v_{xy}v_y,\, uv_{xy}^4 - 4uv_{xy}^2 + 4u - 4v_{xy}^2v_y^2] \\ D & = & [\{v_{xx}^4 - 8v_{xx}^2 - 32v_y^2 + 16,\, 4v_{xxy}v_y - v_{xx}^2 + 4\}] \\ P & = & [v_{xy}^4 - 4v_{xy}^2 - 8v_y^2 + 4] \\ S & = & [v_{xy}^2 - 2,\, v_{xx},\, v_y,\, v_{xxy},\, v_{xx}^2 - 4,\, u,\, u_x,\, u_y] \end{array}$$

At the ninth turn, the differential polynomial $v_{xy}^4 - 4v_{xy}^2 - 8v_y^2 + 4$ is picked and removed from P. It is stored in A'. Its separant factors. Only the factor v_{xy} needs to be stored in S'. One critical pair is stored in D'.

$$
\begin{array}{rcl}
A & = & [v_{xy}^4 - 4v_{xy}^2 - 8v_y^2 + 4,\ 2v_{xx}v_y - v_{xy}^3 + 2v_{xy},\ 2u - v_{xy}^2] \\
D & = & [\{v_{xx}^4 - 8v_{xx}^2 - 32v_y^2 + 16,\ 4v_{xxy}v_y - v_{xx}^2 + 4\}, \\
 & & \{v_{xy}^4 - 4v_{xy}^2 - 8v_y^2 + 4,\ 2v_{xx}v_y - v_{xy}^3 + 2v_{xy}\}] \\
P & = & [] \\
S & = & [v_{xy},\ v_{xy}^2 - 2,\ v_{xx},\ v_y,\ v_{xxy},\ v_{xx}^2 - 4,\ u,\ u_x,\ u_y]
\end{array}
$$

At the tenth turn, the first critical pair is picked and removed from D. The Δ–polynomial is computed and reduced by A. The content of the result is removed. One gets a differential polynomial $p = v_{xy}^3 - 2v_{xy} - 4v_{yy}v_y$.

The situation is very similar to that of the eighth turn. The lsr function is called on p and the polynomial $q \in A$ with rank v_{xy}^4. It returns a gcd $g = v_{xy}v_{yy}^2 - v_{xy} - 2v_{yy}v_y$ (after removal of its content). It stores in P' a differential polynomial representing the resultant of p and q (after removal of its content) but which is not reduced to zero by A. The initial of g is stored in S'.

The implementation of insert_and_rebuild pseudoreduces the differential polynomial with leader v_{xx} using it (this does not simplify the differential polynomial).

The gcd is stored in A' and replaces the differential polynomial with rank v_{xy}^4. The function insert_and_rebuild pseudoreduces the differential polynomial with leader v_{xx} using it. This does not simplify this polynomial at first sight. A critical pair is stored in D'. An old critical pair is removed from D using proposition 2.4.

$$
\begin{array}{rcl}
A & = & [v_{xy}v_{yy}^2 - v_{xy} - 2v_{yy}v_y, \\
 & & v_{xx}v_{yy}^6 - 3v_{xx}v_{yy}^4 + 3v_{xx}v_{yy}^2 - v_{xx} + 2v_{yy}^5 - 4v_{yy}^3v_y^2 - 4v_{yy}^3 + 2v_{yy}, \\
 & & uv_{yy}^4 - 2uv_{yy}^2 + u - 2v_{yy}^2v_y^2] \\
D & = & [\{v_{xy}v_{yy}^2 - v_{xy} - 2v_{yy}v_y, \\
 & & \quad v_{xx}v_{yy}^6 - 3v_{xx}v_{yy}^4 + 3v_{xx}v_{yy}^2 - v_{xx} + 2v_{yy}^5 - 4v_{yy}^3v_y^2 - 4v_{yy}^3 + 2v_{yy}\}] \\
P & = & [v_{yy}^4 - 2v_{yy}^2 - 2v_y^2 + 1] \\
S & = & [v_{yy}^2 - 1,\ v_{xy},\ v_{xy}^2 - 2,\ v_{xx},\ v_y,\ v_{xxy},\ v_{xx}^2 - 4,\ u,\ u_x,\ u_y]
\end{array}
$$

At the eleventh turn, the differential polynomial $v_{yy}^4 - 2v_{yy}^2 - 2v_y^2 + 1$ is picked and removed from P. It is inserted in A'. The function insert_and_rebuild pseudoreduces the other differential polynomials of A' using it and removes the contents. This simplifies A'. An analogue of Buchberger's second criterion Boulier et al. [1997] not stated in this paper permits us to generate only one critical pair instead of two (the fact that the corresponding Δ–polynomial is reduced to zero by A can be checked directly).

$$\begin{aligned}
A &= [v_{yy}^4 - 2v_{yy}^2 - 2v_y^2 + 1,\ v_{xy}v_y - v_{yy}^3 + v_{yy},\ v_{xx} - 2v_{yy},\ u - v_{yy}^2] \\
D &= [\{v_{xy}v_y - v_{yy}^3 + v_{yy},\ v_{xx} - 2v_{yy}\}, \\
&\quad \{v_{yy}^4 - 2v_{yy}^2 - 2v_y^2 + 1,\ v_{xy}v_y - v_{yy}^3 + v_{yy}\}] \\
P &= [] \\
S &= [v_{yy},\ v_{yy}^2 - 1,\ v_{xy},\ v_{xy}^2 - 2,\ v_{xx},\ v_y,\ v_{xxy},\ v_{xx}^2 - 4,\ u,\ u_x,\ u_y]
\end{aligned}$$

There are two critical pairs left. At the next steps, A pseudoreduces the first one to zero, it pseudoreduces the second one to zero and PARDI calls strip_charset.

3. The strip_charset algorithm

The following definition is borrowed from Boulier et al. [1997].

Definition: A differential system $A = 0,\ S \neq 0$ is a *regular differential system* if

C1 A is differentially triangular (partially autoreduced and triangular) ;

C2 the separants of A belong to S and S is partially reduced w.r.t. A ;

C3 all the critical pairs that can be formed with the elements of A are solved by the system $A = 0,\ S \neq 0$.

PROPOSITION 3.1: *Let $G = \langle A,\ D,\ P,\ S\rangle$ denote the quadruple that* PARDI *provides to* strip_charset *as first parameter.*

Then $A = 0,\ S \neq 0$ is a regular differential system and $[A] : S^\infty = \mathfrak{p}$.

Proof: The quadruple G satisfies properties **I1** to **I5** by proposition 2.11. It also satisfies $D = P = \emptyset$.

Property **I2** implies property **C1**. Property **I4** and the fact that PARDI partially reduces the elements of S by A just before calling strip_charset implies that **C2** holds. Property **I3** combined with the fact that D is empty implies that **C3** holds. Therefore $A = 0,\ S \neq 0$ is a regular differential system.

Property **I1** combined to the fact that $D = P = \emptyset$ imploes that $[A] : S^\infty = \mathfrak{p}$. □

We recall the two main theorems satisfied by regular differential systems.

THEOREM 3.1: *(Rosenfeld's lemma)*

If $A = 0,\ S \neq 0$ is a regular differential system then $[A] : S^\infty \cap R_0 = (A) : S^\infty$ where R_0 denotes the ring of the differential polynomials partially reduced w.r.t. A.

Proof: Rosenfeld's lemma is due to Rosenfeld [1959] who generalizes a result of Seidenberg [1956]. Its formulation for regular differential systems is given in Boulier et al. [1997]. A generalized version is given in Morrison [1999]. □

THEOREM 3.2: *(Lazard's lemma)*

Let A be a triangular set of a polynomial ring $K[X]$ and S_A be the set of its separants. The ideal $(A) : S_A^\infty$ is radical. The set of the indeterminates which are not leaders of elements of A provides a transcendence basis of the field of fractions of $K[X]/\mathfrak{b}$ over K where $\mathfrak{b}$ is any prime ideal minimal over $(A) : S_A^\infty$.

Proof: Lazard's lemma is stated for the first time in Boulier et al. [1995] with an incomplete proof. The first complete proof is due to Morrison [1995] and published in Morrison [1999]. Different proofs of Lazard's lemma were written later by Schicho and Li [1995], Boulier et al. [1997], Hubert [2000] and Sadik [2000]. See also [Boulier et al., 2001b, theorems 1 and 2]. □

THEOREM 3.3: *(corollary to Lazard's and Rosenfeld's lemmas)*

Let $A = 0,\ S \neq 0$ be a regular differential system. Then the ideal $[A] : S^\infty$ is radical. Denote $\mathfrak{p}_1, \ldots, \mathfrak{p}_t$ its minimal differential primes, R_0 the ring of the differential polynomials partially reduced w.r.t. A and $\mathfrak{b}_i = \mathfrak{p}_i \cap R_0$. Then $\mathfrak{b}_1, \ldots, \mathfrak{b}_t$ are the minimal primes of $(A) : S^\infty$.

Proof: This theorem is already present in [Boulier et al., 1997, lifting of Lazard's lemma] and Hubert [2000]. We prove it anew.

The first part is well known. Assume $p^\alpha \in [A] : S^\infty$. Denote $\overline{p}$ =partial_rem (p, A). For some power product h of separants of A we have $h\,p = \overline{p} \mod [A]$. By Rosenfeld's lemma $\overline{p}^\alpha \in (A) : S^\infty$. Thus by Lazard's lemma, $\overline{p} \in (A) : S^\infty$. Since the separants of A are inequations of the system $A = 0,\ S \neq 0$ we have $p \in [A] : S^\infty$.

The second part. We have $(A) : S^\infty = [A] : S^\infty \cap R_0$ by Rosenfeld's lemma whence $(A) : S^\infty = \mathfrak{b}_1 \cap \cdots \cap \mathfrak{b}_t$. It is thus sufficient to prove that none of the $\mathfrak{b}$ is redundant. Let $p \in \mathfrak{p}_2 \cap \cdots \cap \mathfrak{p}_t$. Let $\overline{p}$ and h be defined as above. Then $\overline{p} \in \mathfrak{b}_2 \cap \cdots \cap \mathfrak{b}_t$. Assume $\mathfrak{b}_1$ is redundant. Then $\overline{p} \in \mathfrak{b}_1$. Since $A, \mathfrak{b}_1 \subset \mathfrak{p}_1$ and $h \notin \mathfrak{p}_1$ we conclude that $p \in \mathfrak{p}_1$ i.e. that $\mathfrak{p}_1$ is redundant. Contradiction. □

The strip_charset algorithm can be implemented by at least two different algorithms. The first one, called specialized_regCharacteristic, always applies. The second one, called regalise always applies in the algebraic case or for ODE systems. It may not work for PDE systems (without a few preliminary computations).

PROPOSITION 3.2: *The* specialized_regCharacteristic *function takes four parameters: a regular differential system $A = 0,\ S \neq 0$ w.r.t. ranking $\overline{\mathcal{R}}$ such that $[A] : S^\infty = \mathfrak{p}$, the known characteristic set C of $\mathfrak{p}$ w.r.t. ranking $\mathcal{R}$ and the two rankings.*

It returns a characteristic set C of $\mathfrak{p}$ w.r.t. $\overline{\mathcal{R}}$.

function specialized_regCharacteristic($\{A = 0,\ S \neq 0\}$, C, $\mathcal{R}$, $\overline{\mathcal{R}}$)
begin
  Denote $S = \{s_1, \ldots, s_m\}$

```
  C̄ := A
  k := 1
  while k ≤ m do
    (b, g) := is_regular(s_k, A)
    if b is false then
      Let x_ℓ be the leader of g
      if g ∉ 𝔭 then
        Replace p_ℓ by pquo(p_ℓ, g) in C̄
      else
        Replace p_ℓ by g in C̄
      fi
    else
      k := k + 1
    fi
  od
  return C̄
end
```

Over the example, is_regular always returns pairs of the form (*true*, $\cdot$) and the set $A = \overline{C}$ is returned.

Proposition 3.3: *The function* specialized_regCharacteristic *terminates.*

Proof: This function implements the mechanism whose termination proof is given in proposition 1.6. □

Proposition 3.4: *The pseudocode of function* specialized_regCharacteristic *satisfies the properties stated in proposition 3.2.*

Proof: The rings of the differential polynomials partially reduced w.r.t. $\overline{C}$ and A are the same. Denote it R_0. The set $\overline{C}$ is a partially autorduced squarefree regular chain and by proposition 1.6 we have:

$$(A) : H_A^\infty \subset (A) : S^\infty \subset (\overline{C}) : H_{\overline{C}}^\infty \subset \mathfrak{p} \cap R_0.$$

By Rosenfeld's lemma $[A] : S^\infty \cap R_0 = (A) : S^\infty$. Since $[A] : S^\infty = \mathfrak{p}$ we have $(A) : S^\infty = \mathfrak{p} \cap R_0$ hence, using the above inclusions $\mathfrak{p} \cap R_0 = (\overline{C}) : H_{\overline{C}}^\infty$. Now, consider any $p \in \mathfrak{p}$, reduced w.r.t. $\overline{C}$. It belongs to $\mathfrak{p} \cap R_0$. It must then be zero for $\overline{C}$ is a characteristic set (a regular chain) of that ideal. Thus $\overline{C}$ is a characteristic set in the differential sense of $\mathfrak{p}$. □

3.1. The regalise subalgorithm

It computes $\overline{C}$ from A and the known characteristic set C of $\mathfrak{p}$. Basic idea: building a set $\overline{C}$ which reduces C to zero and such that the initials and separants of C are non zero divisors modulo the ideal defined by $\overline{C}$. This gives us the inclusion $\mathfrak{p} = [C] : H_C^\infty \subset [\overline{C}] : H_{\overline{C}}^\infty$. The other inclusion follows easily from the way $\overline{C}$ is built.

The basic idea does not work in general because the system $A = 0,\ H_A \neq 0$ does not necessarily satisfy condition **C3** whence is not necessarily a regular differential system though $A = 0,\ S \neq 0$ is.

Observe that condition **C3** is irrelevant for both purely algebraic and ordinary differential equations. In these cases, the basic idea works perfectly.

PROPOSITION 3.5: *The* regalise *function takes four parameters: a regular differential system $A = 0,\ S \neq 0$ w.r.t. ranking $\overline{\mathcal{R}}$ such that $[A] : S^\infty = \mathfrak{p}$, the known characteristic set C of $\mathfrak{p}$ w.r.t. ranking $\mathcal{R}$ and the two rankings.*

It is assumed moreover that $A = 0,\ S \neq 0$ is an ordinary differential system or that it is a partial differential system such that every Δ–polynomial that can be formed between any two elements of A is reduced to zero by A.

It returns a characteristic set C of $\mathfrak{p}$ w.r.t. $\overline{\mathcal{R}}$.

function regalise($\{A = 0,\ S \neq 0\},\ C,\ \mathcal{R},\ \overline{\mathcal{R}}$)
begin
  Denote $C = \{f_1, \ldots, f_n\}$
  $\overline{C} := A$
  $k := 1$
  while $k \leq n$ do
    $\overline{f}_k :=$ partial_rem $(f_k, \overline{C})$
    if prem$(\overline{f}_k, \overline{C}) \neq 0$ then
      $(b,\ g) :=$ is_regular$(\overline{f}_k, \overline{C})$
      This call necessarily returns with $b =$ *false*.
      Let x_ℓ be the leader of g
      if $g \in \mathfrak{p}$ then
        Replace p_ℓ by g in $\overline{C}$
      else
        Replace p_ℓ by pquo(p_ℓ, g) in $\overline{C}$
      fi
      $k := 1$
    else
      $k := k + 1$
    fi
  od
  Denote $H_C = \{h_1, \ldots, h_m\}$
  $k := 1$
  while $k \leq m$ do
    $\overline{h}_k :=$ partial_rem $(h_k, \overline{C})$
    $(b,\ g) :=$ is_regular$(\overline{h}_k, \overline{C})$
    if b is *false* then

```
      Let x_ℓ be the leader of g
      if g ∉ 𝔭 then
        Replace p_ℓ by pquo(p_ℓ, g) in C̄
      else
        Replace p_ℓ by g in C̄
      fi
    else
      k := k + 1
    fi
  od
  return C̄
end
```

Over the example, it is sufficient to verify that the Δ–polynomial between the second and the third element of A is reduced to zero by A in order to prove that $A = 0,\ H_A \neq 0$ is a regular differential system (for the Δ–polynomial between the first and the second element of A has just been considered and the Δ–polynomial between the first and the third does not need to be considered by the analogue of Buchberger's second criterion). This verification done, regalise can be applied to G. All the elements of C are reduced to zero by A in the first loop. is_regular always returns pairs of the form (*true*, $\cdot$) in the second loop. The set $A = \overline{C}$ is returned.

Observe that it is actually not necessary to reset k to 1 in the if statement of the first loop. The proof of this claim is left to the reader.

Proposition 3.6: *The* regalise *algorithm terminates.*

Proof: Both loops carry out the mechanism whose termination is proven in proposition 1.6. □

Proposition 3.7: *The properties* **J1** *to* **J5** *are invariants of both loops of* regalise.

J1 *The set $\overline{C}$ is a squarefree partially autoreduced regular chain. It has the same set of leaders as A. It satisfies the relation: $(A) : H_A^\infty \subset (\overline{C}) : H_{\overline{C}}^\infty \subset \mathfrak{p} \cap R_0$.*

J2 *The ideal $(\overline{C}) : H_{\overline{C}}^\infty$ is radical. Its minimal primes are minimal over $(A) : H_A^\infty$.*

J3 *The system $\overline{C} = 0,\ H_{\overline{C}} \neq 0$ is a regular differential system.*

J4 *The differential ideal $[\overline{C}] : H_{\overline{C}}^\infty$ is radical. Its minimal differential primes are minimal differential primes of $[A] : H_A^\infty$.*

J5 *The set $\overline{C}$ is a characteristic set of the differential ideal $[\overline{C}] : H_{\overline{C}}^\infty$.*

Proof: Property **J1** comes from proposition 1.6. Invariant **J2** then follows from Lazard's lemma, the fact that A and $\overline{C}$ have the same set of leaders and the inclusion $(A) : H_A^\infty \subset (\overline{C}) : H_{\overline{C}}^\infty$.

To prove **J3** it suffices to prove that the system $\overline{C} = 0,\ H_{\overline{C}} \neq 0$ satisfies condition **C3**. For this, we consider some critical pair $\{p, p'\} \subset \overline{C}$ and prove it is reduced to zero by $\overline{C}$. Since $\overline{C}$ is a characteristic set of $(\overline{C}) : H_{\overline{C}}^{\infty}$, it suffices to show that $\mathsf{partial\_rem}\ (\Delta(p, p'), \overline{C})$ lies in that ideal. The ideal $(\overline{C}) : H_{\overline{C}}^{\infty}$ is the intersection of some of the minimal primes $\mathfrak{b}_1, \ldots, \mathfrak{b}_k$ of $(A) : H_A^{\infty}$ by **J2**. By theorem 3.3 applied to $A = 0,\ H_A \neq 0$, each $\mathfrak{b}_i$ is the intersection with R_0 of some prime differential ideal $\mathfrak{p}_i$. Thus $\Delta(p, p') \in \mathfrak{p}_i$. Thus the differential polynomial $\mathsf{partial\_rem}\ (\Delta(p, p'), \overline{C})$ lies in the ideals $\mathfrak{b}_i$ for $1 \leq i \leq k$ whence it lies in their intersection $(\overline{C}) : H_{\overline{C}}^{\infty}$. This concludes the proof of **J3**.

Invariant **J4** comes from **J3**, theorem 3.3 and the fact that the minimal primes of $(\overline{C}) : H_{\overline{C}}^{\infty}$ are minimal over $(A) : H_A^{\infty}$. Invariant **J5** comes from the fact that every differential polynomial lying in $[\overline{C}]{:}H_{\overline{C}}^{\infty}$ and reduced w.r.t $\overline{C}$ lies in $(\overline{C}){:}H_{\overline{C}}^{\infty}$ by **J3** and Rosenfeld's lemma. It must be zero for $\overline{C}$ is a characteristic set of this latter ideal. □

PROPOSITION 3.8: *Assume that, in the second loop,* $\mathsf{is\_regular}(\overline{h}_k, \overline{C})$ *returns a pair* (true, $\cdot$). *Then h_k is a non zero divisor modulo* $[\overline{C}] : H_{\overline{C}}^{\infty}$.

Proof: By **J3**, theorem 3.3 applies. Denote $\mathfrak{b}_1, \ldots, \mathfrak{b}_k$ the minimal primes of the radical ideal $(\overline{C}) : H_{\overline{C}}^{\infty}$ and $\mathfrak{p}_1, \ldots, \mathfrak{p}_k$ the minimal differential primes of the radical differential ideal $[\overline{C}]{:}H_{\overline{C}}^{\infty}$. Assume the call $\mathsf{is\_regular}(\overline{h}_k, \overline{C})$ returns a pair (*true*, $\cdot$). Then $\overline{h}_k$ is a non zero divisor modulo $(\overline{C}) : H_{\overline{C}}^{\infty}$ i.e. it lies in none of the $\mathfrak{b}$'s. Thus $\overline{h}_k$ lies in none of the $\mathfrak{p}$'s. Thus h_k does not either whence does not divide zero modulo $[\overline{C}] : H_{\overline{C}}^{\infty}$. □

The following proposition proves the claim stated in the first loop.

PROPOSITION 3.9: *Assume that, in the first loop,* $\mathsf{prem}(\overline{f}, \overline{C}) \neq 0$. *Then the call* $\mathsf{is\_regular}(\overline{f}, \overline{C})$ *returns a pair* (false, g).

Proof: Observe $\overline{f} \notin (\overline{C}) : H_{\overline{C}}^{\infty}$ for $\overline{C}$ is a characteristic set of that ideal (by **J1**). It is sufficient to show that $\overline{f}$ belongs to a prime ideal minimal over $(\overline{C}) : H_{\overline{C}}^{\infty}$. Since $A = 0,\ S \neq 0$ is a regular differential system, Rosenfeld's lemma applies and $[A] : S^{\infty} \cap R_0 = (A) : S^{\infty}$. Thus $(A) : S^{\infty} = \mathfrak{p} \cap R_0$ (for G satisfies **I1**). On one hand, the ideal $\mathfrak{p} \cap R_0$ is minimal over $(A) : H_A^{\infty}$ for it is obtained from it by saturation ; on the other hand, $(A) : H_A^{\infty} \subset (\overline{C}) : H_{\overline{C}}^{\infty} \subset \mathfrak{p} \cap R_0$. Thus $\mathfrak{p} \cap R_0$ is minimal over $(\overline{C}) : H_{\overline{C}}^{\infty}$ and since f and $\overline{C}$ lie in $\mathfrak{p}$ we have $\overline{f} \in \mathfrak{p} \cap R_0$. □

PROPOSITION 3.10: *The pseudocode of* $\mathsf{regalise}$ *satisfies the properties stated in proposition 3.5.*

Proof: According to loop invariant **J5**, it suffices to show $[\overline{C}] : H_{\overline{C}}^{\infty} = \mathfrak{p}$. The inclusion from left to right. Invariant **J1** proves that $\overline{C} \subset \mathfrak{p}$ and that the initials and separants of the elements of $\overline{C}$ do not lie in $(\overline{C}) : H_{\overline{C}}^{\infty}$ (for they are nonzero and reduced w.r.t. $\overline{C}$). They thus do not lie in $\mathfrak{p}$ using **J3** and Rosenfeld's lemma. Thus $[\overline{C}] : H_{\overline{C}}^{\infty} \subset \mathfrak{p}$. The converse inclusion. At the end of the first loop C is reduced to zero by $\overline{C}$. At the end of the second loop the initials and the separants of C are nonzero divisors modulo $[\overline{C}] : H_{\overline{C}}^{\infty}$ by proposition 3.8. Thus we have $\mathfrak{p} = [C] : H_C^{\infty} \subset [\overline{C}] : H_{\overline{C}}^{\infty}$. □

References

Philippe Aubry. *Ensembles triangulaires de polynômes et résolution de systèmes algébriques. Implantation en Axiom.* PhD thesis, Université Paris VI, 1999.

Philippe Aubry, Daniel Lazard, and Marc Moreno Maza. On the theories of triangular sets. *Journal of Symbolic Computation*, 28:105–124, 1999.

Thomas Becker and Volker Weispfenning. *Gröbner Bases: a computational approach to commutative algebra*, volume 141 of *Graduate Texts in Mathematics*. Springer Verlag, 1991.

François Boulier. A new criterion to avoid useless critical pairs in Buchberger's algorithm. Technical report, Université Lille I, 59655, Villeneuve d'Ascq, France, October 2001. (ref. LIFL 2001–07).

François Boulier. Efficient computation of regular differential systems by change of rankings using Kähler differentials. Technical report, Université Lille I, 59655, Villeneuve d'Ascq, France, November 1999. (ref. LIFL 1999–14, presented at the MEGA2000 conference).

François Boulier, Daniel Lazard, François Ollivier, and Michel Petitot. Representation for the radical of a finitely generated differential ideal. In *proceedings of ISSAC'95*, pages 158–166, Montréal, Canada, 1995.

François Boulier, Daniel Lazard, François Ollivier, and Michel Petitot. Computing representations for radicals of finitely generated differential ideals. Technical report, Université Lille I, LIFL, 59655, Villeneuve d'Ascq, France, 1997. (ref. IT306, december 1998 version published in the habilitation thesis of Michel Petitot).

François Boulier, François Lemaire, and Marc Moreno Maza. PARDI ! In *proceedings of ISSAC'01*, pages 38–47, London, Ontario, Canada, 2001a.

François Boulier and François Lemaire. Computing canonical representatives of regular differential ideals. In *proceedings of ISSAC 2000*, pages 37–46, St Andrews, Scotland, 2000.

François Boulier, François Lemaire, and Marc Moreno Maza. Well known theorems on triangular systems. Technical report, Université Lille I, 59655, Villeneuve d'Ascq, France, November 2001b. ref. LIFL 2001–09.

Driss Bouziane, Abdelillah Kandri Rody, and Hamid Maârouf. Unmixed–Dimensional Decomposition of a Finitely Generated Perfect Differential Ideal. *Journal of Symbolic Computation*, 31:631–649, 2001.

Lionel Ducos. Optimizations of the subresultant algorithm. *Journal of Pure and Applied Algebra*, 145:149–163, 2000.

Keith O. Geddes, Stephen R. Czapor, and George Labahn. *Algorithms for Computer Algebra*. Kluwer Academic Publishers, 1992.

Évelyne Hubert. Factorization free decomposition algorithms in differential algebra. *Journal of Symbolic Computation*, 29(4,5):641–662, 2000.

Mickael Kalkbrener. A Generalized Euclidean Algorithm for Computing Triangular Representations of Algebraic Varieties. *Journal of Symbolic Computation*, 15:143–167, 1993.

Donald Erwin Knuth. *The art of computer programming*. Addison–Wesley, 1966. Second edition.

Ellis R. Kolchin. *Differential Algebra and Algebraic Groups*. Academic Press, New York, 1973.

D. König. *Theorie der endlichen und unendlichen Graphen*. Chelsea publ. Co., New York, 1950.

Daniel Lazard. A new method for solving algebraic systems of positive dimension. *Discrete Applied Mathematics*, 33:147–160, 1991.

Henri Lombardi, Marie-Françoise Roy, and Mohab Safey El Din. New structure theorem for subresultants. *Journal of Symbolic Computation*, 29(4,5):663–690, 2000.

Marc Moreno Maza. On Triangular Decompositions of Algebraic Varieties. Technical report, NAG, 2000. (presented at the MEGA2000 conference).

Marc Moreno Maza and Renaud Rioboo. Polynomial gcd computations over towers of algebraic extensions. In *Proceedings of AAECC11*, pages 365–382. Springer Verlag, 1995.

Sally Morrison. Yet another proof of Lazard's lemma. private communication, december 1995.

Sally Morrison. The Differential Ideal $[P] : M^\infty$. *Journal of Symbolic Computation*, 28:631–656, 1999.

François Ollivier. *Le problème de l'identifiabilité structurelle globale : approche théorique, méthodes effectives et bornes de complexité*. PhD thesis, École Polytechnique, Palaiseau, France, 1990.

Michel Petitot. Quelques méthodes de Calcul Formel appliquéees à l'étude des équations différentielles, February 1999. Mémoire d'habilitation à diriger des recherches, Université Lille I, LIFL, 59655 Villeneuve d'Ascq, France.

Joseph Fels Ritt. *Differential Algebra.* Dover Publications Inc., New York, 1950.

Azriel Rosenfeld. Specializations in differential algebra. *Trans. Amer. Math. Soc.*, 90:394–407, 1959.

Brahim Sadik. Une note sur les algorithmes de décomposition en algèbre différentielle. *Comptes Rendus de l'Académie des Sciences*, 330:641–646, 2000.

Josef Schicho and Ziming Li. A construction of radical ideals in polynomial algebra. Technical report, RISC, Johannes Kepler University, Linz, Austria, august 1995.

Abraham Seidenberg. An elimination theory for differential algebra. *Univ. California Publ. Math. (New Series)*, 3:31–65, 1956.