General algorithms in asymptotics II
Common operations

by Joris van der Hoeven

LIX
École Polytechnique
91128, Palaiseau
France

Email: vdhoeven@lix.polytechnique.fr

July 1994

Abstract

We continue the study of automatic transseries, which has been started in the previous paper on general algorithms. We start by giving algorithms for expanding solutions to polynomial equations and then proceed with differentiation, integration, functional composition and inversion. Functions, constructed by a succession of these operations can also be handled.

Key words: Asymptotic expansion, transseries, algorithm, real field, differential calculus, composition, inversion.

1 Introduction

In [VdH 94] we have put Gonnet and Gruntz' algorithm to expand exp-log functions in a theoretical perspective. In this paper we will show that the technique used naturally extends to obtain automatic expansions of more general types of functions. Our results slightly generalize those of Shackell, which were obtained by his technique of nested expansions (see [Sh 91], [SalSh 92]). Furthermore, our approach is expected to be a little bit more efficient (at least, a constant factor should be gained). Before we run actual benchmarks, we think that in any case our algorithms are a bit simpler and we think that the obtained expansions are a bit more natural than nested expansions.

In section 2, we consider solutions to polynomial equations and we use the standard technique to expand them. In section 3, we consider derivatives and integrals. The "upward movement" technique (see [GoGr 92]) will be used to get rid of logarithms in the expansions. Then derivatives and integrals of standard expansions can easily be computed. Results are converted back by using "downward movement". Here we essentially use the fact that we expand w.r.t. normal and not w.r.t. weakly normal bases. Finally, in section 4, we show how to deal with functional composition and inversion and we will see that the methods used rely on the computation of derivatives from section 3.

We observe that in the cases of solving polynomial equations and integration, solutions are not unique. In the case of polynomials, there are only a finite number of solutions and we will show that is possible to select particular ones. Alternatively, one perform computations on the entire solution set, and we will speak about *generic transseries*. In the case of integration, these generic transseries depend on continuous parameters and constant problems arise, if we want to select a particular solution. We will not go into these problems here and refer to [VdH *a] and [VdH *b] for a more detailed discussion in a more general context.

Finally, we remark that we use the same notation as in [VdH 94] and we will refer to this paper by prefixing the numbers of theorems and such by I. We would like to thank J.M. Steyaert for many suggestions and the detection of a certain number of errors in previous versions of this article.

2 Polynomial equations

We will show how to expand *real algebraic exp-log functions*, which are functions build up with x, $\mathbb{Q}$, field operations, exponentiation, logarithm and solving real algebraic equations. As usual, we start by examining the case of equations in $K[\![x^A]\!]$, where A is a totally ordered abelian group. Moreover, we suppose that for each $\alpha \in A$ and $n \in \mathbb{N}^*$ there exists a β in A with $\alpha = n\beta$.

Theorem 1. *If K is a (real) algebraically closed field, then so is $K[\![x^A]\!]$.*

Proof. We will have to show, that given a polynomial equation $P(\varphi)$, with $P \in K[\![x^A]\!][\varphi]$, all (real) solutions of the equation lie in $K[\![x^A]\!]$. By considering the g.c.d. of P and $\partial P/\partial\varphi$, we may assume without loss of generality that P has no multiple roots. We adapt a classical argument from the theory of Puiseux series. Let us first deal with the case when A is Archimedian.

We will search solutions to the equation $P(\varphi) = 0$ of the form $\varphi = \psi + \varphi'$, where $\mu_{\varphi'} > \mu \geq \max \mathbf{supp}\, \psi$. Moreover, we will make the assumption that ψ is a *potential start* of order μ of a solution to $P(\varphi) = 0$. That is, if we consider the equation $P(\varphi) = 0$ as an equation

$$P'_n(\varphi')^n + \cdots + P'_0 = 0$$

in φ', there exists an i such that $\mu_{P'_j} + j\mu \geq \mu_{P'_i} + i\mu$, for each j, and where the inequality is strict when $j=0$. If this is the case, we define[1]

$$\mu' = \min\{\mu' > \mu \mid \exists i > j \forall k \quad \mu_{P'_k} + k\mu' \geq \mu_{P'_i} + i\mu' = \mu_{P'_j} + j\mu'\}$$

and $\alpha = \mu_{P'_i} + i\mu = \mu_{P'_j} + j\mu$, for corresponding i and j. We also define

$$(P'_n)_{\alpha - n\mu'}\lambda^n + \cdots + (P'_0)_\alpha = 0$$

to be the *dominant equation* associated to (ψ, μ). The equation is of the form $Q(\lambda)$ in K, where Q is the *dominant polynomial*, with coefficients in K. We claim three points:

(a) Each (real) solution λ to the dominant equation induces a new potential start $\psi' = \psi + \lambda x^{\mu'}$ of order μ' of a solution to $P(\varphi) = 0$.
(b) The degree of the dominant equation associated to this new potential start is equal to the multiplicity of the root λ.
(c) Whenever the dominant equation is of degree 1, then the potential start gives rise to a unique solution to P, which lies in $K[\![x^A]\!]$.

We first prove (a). Let

$$P''_n(\varphi'')^n + \cdots + P''_0 = 0$$

be the polynomial equation, rewritten in $\varphi'' = \varphi - \psi'$. We have

$$P''_j = P'_j + \cdots + \binom{n}{j} P'_n \lambda^{n-j} x^{(n-j)\mu'},$$

for each j and $\alpha = \mu_{P'_i} + i\mu'$ for some i, which we may suppose maximal. Therefore,

$$\mu_{P''_j} + j\mu' \geq \max_{j \leq k \leq n} \mu_{P'_k} + k\mu' \geq \mu_{P'_i} + i\mu',$$

for each j. Moreover, for $k > i$, we have $\mu_{P'_k} + k\mu' < \mu_{P'_i} + i\mu'$, so $\mu_{P''_i} = \mu_{P'_i}$. Finally, $\mu_{P''_0} > \mu_{P''_i} + i\mu$, since λ is a root of the dominant equation.

Next, we prove (b). Let Q be the dominant polynomial, which is of degree i (with i as in the proof of (a)). Then we observe that $(P''_j)_{\alpha - j\mu'} = (\partial^j Q / \partial \lambda^j)(\lambda)$. Let m be the multiplicity of the root λ. We deduce that $\mu_{P''_m} = \alpha - m\mu'$ and $\mu_{P''_j} > \alpha - j\mu'$ for $j < m$. Now let l and μ'' be such that

$$\mu_{P''_k} + k\mu'' \geq \mu_{P''_l} + l\mu'' = \mu_{P''_m} + m\mu'',$$

for $0 \leq k \leq m$. It is straigtforward to verify that the equation is verified for all k, that μ'' is minimal with the property that there exist l, m such that the above

[1]This definition can be interpreted graphically in terms of Newton polygons: suppose that we plotted the $\mu_{P'_i}$'s as a function of i and take the lower part of the convex hull of this graph. Then $-\mu'$ is the smallest slope bigger than $-\mu$.

inequality is verified for each k, and that m is maximal satisfying this property. Therefore m equals the degree of the dominant polynomial with respect to (ψ', μ').

Finally, let us prove (c). By dividing the P_i''s by $x^{\mu_{P_0'}}$, we may assume without loss of generality that $\mu_{P_1'} = 0$. Let $S = (\mathbf{supp}\, P_0' - \mu_{P_0'} \cup \cdots \cup \mathbf{supp}\, P_n' + (n-1)\mu_{P_0'})^{\diamond}$. Now if we repeat the procedure to compute new potential starts, by (b) the dominant equation remains linear. We claim that the dominant exponents will all be included in the finitely generated S, thus proving (c). Indeed, we already have $\mu_{P_0''} - \mu_{P_0'} \in S$. Next, for each $j \geq 1$, we have $\mathbf{supp}\, P_j'' + (j-1)\mu_{P_0'} \subseteq \bigcup_{0 \leq i \leq n-j} P_{j+i}'' + i\mu_{P_0'} \subseteq S + (j-1)(\mu_{P_0''} - \mu_{P_0'}) \subseteq S$. Similarly $\mathbf{supp}\, P_0'' - \mu_{P_0''} \in S$. Therefore $(\mathbf{supp}\, P_0'' - \mu_{P_0''} \cup \cdots \cup \mathbf{supp}\, P_n'' + (n-1)\mu_{P_0''})^{\diamond} \subseteq S$, and we apply an easy induction argument to conclude.

Now suppose that we have a chain $\psi, \psi', \psi'', \cdots$ of potential starts of orders $\mu, \mu', \cdots$, where each one is obtained from the previous one, in the way described above. We start with $\psi = 0$ and $\mu = -\infty$. We claim that the degrees of their associated dominant equations are ultimately equal to $d = 1$. Suppose that this is not the case, and that $d > 1$. Then we observe that each $\psi^{(i)}$ is also a potential start of some root to each of the $(\partial^j P / \partial \varphi^i)$'s, with $1 \leq j \leq d-1$. But this root is an element of $K[\![x^A]\!]$, because the degree of the dominant equation of $\psi^{(i)}$, for some large i and with respect to $\partial^{d-1} P / \partial \varphi^{d-1}$, is equal to 1. Therefore, P and P' have a common root, which contradicts our hypothesis. Finally, from our claim and (c) we deduce that the potential starts converge term by term to a root of P.

In the case of an algebraically closed field, the property (b) implies that there are exactly n distinct chains of potential starts such as above. These chains give rise to exactly n distinct roots of P. As $K[i]$ is algebraically closed if K is real algebraically closed, we also deduce that P admits exactly n solutions in $K[i][\![x^A]\!]$. Moreover each root φ of P which lies in $K[i][\![x^A]\!] \backslash K[\![x^A]\!]$ is imaginary, because $K[\![x^A]\!][\varphi] = K[i][\![x^A]\!]$. Therefore all real roots of P are also obtained as limits of chains of potential starts.

We finally have to prove the theorem, when A is not Archimedian. Now the reader may check that the coefficients $P_0, \cdots, P_n$ can be embedded in $K[\![x_1; \cdots; x_k]\!]$, for suitable $x_1, \cdots, x_k$, so that we only need to verify the theorem when $K[\![x^A]\!] = K[\![x_1; \cdots; x_k]\!]$ (in fact, this is the only case which interests us). Applying repeatedly what we have proved already, we know that $K[\![x_1]\!] \cdots [\![x_k]\!]$ is (real) algebraically closed. It is not hard to check that if the coefficients of the polynomial all have finitely generated support, then so have its solutions. ♡

Remark. The property (c) can also be seen as the result of some kind of implicit function theorem for multiseries. In later papers, where we will solve more complicated equations, we will see that similar properties can be deduced from generalisations of the implicit function theorem.

Let us now establish the effective version of the above theorem, which extends lemma I.1. We will make the same assumptions. A field $\mathfrak{K}$ is said to be an *algorithmic (real) algebraically closed field*, if $\mathfrak{K}$ is an algorithmic (ordered) field and if we can find the set of (real) solutions to a polynomial equation with coefficients in $\mathfrak{K}$ by algorithm. We then have the

Lemma 1. *If $\mathfrak{M}$ is an algorithmic (real) algebraically closed field, then so is $\mathfrak{M}_a$.*

Proof. The proof is similar to the proof of lemma I.1. We remark that we can indeed compute the g.c.d. of P and $\partial P/\partial\varphi$ by the euclidean algorithm in the algorithmic polynomial ring $\mathfrak{M}_a[\varphi]$. Moreover the property (b) ensures that the number of branches in the computation of the solution set (for different choices of solutions to the dominant equations) remains bounded by n. ♡

Remark. In the algorithmic context, the absence of multiple roots ensures that after a finite number of computations we are able to decide which potential starts give rise to solutions. In cases where two solutions have a lot of initial terms in common, this finite number might however be very big. The algorithm can then be accelerated by computing solutions to the equation $P(\psi+\varphi')=0$, where ψ is an appropriate root of $\partial P/\partial\varphi$. Alternatively, Sturm sequences can be used to isolate roots and to count their number.

Remark. By the lemma, we know how to compute the set of solutions to an algebraic equation in $\mathfrak{M}_a$. Algorithms that give the set of multiseries which are solution to some problem will be called *automatic generic multiseries*. Similarly, we have *automatic generic transseries*. However, we often want to fix some particular solution. In the present case, this implies that we have to rank the solutions using some numbering convention. There are two ways of doing this: numbering the $s \leq n$ solutions in increasing order, or numbering them loosely in increasing order by possibly inserting "illegal" solutions, which correspond to the imaginary solutions. For example, the equation $\varphi(\varphi^2-2\varphi+1+x^2)(\varphi-2)=0$ has $0, 1\pm ix$ and 2 as solutions. The first convention would give $\varphi_1=0$ and $\varphi_2=2$; the second one $\varphi_1=0$ and $\varphi_4=2$. From a computational point of view, the second one can be implemented efficiently, because we don't need to compute the entire set of solutions first, in order to compute a particular solution.

Let us now generalize theorem 2 to real algebraic exp-log functions. We proceed by representing real algebraic exp-log functions by *real algebraic exp-log expressions*, which are trees, whose leafs are labeled by x or rational numbers and whose nodes are labeled by the operators $+,-,\cdot,/,\exp,\ln$, or $(n+1)$-ary operators P_i, with $1\leq i\leq n$. If $f_0,\cdots,f_n$ are transseries, which can be represented by real algebraic exp-log expressions, then the i-th solution (using some numbering convention) to the equation

$$f_n\varphi^n + \cdots + f_0 = 0,$$

will be represented by $P_i(f_0, \cdots, f_n)$. Hence, by structural induction, we can represent all real algebraic exp-log functions. Moreover, this representation has the advantage that if we have expansions of the f_i's w.r.t. some common normal base B, then the preceding lemma implies that all solutions to the equation will also have expansions w.r.t. B.

We finally need to pay some attention to the constants problem. Shackell established a zero-equivalence algorithm for solutions to algebraic differential equations, modulo the algebraic exp-log constants problem (see [Sh 89b]) and modulo the possibility to find the corresponding initial conditions to a given problem. The algebraic exp-log constant problem has been solved by Richardson (see [Rich 92]). Next, we have to be able to find the corresponding initial conditions to a given solution to a polynomial equation. We will show in [VdH *b] how to do this. Under the above assumptions, we have the following extension of the expansion algorithm:

Extension to expand(f). Adds the case of an expression f whose root is labeled by P_i to the expansion algorithm.

case $f = P_i(f_1, \cdots, f_n)$: Compute standard expansions of $f_1, \cdots, f_n$ and apply lemma 1 to obtain a standard expansion of f.

As an application, we have

Theorem 2. *Schanuel's conjecture implies that the field of real algebraic exp-log functions is an automatic expansion field.* ♡

3 Differention and integration

The field of transseries can easily be given a derivation $'$ and a composition operator $\circ$, where $f \circ g$ is defined, whenever $g \succ\!\!\succ 1$ and $g > 0$. We leave it as an exercise to the reader to include these definitions in a natural way in the definition of transseries (see section I.4). Definitions are also given in [Ec 92]. Still alternatively, by using abstract algorithmic arguments, the reader can verify that the algorithms given in this and the next section can also serve as definitions. Finally, we will denote by $\hat{f}$ the logarithmic derivative f'/f of f.

Before coming to algorithms, we discuss the principle of *upward* and *downward movements* or *shiftings*. If f is a transseries for which we computed a standard expansion, then we claim that we can easily obtain standard expansions for $f \circ \ln_n$ and $f \circ \exp_n$, for any $n \in \mathbb{N}$ (here $\exp_n = \exp \circ \overset{n\times}{\cdots} \circ \exp$). Indeed, it suffices to show this for $n = 1$. Now if B is a normal base, we can shift this base upwards, by replacing each element $g \in B$ by $g \circ \exp$ (of course, we replace $\ln_{n+1} \exp x$ by $\ln_n x$). Then we observe that this yields a normal base for $f \circ \exp$ (possibly we will have to insert x in the shifted base). Similarly, we can shift B downwards, which yields a base for $f \circ \ln$. We will note $f{\uparrow}_n = f \circ \exp_n$ and $f{\downarrow}_n = f \circ \ln_n$. Similarly, we define $B{\uparrow}_n$ and $B{\downarrow}_n$.

Now let us show why this principle is very usefull, when doing automatic asymptotic differential calculus. Consider for example $f_1 = 1/(1 - e^{-x})$ and $f_2 = 1/(1 - \ln^{-1} x)$. The derivative of f_1 can be computed in a straightforward way, by expanding f_1 in e^x and then differenting componentwise. In the case of f_2 this doesn't work, because the derivative should first be expanded in x, due to the logarithm occuring in f_2. Now the technique of shifting can be used to avoid this second possibility. Indeed, if $\ln_n x$ were the smallest element (w.r.t. $\lll$) occuring in the standard expansion of some f (that is the smallest element of a normal subbase of B w.r.t. which f can be expanded), then shifting f and B upwards n times eliminates all occurences of ln in the expansion of f. Furthermore, the derivatives of $f, f{\uparrow}_n$ and $f{\downarrow}_n$ are related by $(f{\uparrow}_n)' = E_n f'{\uparrow}_n$ and $(f{\downarrow}_n)' = L_n f'{\downarrow}_n$, where

$$\begin{cases} E_n(x) = \exp x \cdots \exp_n x \\ L_n(x) = \dfrac{1}{x \cdots \ln_{n-1} x}. \end{cases}$$

When doing more complex differential calculus, further shifting need sometimes to be done. For instance, when computing integrals, the integration of $1/x$ introduces logarithms. However, if $\ln_{n-1}$ is smallest element occuring in the standard expansion of some f, then $f{\uparrow}_n$ can be expanded w.r.t. $g_1 \lll \cdots \lll g_k$, with $x \lll g_1$, and no logarithms will be introduced when integrating f. By convention, n may be equal to 1. In [VdH *a] we will see that even further upward shifting may be necessary when dealing with algebraic differential equations.

We now give algorithms to compute derivatives and integrals. Of course, it is necessary to suppose that $\mathfrak{T}$ is algorithmically stable under differention and integration. We refer to [Sh 89], where it is proved that the set of functions verifying an algebraic differential equation over $\mathbb{Q}$ is an algorithmic field stable under differention and integration, under the restriction that the initial conditions are given w.r.t. an algorithmic subfield of $\mathbb{R}$. In order to integrate expansions we will need to solve Risch differential equations, that is equations of the type $f' = f_1 f + f_2$. We will note by $I(f_1, f_2)$ a solution to this equation; in particular $\int f_2 = I(0, f_2)$. Moreover, if f_1 and f_2 are not constants, we require that $I(f_1, f_2)$ can be expanded w.r.t. $x \lll g_1 \lll \cdots \lll g_k$, if g_k is the biggest element of B occuring in the expansions of f_1 and f_2. For instance, $f = e^{e^x}$ is an illegal solution to the equation $f' = e^x f$. Finally, we refer to the introduction for a discussion on the selection of particular integration constants for $I(f_1, f_2)$.

Extension to expand(f). Adds the case of an expression f whose root is labeled by $'$ or I to the expansion algorithm.

case $f = f_1'$: If f_1 is a constant, then $f = 0$. Else, compute a standard expansion of f_1 and determine the smallest iterated logarithm $\ln_{n-1} x$ occuring in the expansion of f_1. Let g be the biggest element of $B\uparrow_n$ occuring in the expansion of $f_1\uparrow_n$. Recursively expand $\varphi = f_1\uparrow_n'$ w.r.t. g^{-1} by computing the

$$\varphi_\alpha = ((f_1\uparrow_n)_\alpha)' - \alpha(f_1\uparrow_n)_\alpha \hat{g}.$$

Finally, we translate the obtained expansion for φ back using the identity $f = L_n \varphi\downarrow_n$.

case $f = I(f_1, f_2)$: If f_1 and f_2 are constants, then $f = f_2 x$, if $f_1 = 0$, and $f = -f_2/f_1$, if $f_1 \neq 0$. Else, compute standard expansions of f_1 and f_2 and determine the smallest iterated logarithm $\ln_{n-1} x$ occuring in these expansions. Put $\varphi_1 = E_n f_1\uparrow_n$ and $\varphi_2 = E_n f_2\uparrow_n$. Let g be the biggest element of $B\uparrow_n$ occuring in the expansions of φ_1 and φ_2. Recursively expand a solution of the equation $(f\uparrow_n)' = \varphi_1 f\uparrow_n + \varphi_2$ w.r.t. g^{-1} using either

$$(f\uparrow_n)_\alpha = I\left(\varphi_{1,0} + \alpha\hat{g}, \varphi_{2,\alpha} + \sum_{\beta>0} \varphi_{1,\beta}(f\uparrow_n)_{\alpha_\beta}\right),$$

if $\mu_{\varphi_1} \geq 0$, or

$$(f\uparrow_n)_\alpha = \frac{-1}{\varphi_{1,\mu_{\varphi_1}}}\left(\varphi_{2,\alpha+\mu_{\varphi_1}} + (\alpha + \mu_{\varphi_1})\hat{g}(f\uparrow_n)_{\alpha+\mu_{\varphi_1}} - (f\uparrow_n)'_{\alpha+\mu_\varphi} + \sum_{\beta>\mu_{\varphi_1}} \varphi_{1,\beta}(f\uparrow_n)_{\alpha-\beta}\right),$$

if $\mu_{\varphi_1} < 0$. In these two cases we take $(f\uparrow_n)_\alpha = 0$, for $\alpha < \mu_{\varphi_2}$, resp. $\alpha < \mu_{\varphi_2} - \mu_{\varphi_1}$. Finally, $f\uparrow_n\downarrow_n$ yields the expansion of f.

The algorithm is justified by the fact that as $B\uparrow_n$ is a normal base, and if $g = e^h \in B$, then $\hat{g} = h'$ can be expanded w.r.t. elements of B which are strictly smaller than g. Indeed, this can be seen by using a simple induction argument over the cardinality of $B\uparrow_n$. We remark that in the case of the derivative no new elements need to be inserted in B, (if, by coincidence, x had to be inserted in $B\uparrow_n$, we may remove it, when shifting back).

4 Functional composition and inversion

Let us now treat functional composition and inversion. As usual, we suppose that $\mathfrak{T}$ is algorithmically stable under composition and functional inversion. We claim that this is the case, if $\mathfrak{T}$ is the subfield of $\mathbb{R}[\![x]\!]$ of functions verifying an algebraic differential equation over $\mathbb{Q}$, with real algebraic exp-log initial conditions (modulo Schanuel's conjecture).

Suppose first that f and g are in $\mathfrak{T}$, with $g>0$ and $g \succ\!\!\succ 1$. Then the family $f\circ g, (f\circ g)' = g'(f'\circ g), \cdots$ has finite algebraic trancendence degree and for example by Gröbner basis techniques one can find a non trivial algebraic differential equation verified by $f\circ g$. Alternatively, one observes that the $(f^{(i)}\circ g)$'s are given by $f^{(i)}\circ g = P_i((f\circ g)', \cdots, (f\circ g)^{(i)}, g, \cdots, g^{(i)})/(g')^{2i+1}$, for certain polynomials P_i. We get an algebraic differential equation for $f\circ g$, by composing the equation for f with g and then substituting the $(f^{(i)}\circ g)$'s by the above formulas. The obtained equation has coefficients in $\mathbb{Q}[g, g', \cdots]$, but can be transformed to yield one over $\mathbb{Q}$, by using the algebraic differential equation for g.

Suppose now that $g\circ f = x$, with $g>0$ and $g \prec\!\!\prec 1$. Then the $g^{(i)}$'s are rational fractions in the $(f^{(i)}\circ g)$'s. We can therefore compute an algebraic relation between the $(f^{(i)}\circ g)$'s and g. Right composition by f yields an algebraic relation between the $f^{(i)}$'s and x. From this equation it is easy to obtain a differential algebraic equation for f.

As to the initial conditions, it suffices to choose a "good" exp-log constant x_0 as our point of analyticity. In the case of composition, g should be analytic at x_0 and f at $g(x_0)$. In the case of inversion, x_0 should be the image of a regular point of analycity of g. It therefore suffices to take x_0 sufficiently large. In practice, randomly chosen small x_0 should (almost always) do as well. Finally, the initial conditions are given by exp-log constants, whenever the initial conditions for (f and) g are. In particular, this is the case when (f and) g are constructed by the operations considered so far, with the exception of integration.

Extension to expand(f). Adds the case of an expression f whose root is labeled by $\hat{}$, $\circ$, or ^{inv} to the expansion algorithm.

Case $f = f_1^c$ (where c is a constant): Put the standard expansion of f_1 in the form $f_1 = (f_{1,\mu} + \sum_{\alpha>0} f_{1,\alpha}\psi^{\mu-\alpha})\psi^\mu$. Then we get an expansion of f by extracting coefficients from

$$f^c = (f_{1,\mu} + \sum_{\alpha>0} f_{1,\alpha}\psi^{\mu-\alpha})^c\psi^{\mu c}.$$

The $f_{1,\mu}^c$ are computed recursively.

Case $f = f_1\circ f_2$: Compute standard expansions of f_1 and f_2. Let us show first that without loss of generality, we may assume that no logarithm occurs in the expansion of f_1. This can be done by determining the smallest iterated logarithm $\ln_k$ occuring in the expansion of f_1. Then shift f_1, f_2 and B upwards k times. We finally compute $f_1\circ f_2 = (f_1{\uparrow_k} \circ \ln_k f_2{\uparrow_k}){\downarrow_k}$.

Determine the elements $g_1 \prec\!\!\prec\!\!\prec \cdots \prec\!\!\prec\!\!\prec g_n$ of B, occuring in f_1. Recursively compute standard expansions of the $(g_i\circ f_2)$'s (a "remember option" is used for this computation), by $g_i\circ f_2 = \exp(h_i\circ f_2)$, where $g_i = \exp h_i$. We remark that although no logarithms may occur in the expansions of the g_i's, they might occur in B itself. Let ψ be the biggest element occuring in the expansions of f_2 and the $(g_i \circ f_2)$'s. We distinguish two cases:

If $g_n \circ f_2 \asymp \psi$, we put the standard expansion of $g_n \circ f_2$ in the form $g_n \circ f_2 = ((g_n \circ f_2)_\mu + \sum_{\alpha > 0} (g_n \circ f_2)_{\alpha - \mu} \psi^{\mu - \alpha}) \psi^{-\mu}$, with $\mu > 0$. Let $f_1 = \sum_\beta f_{1,\beta} g_n^\beta$ be the standard expansion of f_1 w.r.t. g_n (thus, exceptionally, $f_{1,\beta}$ is not the coefficient of $\psi^{-\beta}$ in f_1). Then the standard expansion of f is obtained by extracting coefficients from the formula

$$f_1 \circ f_2 = \sum_\beta \left(\sum_\alpha (f_{1,\beta} \circ f_2)_\alpha \psi^\alpha \right) \left((g_n \circ f_2)_\mu + \sum_{\alpha > \mu} (g_n \circ f_2)_\alpha \psi^{\mu - \alpha} \right)^\beta \psi^{-\mu\beta}.$$

If $g_n \circ f_2 \prec\!\!\prec\!\!\prec \psi$, we have necessarily $\mu_{f_2} = 0$, because g_n is bigger or equal to x for $\prec\!\!\prec\!\!\prec$. Decompose $f_2 = f_{2,0} + \varepsilon$. Next compute $g_i \circ f_{2,0} = (g_i \circ f_2)_0$, for each i. Then the standard expansion of f is obtained by extracting coefficients from

$$f_1 \circ f_2 = f_1 \circ f_{2,0} + f_1' \circ f_{2,0}\, \varepsilon + \cdots.$$

Case $f = \varphi^{inv}$: Compute a standard expansion of f and let $g \in B$ be such that $\varphi \asymp g$. If $g = e^h$ then compute f by $f = (\ln \circ \varphi)^{inv} \circ \ln$. If $g = \ln_k x$, with $k > 0$, then compute f by $f = \exp_k \circ (\varphi \circ \exp_k)^{inv}$. Finally, the case $g = x$ is reduced to the case, where $\varphi = x + \varepsilon$, with $\varepsilon \prec\!\!\prec 1$ by using the formula $\varphi^{inv} = \exp \circ (x/c \circ \ln \circ \varphi \circ \exp)^{inv} \circ x/c \circ \ln$, for suitable c. Next, determine the smallest iterated logarithm $\ln_k$ occuring in the expansion of φ. We then reduce the general case to the case when $k = 0$, by writing $f = \exp_k \circ (\ln_k \circ \varphi \circ \exp_k)^{inv} \circ \ln_k$. Let ψ be the biggest element of B occuring in the expansion of φ. Put $\varphi = \varphi_0 + \varepsilon$. Then we recursively expand f w.r.t. ψ by extracting coefficients from

$$x = \sum_{\alpha \geq 0} (f_\alpha \circ \varphi_0 + f_\alpha' \circ \varphi_0 \varepsilon + \cdots)(\psi \circ \varphi_0 + \psi' \circ \varphi_0 \varepsilon + \cdots)^{-\alpha}.$$

In particular, $f_0 = \varphi_0^{inv}$.

Let us justify the composition algorithm. We will prove its termination by induction over the pair $(|B|, n)$, where $|B|$ designates the cardinality of B. We claim that after having computed the $(g_i \circ f_2)$'s, no new elements have to be inserted into B, during the remaining part of the computations. This holds, under the condition that all constant powers of functions are expanded using the special algorithm we have given for this case. We have to consider the two cases $g_n \circ f_2 \asymp \psi$ and $g_n \circ g_2 \prec\!\!\prec\!\!\prec \psi$.

In the first case, the $f_{1,\beta} \circ f_2$ can be computed by reccurence, because $f_{1,\beta}$ can be expanded w.r.t. $g_1, \cdots, g_{n-1}$ and our claim holds by induction, because $g_1 \circ f_2, \cdots, g_{n-1} \circ f_2$ have already been computed (and remembered). But then the coefficients in the expansion of f are linear combinations of products of the $(f_{1,\beta} \circ f_2)_\alpha$ and the $(g_n \circ f_2)_\alpha$ times constant powers of $(g_n \circ f_2)_\mu$. These can clearly be computed without inserting new elements into B.

In the second case, it is easy to verify the formula $g_i \circ f_{2,0} = (g_i \circ f_2)_0$ by induction. Therefore no new elements have to be inserted into B for their computation. Moreover, $f_{2,0}$ and the $(g_i \circ f_{2,0})$'s can all be expanded w.r.t. $B - \{\psi\}$. By the remark at the end of the last section, it follows also that the $f_1^{(i)}$'s can be expanded w.r.t. $g_1, \cdots, g_n$. Therefore, the recursive application of expand is justified.

5 References

[GoGr 92] G.H. Gonnet, D. Gruntz. Limit computation in computer algebra. *Technical report 187, ETH. Zürich.*

[Lang 84] S. Lang. *Algebra.* Addison-Wesley, 2-ième éd. (p 390-403).

[Rich 92] D. Richardson. The elementary constant problem. *Proc. ISSAC 92 (p 108-116).*

[SalSh 92] B. Salvy, J. Shackell. Asymptotic expansions of functional inverses. *Technical report nr. 1673, INRIA, France.*

[Sh 89] J. Shackell. Zero-equivalence in function fields defined by algebraic differential equations. *Preprint.*

[Sh 91] J. Shackell. Limits of Liouvillian functions. *Technical report, University of Kent, Canterbury, England.*

[VdH 94] J. van der Hoeven. General algorithms in asymptotics I. Gonnet and Gruntz's algorithm. *Research report LIX/RR/94/10, Ecole Polytechnique, France.*

[VdH *a] J. van der Hoeven. General algorithms in asymptotics III. Algebraic differential equations. *In preparation**.

[VdH *b] J. van der Hoeven. General algorithms in asymptotics IV. Comparing constants. *In preparation**.

* Will appear as a research report at the Ecole Polytechnique.