Algorithms for asymptotic extrapolation*

Joris van der Hoeven

CNRS, Département de Mathématiques
Bâtiment 425
Université Paris-Sud
91405 Orsay Cedex
France

Email: joris@texmacs.org
Web: http://www.math.u-psud.fr/~vdhoeven

December 9, 2008

Consider a power series $f \in \mathbb{R}[[z]]$, which is obtained by a precise mathematical construction. For instance, f might be the solution to some differential or functional initial value problem or the diagonal of the solution to a partial differential equation. In cases when no suitable method is beforehand for determining the asymptotics of the coefficients f_n, but when many such coefficients can be computed with high accuracy, it would be useful if a plausible asymptotic expansion for f_n could be guessed automatically.

In this paper, we will present a general scheme for the design of such "asymptotic extrapolation algorithms". Roughly speaking, using discrete differentiation and techniques from automatic asymptotics, we strip off the terms of the asymptotic expansion one by one. The knowledge of more terms of the asymptotic expansion will then allow us to approximate the coefficients in the expansion with high accuracy.

Keywords: extrapolation, asymptotic expansion, algorithm, guessing

A.M.S. subject classification: 41A05, 41A60, 65B05, 68W30

1. Introduction

Consider an infinite sequence $f_0, f_1, \ldots$ of real numbers. If $f_0, f_1, \ldots$ are the coefficients of a formal power series $f \in \mathbb{R}[[z]]$, then it is well-known [Pól37, Wil04, FS96] that a lot of information about the behaviour of f near its dominant singularity can be obtained from the asymptotic behaviour of the sequence $f_0, f_1, \ldots$. However, if f is the solution to some complicated equation, then it can be hard to compute the asymptotic behaviour using formal methods. On the other hand, the coefficients $f_0, f_1, \ldots$ of such a solution f can often be computed numerically up to a high order [vdH02a]. This raises the question of how to *guess* the asymptotic behaviour of $f_0, f_1, \ldots$, based on this numerical evidence.

Assume for instance that we have fixed a class $\mathcal{E}$ of "admissible asymptotic extrapolations", such as all expressions of the form

$$\varphi = \left(c_0 + \cdots + \frac{c_k}{n^k}\right) n^{\alpha} \mathrm{e}^{\beta n^{\gamma}}. \tag{1}$$

Given the first coefficients $f_0, \ldots, f_N$ of f, the problem of *asymptotic extrapolation* is to find a "simple" expression $\varphi \in \mathcal{E}$, which approximates $\tilde{f}$ well in the sense that the *relative error*

$$\varepsilon_{\tilde{f},\varphi} = \max_{k \in \{L,\ldots,N\}} \frac{|f_n - \varphi_n|}{|f_n| + |\varphi_n|} \tag{2}$$

*. This work has partially been supported by the ANR Gecko project.

is small. Here $L < N$ is a suitably chosen number, such as $L = N/2$. In general, we are just interested in a good extrapolation formula, and not necessarily in the best one in the class $\mathcal{E}$. It is also important that the extrapolation continues to provide good approximations for $n \gg N$, even though we have no direct means of verification.

A good measure for the complexity of an expression $\varphi \in \mathcal{E}$ is its number ν_{φ} of *continuous parameters*. For instance, taking φ as in the formula (1), the continuous parameters are $c_0, \ldots, c_k$, α, β and γ, whence $\nu_{\varphi} = k + 4$. Another possible measure is the *size* σ_{φ} of φ as an expression. For our sample expression φ from (1), this yields $\sigma_{\varphi} = 9 + 6\,k$, when counting 1 for each of the operations $+$, $\times$, $/$, ^, exp, as well as 1 for n and each of the constants. Since the expression size σ_{φ} depends on the way operations are encoded, the measure ν_{φ} should generally be preferred. Notice also that one usually has $\sigma_{\varphi} \leqslant C\,\nu_{\varphi}$ for a fixed constant C.

In fact, the systematic integration of "guessing tools" into symbolic computation packages would be a useful thing. Indeed, current systems can be quite good at all kinds of formal manipulations. However, in the daily practice of scientific discovery, it would be helpful if these systems could also detect hidden properties, which may not be directly apparent or expected, and whose validity generally depends on heuristics. Some well-known guessing tools in this direction are the LLL-algorithm [LLL82] and algorithms for the computation of Padé-Hermite forms [BL94, Der94], with an implementation in the Gfun package [SZ94]. Padé-Hermite forms are used by Gfun in order to guess closed form formulas for sequences of which only a finite number of coefficients are known.

In the area of numerical analysis, several algorithms have been developed for accelerating the convergence of sequences, with applications to asymptotic extrapolation. One of the best available tools is the E-algorithm [Wen01, BZ91], which can be used when $\mathcal{E}$ consists of linear expressions of the form

$$\varphi = c_1\,\psi_1 + \cdots + c_k\,\psi_k, \tag{3}$$

with continuous parameters c_1, ..., c_k, and where $\psi_1 \succ \cdots \succ \psi_k$ are general functions, possibly subject to some growth conditions at infinity. Here we use the standard notations $f \prec g \Leftrightarrow f = o(g)$, $f \preccurlyeq g \Leftrightarrow f = O(g)$ and $f \asymp g \Leftrightarrow f \preccurlyeq g \preccurlyeq f$. Notice that the E-algorithm can sometimes be used indirectly: if $\gamma = 1$ in (1), then

$$\log \varphi = \beta\,n + \alpha \log n + c_0' + \cdots + \frac{c_k'}{n^k} + \cdots,$$

with $c_0' = \log c_0$, $c_1' = c_1 / c_0$, etc., is asymptotically of the required form (3).

We are aware of little work beyond the identification of parameters which occur linearly in a known expression. A so called "Gevreytiseur" is under development [CDFT01] for recognizing expansions of Gevrey type and with an asymptotic behaviour

$$f_n = C\,n^{\alpha}\,\mathrm{e}^{\beta n}\,(n!)^{\gamma} + \cdots. \tag{4}$$

One of the referees also made us aware of work [JG99, CGJ+05] on extrapolations of the form

$$\varphi = c_1\,n^{\alpha_1} + \cdots + c_k\,n^{\alpha_k}, \qquad (\alpha_1 > \cdots > \alpha_k)$$

where, in addition to the coefficients $c_1, \ldots, c_k$, some of the exponents α_i may be unknown (even though guesses often exist, in which case one only has to determine whether the corresponding coefficient c_i vanishes).

In this paper, we will be concerned with the case when $\mathcal{E}$ is the set of *exp-log expressions*. An exp-log expression is constructed from the rational numbers, real parameters and n using field operations, exponentiation and logarithm. It was already pointed out by Hardy that the asymptotic behaviour of many functions occurring in analysis, combinatorics and number theory can be expressed in asymptotic scales formed by exp-log functions [Har10, Har11, Sha90, Sal91, RSSvdH96]. More generally, the field of transseries (which can be seen as the "exp-log completion" of the field of germs of exp-log functions at infinity) enjoys remarkable closure properties [vdH06c, Éca92].

The main problem with naive approaches for asymptotic extrapolation is that, even when we have a good algorithm for the determination of the dominant term τ of the expansion of f_n, it makes little sense to recurse the algorithm for the difference $f_n-\tau$. Indeed, a small error in the determination of C, α, β and γ for an expansion of type (4) will result in a difference $f_n-\tau$ which is either asymptotic to f_n or τ. Even if we computed $f_0,\ldots,f_N$ with great accuracy, this will be of little help with current extrapolation techniques. Without further knowledge about the remainder $R_n=f_n-C\,n^{\alpha}\,\mathrm{e}^{\beta n}\,(n!)^{\gamma}$, it is also hard to obtain approximations for C, α, β, γ with relative errors less than R_n/f_n, no matter which classical or clever numerical algorithm is used.

Instead of directly searching the dominant term τ of the asymptotic expansion of f and subtracting it from f, a better idea is to search for a transformation $f\mapsto\Phi(f)$ which kills the dominant term, and recursively apply asymptotic extrapolation to the transformed sequence $\Phi(f)$. For instance, if $f_n=c_0+c_1\,n^{-1}+\cdots$, then we may take

$$f_n^{[1]}=\Phi(f)_n=\Delta(f)_n=f_{n+1}-f_n=-\frac{c_1}{n^2}+\cdots$$

as our first transformation and $f^{[1]}\mapsto\Delta(n^2\,f^{[1]})$ for the second transformation. Similarly, if $f_n=c_0\,\mathrm{e}^{c_1 n+\cdots}$, then we may take

$$f_n^{[1]}=\Phi(f)_n=Q(f)_n=\log\left(\frac{f_{n+1}}{f_n}\right)=c_1+\cdots$$

and proceed with the transformation Δ. After a finite number of transformations

$$f^{[k]}=(\Phi_k\circ\cdots\circ\Phi_1)(f),$$

the resulting sequence $f^{[k]}$ looses its accuracy, and we brutally extrapolate it by zero

$$f^{[k]}\approx\tilde{f}^{[k]}=0.$$

Following the transformations in the opposite way, we next solve the equations

$$\Phi_i(\tilde{f}^{[i-1]})=\tilde{f}^{[i]}\tag{5}$$

for $i=k,\ldots,1$. The resolution of (5) involves the numeric determination of the continuous parameter c_i (or parameters $c_{i,1},\ldots,c_{i,r_i}$, in general) which was killed by Φ_i. For decreasing $i=k,\ldots,1$ the accuracy of the so-computed parameters c_i and extrapolations $\tilde{f}^{[i]}\approx f^{[i]}$ tends to increase. The end-result $\tilde{f}^{[0]}$ is the desired extrapolation of f.

The above approach is actually a "meta-algorithm" in the sense that we have not yet specified the precise transformations which are applied. In section 2, we will discuss in more detail some of the "atomic" transformations which can be used. Each atomic transformation comes with a numeric criterion whether it makes sense to apply it. For instance, the finite difference operator Δ should be used on sequences which are not far from being constant. In section 3, we next describe our main asymptotic extrapolation algorithm, based on the prior choice of a finite number of atomic transformations.

A priori, the main algorithm either returns the extrapolation in a numeric form or as a sequence of inverses of atomic transformations. However, such extrapolations are not necessarily readable from a human point of view. In [vdH08], it is shown how to solve the equations (5) in the field of formal transseries. Finite truncations of these transseries then yield exp-log extrapolations for f, as will be detailed in section 4. Once the shape of a good exp-log extrapolation is known, the continuous parameters may be refined *a posteriori* using iterative improvements.

The new extrapolation algorithms have been implemented (but not yet documented) in the Mathemagix system [vdH02b]. A more manual version of the algorithm has been used extensively in [PMFB06, PF07, Pau07] and produced satisfactory extrapolations in this context. In section 5, we will present some examples on which we have tested our implementation. Of course, problems occur when the atomic transformations and corresponding numeric criteria have not been chosen with care. In section 6, we examine such problems and possible remedies in more detail. In the last section 7, we outline some possible generalizations of the algorithm.

Combining both *ad hoc* and classical techniques, such as the E-algorithm, experts will be able to obtain most of the asymptotic extrapolations considered in this paper by hand. Indeed, the right transformations to be applied at each stage are usually clear on practical examples. Nevertheless, we believe that a more systematic and automatic approach is a welcome contribution. Despite several difficulties which remain to be settled, our current algorithm manages to correctly extrapolate sequences with various forms of asymptotic behaviour. Our algorithm also raises some intriguing problems about appropriate sets of atomic transformations and the approximation of sequences with regular behaviour at infinity by linear combinations of other sequences of the same kind.

Remark 1. The present article is a completely reworked version of [vdH06a]. We adopted a more systematic exposition, centered around the notion of atomic transformations. We also optimized our propositions for good sets of atomic transformations and implemented the exp-log extrapolation algorithm.

2. Atomic transformations

As explained in the introduction, our main approach for finding an asymptotic extrapolation for f_n is through the application of a finite number of *atomic transformations* Φ on the sequence f_n. Each time that we expect f_n to satisfy a specific kind of asymptotic behaviour, we apply the corresponding atomic transformation. For instance, if we expect f_n to tend to a constant, which corresponds to an asymptotic behaviour of the form $f_n = c + R_n$ with $R_n \prec 1$, then we may apply the finite difference operator Δ.

More generally, we are thus confronted to the following subproblem: given a suspected asymptotic behaviour

$$f_n = \varphi_{c_1,\ldots,c_r}(R_n), \tag{6}$$

where $c_1, \ldots, c_r$ are continuous parameters and R_n is a remainder sequence, we should

1. Find a numeric criterion for deciding whether f_n potentially admits an asymptotic behaviour of the form (6), or very likely does not have such a behaviour.

2. Find a sequence transformation $g = \Phi(f)$, such that g only depends on R_n, but not on the continuous parameters $c_1, \ldots, c_r$.

3. Find a way to numerically solve the equation $\Phi(\tilde{f}) = \tilde{g}$, when a numerical extrapolation $\tilde{g} \approx g$ for g is known.

In this section, we will solve this subproblem for several basic types of asymptotic behaviour (6). In the next section, we will show how to combine the corresponding atomic transformations into a full asymptotic extrapolation algorithm. If a sequence f_n satisfies the numeric criterion associated to the atomic transformation Φ, then we say that Φ is *acceptable* for f_n.

Finite limits. When f_n tends to a finite limit c, we have

$$f_n = c + R_n \qquad (R_n \prec 1). \tag{7}$$

A simple numerical criterion for this situation is

$$\varepsilon_{f, f_N} < \delta_1 \tag{8}$$

for a suitable threshold $\delta_1 > 0$. The continuous parameter c is killed by the finite difference operator Δ:

$$g_n = \Delta(f)_n = f_{n+1} - f_n = R_{n+1} - R_n.$$

Given an asymptotic extrapolation $\tilde{g} \approx g$ for g, we may extrapolate f by

$$\begin{aligned} \tilde{f}_n &\approx a + \sum_{0 \leqslant i < n} g_i \\ a &= f_N - \sum_{0 \leqslant i < N} g_i \end{aligned}$$

In compact form, we thus have $\tilde{f} = \Delta_{f_N;N}^{-1}(\tilde{g})$, where $\Delta_{\nu;N}^{-1}$ is the inverse of Δ for which the N-th coefficient is given by ν.

Explicit dominant behaviour. Assume that we expect f_n to be of the form

$$f_n = R_n \psi_n \qquad (R_n \asymp 1), \tag{9}$$

where ψ_n is a simple explicit function, such as $\psi_n = n$, $\psi_n = n^{-1}$ or $\psi_n = n^{-2}$. A simple numeric criterion for this situation is

$$\varepsilon_{f, \frac{f_N}{\psi_N}\psi} < \delta_\psi \tag{10}$$

for a suitable threshold $\delta_\psi > 0$. We usually take $\delta_\psi < \delta_1$. More generally, δ_ψ is taken smaller for more complicated functions. For instance, one might take $\delta_n = 0.1$, but $\delta_{\sqrt{n}} = 0.025$. Whenever f_n is of the form (9), it is natural to apply the scaling operator $\times_{\psi^{-1}}: f \rightarrow f/\psi$ to f. Given an extrapolation $\tilde{g}$ for $g_n = f_n/\psi_n$, we simply scale back $\tilde{g} \mapsto \psi\, \tilde{g}$ in order to obtain the extrapolation for f.

Other regular behaviour at infinity. In cases when f_n has a regular asymptotic behaviour, which does not fall into any special case, then we may write

$$f_n = \pm\, \mathrm{e}^{R_n} \qquad (R_n \succ 1).$$

The only necessary condition for this to work is that the sign of f_n ultimately does not change. The corresponding numeric criterion can simply taken to be

$$\frac{f_n}{f_N} > 0 \qquad (n = L, \ldots, N),$$

with the optional exclusion of the other cases (8) and (10). If the criterion is satisfied, then we may apply the transformation $f \mapsto \operatorname{Log} f = \log(f/\operatorname{sign}(f_N))$, whose inverse is given by $\operatorname{Exp}_{f_N;N} = \operatorname{sign}(f_N) \exp$. As a variant, we may consider an asymptotic behaviour of the form

$$f_n = c\, \mathrm{e}^{R_n},$$

and use the atomic transformation $Q = \Delta \circ \mathrm{Log}$: $f_n \mapsto \log(f_{n+1}/f_n) = R_{n+1} - R_n$.

Composite transformations. Sometimes, it is useful to combine the above basic atomic combinations in order to detect more specific types of asymptotic behaviour. For instance, if

$$f_n = c_1\, \psi_{1,n} + \cdots + c_k\, \psi_{k,n} + R_n \qquad (\psi_1 \succ \cdots \succ \psi_k \succ R_n), \tag{11}$$

then we may successively apply the operators

$$\Phi_i = \Delta \circ \times_{\Phi_{i-1}(\psi_i)^{-1}}$$

for $i = 1, \ldots, k$, and where $\Phi_0 = \mathrm{Id}$. This is equivalent to the E-algorithm and related to the "all-in-one" transformation given by

$$f_n \longmapsto \begin{vmatrix} f_n & f_{n+1} & \cdots & f_{n+k} \\ \psi_{1,n} & \psi_{1,n+1} & \cdots & \psi_{1,n+k} \\ \vdots & \vdots & & \vdots \\ \psi_{k,n} & \psi_{k,n+1} & \cdots & \psi_{k,n+k} \end{vmatrix} \Bigg/ \begin{vmatrix} 1 & 1 & \cdots & 1 \\ \psi_{1,n} & \psi_{1,n+1} & \cdots & \psi_{1,n+k} \\ \vdots & \vdots & & \vdots \\ \psi_{k,n} & \psi_{k,n+1} & \cdots & \psi_{k,n+k} \end{vmatrix}.$$

Of course, $\Psi \circ \Phi$ is acceptable for f if and only if Φ and Ψ are acceptable for f resp. $\Phi(f)$. Another example is given by asymptotic behaviour of the type

$$f_n = c_1\, n^{c_2} + R_n \qquad (R_n \prec 1). \tag{12}$$

In that case, we may apply the transformation $\Delta \circ \times_n \circ Q$. More generally, the logarithmic/exponential transformations can be rather violent, so it is often useful to restrict their acceptability.

Transformation synthesis. If we are confronted to any other kind of asymptotic behaviour (6), which does not fall in the above cases, then we may investigate systematic ways to synthesize an appropriate atomic transformation Φ, such that the dominant asymptotic behaviour of $\Phi(f)$ depends on R_n. The main trouble comes from the fact that we need a transformation of the form $f \mapsto \xi(f, \ldots, \Delta^k f)$.

Indeed, consider the modified problem, when f is an infinitely differentiable function on $[0, N]$ rather than a sequence. For simplicity, we will also assume that $\varphi_{c_1,\ldots,c_r}(R_n) = \psi_{c_1,\ldots,c_r} + R_n$, where ψ is an exp-log function in $c_1, \ldots, c_r$ and n. Since the set of algebraically differential functions forms a field which is closed under composition, ψ satisfies an algebraic differential equation $P(\psi, \ldots, \psi^{(s)}) = 0$ in n and we simply take $f \mapsto P(f, \ldots, f^{(s)})$ for our atomic transformation. If g is sufficiently small, then the intermediate value theorem for transseries [vdH06c, Chapter 9] ensures the existence of a transseries solution to $P(f, \ldots, f^{(s)}) = g$, which can actually be given an analytic meaning [vdH06b].

Unfortunately, the field of functions which satisfy an algebraic difference equation is not closed under composition. In particular, let us consider the sequence n^c, where c is a formal parameter. We claim that this sequence does not satisfy an algebraic difference equation. Indeed, assume the contrary and let

$$P(n^c, \ldots, (n+s)^c) = 0$$

be a relation of minimal order and minimal degree. Considering n as a complex number, turning around the singularity at $n = -s$ results in a new relation

$$P(n^c, \ldots, (n+s-1)^c, \mathrm{e}^{2\pi \mathrm{i} c}(n+s)^c) = 0$$

Combining this relation with the previous one, we obtain a relation of smaller degree or order. In particular, this shows that for an asymptotic behaviour of the form (12), there exists no algebraic atomic transformation Φ which will completely eliminate $c_1\, n^{c_2}$ and make R_n apparent in the dominant behaviour of $\Phi(f)$.

3. A meta-algorithm for asymptotic extrapolation

Throughout this section we assume that we have fixed a finite set $\boldsymbol{\Phi}$ of atomic transformations. One possible choice for $\boldsymbol{\Phi}$ is $\{\Delta, Q, \times_{n^{-1}}, \times_n, \times_{n^2}\}$, but other selections will be discussed below. Each transformation $\Phi \in \boldsymbol{\Phi}$ comes with a criterion for acceptability. Furthermore, given an asymptotic extrapolation $\tilde{g}$ for $g = \Phi(f)$, we assume that we may compute the inverse operator $\Phi_\pi^{-1} = \Phi_{f_{N-r_\Phi+1},\ldots,f_N;N}^{-1}$ of Φ, for which $\tilde{f} = \Phi_\pi^{-1}(\tilde{g})$ coincides with f on the last r_Φ known coefficients $f_{N-r_\Phi+1}, \ldots, f_N$.

Given an input series f and an "extrapolation order" l, the asymptotic extrapolation algorithm simply attempts to apply all composite transformations $\Phi_k \circ \cdots \circ \Phi_1$ on f, where $\Phi_1, \ldots, \Phi_k \in \boldsymbol{\Phi}$ and $k \leqslant l$. The algorithm returns a set of possible extrapolations. Each asymptotic extrapolation is returned in the form of a *recipe* $\Psi = \Phi_{1,\pi_1}^{-1} \circ \cdots \circ \Phi_{k,\pi_k}^{-1}$, for suitable parameters $\pi_1, \ldots, \pi_k$. When applying this recipe to the zero sequence, we obtain a numeric extrapolation $\tilde{f} = \Phi(0)$ of f. In particular, the resulting extrapolations can optionally be sorted on the relative error $\varepsilon_{f,\tilde{f}}$, after which we may keep only the best or K best results. The inverse $\Phi_k \circ \cdots \circ \Phi_1$ of a recipe will be called a *reductor*.

Algorithm `extrapolate`(f, l)
Input: a finite sequence $f_0, \ldots, f_N$ and the extrapolation order l
Output: a set of extrapolations of the form $(\tilde{f}, \Psi)$ for f

```
Let $E := \{\mathrm{Id}\}$
For each $\Phi \in \boldsymbol{\Phi}$ such that $\Phi$ is acceptable for $f$ and $r_\Phi \geqslant l$ do
    $E' := \mathrm{extrapolate}(\Phi(f), l - r_\Phi)$
    For each $\Psi' \in E'$ do
        Compute parameters $\pi$ with $f_i = \tilde{f}_i = (\Phi_\pi^{-1} \circ \Psi')(0)_i$ for $i = N - r_\Phi + 1, \ldots, N$
        Set $E := E \cup \{\Phi_\pi^{-1} \circ \Psi'\}$
Return $E$
```

REMARK 2. In practice, $\boldsymbol{\Phi} = \{\Phi_1, \ldots, \Phi_t\}$ is often designed in such a way that no more than one transformation Φ_i is acceptable at a time. For instance, the criterion for acceptability by Φ_i might include the condition that Φ_j is not acceptable for any $j < i$. Indeed, it is important to keep the number of accepted transformations small at each stage, in order to prevent combinatorial explosion. For some additional considerations, we refer to section 6.3.

REMARK 3. In order to increase control over the accuracy, it is possible to perform all computations using interval arithmetic [Moo66, AH83, GPR03]. In that case, the coefficients $f_0, \ldots, f_N$ are intervals instead of floating point numbers, and the deterioration of the relative precision will be more explicit during the application of the successive transformations. As to the actual extrapolation, one should replace 0 by the smallest interval which contains $f_L, \ldots, f_N$ at the last step, and similarly for the determination of the other continuous parameters. Again, this has the advantage of giving a more precise idea on the accuracy of the continuous parameters.

Clearly, the choice of $\mathbf{\Phi}$ and the corresponding criteria is primordial for the success of our extrapolation method under various circumstances. Reduced sets $\mathbf{\Phi}$ of transformations and strict criteria are recommended in the case when the user already has some idea about possible shapes for the asymptotic expansion. Larger sets $\mathbf{\Phi}$ and looser criteria may allow the detection of more complex asymptotic behaviours, at the risk of finding incorrect extrapolations. Let us discuss a few other possible choices for $\mathbf{\Phi}$ in more detail.

Power series expansions. Assume that the sequence f_n admits an asymptotic expansion in the form of a power series

$$f_n \approx c_0 + \frac{c_1}{n} + \frac{c_2}{n^2} + \cdots \tag{13}$$

Applying the transformation $\times_{n^2} \circ \Delta$ on this expansion, we obtain

$$(\times_{n^2} \circ \Delta f)_n \approx -c_1 + \frac{c_1 - 2\,c_2}{n} + \frac{-c_1 + 3\,c_2 - 3\,c_3}{n^2} + \cdots,$$

which has a similar shape. Assuming that none of the coefficients c_i vanishes, the set $\mathbf{\Phi} = \{\times_{n^2} \circ \Delta\}$ or $\mathbf{\Phi} = \{\Delta, \times_{n^2}\}$ therefore suffices for the extrapolation of f. In the case when some of the coefficients vanish, we should rather take $\mathbf{\Phi} = \{\Delta, \times_{n^2}, \ldots, \times_{n^p}\}$.

Logarithmic coefficients. Assume now that f has an asymptotic expansion

$$f_n \approx \sum_{i=0}^{\infty} \sum_{j=0}^{\omega_i} c_{i,j} \frac{(\log n)^j}{n^i}.$$

If $L = \lfloor \lambda N \rfloor$ for a fixed constant $0 < \lambda < 1$, then the relative error $\varepsilon_{1,(\log n)^j}$ tends to zero for $N \to \infty$ and fixed j. In particular, if $\Delta \in \mathbf{\Phi}$, then the asymptotic extrapolation algorithm will attempt to apply Δ on any sequence f_n with $f_n \asymp (\log n)^j$. Since

$$n\,\Delta(\log n)^j = j\,(\log n)^{j-1} + \cdots,$$

it follows that the set $\mathbf{\Phi} = \{\Delta, \times_n, \ldots, \times_{n^p}\}$ usually suffices for the asymptotic extrapolation of f (assuming that N is sufficiently large).

Unknown exponents. In the previous section, we have seen that the operator $\Delta \circ \times_n \circ Q$ allows us to detect asymptotic behaviours of the form (12). More generally, assume that f_n admits a generalized power series expansion with unknown (real) exponents

$$f_n \approx \frac{c_0}{n^{\alpha_0}} + \frac{c_1}{n^{\alpha_1}} + \frac{c_2}{n^{\alpha_2}} + \cdots \qquad (c_0 < c_1 < c_2 < \cdots).$$

Then the set $\mathbf{\Phi} = \{\Delta, \Delta \circ \times_n \circ Q, \times_n, \times_{n^2}\}$ will suffice for the asymptotic extrapolation of f.

Simple tails. It often occurs that the asymptotic expansion of f is the product of a complicated "transmonomial" and a simple power series expansion, as in (1). In that case, we may slightly modify our extrapolation algorithm in order to allow for a larger set $\mathbf{\Phi} = \{\Delta, Q, \times_{n^{-1}}, \times_n, \times_{n^2}\}$ for the first few steps and a reduced set $\mathbf{\Phi} = \{\Delta, \times_{n^2}, \ldots, \times_{n^p}\}$ for later steps.

4. Asymptotic extrapolations in exp-log form

Asymptotic extrapolations of f, as computed in the previous section, are given either numerically, or using a recipe in terms of the inverses of the operators in $\mathbf{\Phi}$. For instance, in the case of a power series expansion (13), the computed recipe is typically of the form

$$\Delta_{a_1;N}^{-1} \circ \times_{n^{-2}} \circ \cdots \circ \Delta_{a_k;N+1-k}^{-1} \circ \times_{n^{-2}}. \tag{14}$$

Unfortunately, the corresponding values of $c_0, \ldots, c_{k-1}$ can not directly be read off from this recipe. In this section, we will discuss techniques for obtaining asymptotic extrapolations in a symbolic and more human readable form.

Remark 4. Of course, "human readability" is a subjective notion. In a sense, recipes of the form (14) are already symbolic expressions for the asymptotics.

Remark 5. Sometimes, asymptotic extrapolation is used for the accurate computation of some parameters, such as the radius of convergence, rather than a complete asymptotic expansion. In such cases, symbolic post-treatment is not necessary and may even deteriorate the accuracy of the extrapolation.

A convenient and quite general setting for the computation with asymptotic expansions is to restrict our attention to exp-log functions or transseries. An *exp-log function* is constructed from the constants and an infinitely large indeterminate n using the field operations, exponentiation and logarithm. Several algorithms exist for the computation with exp-log functions at infinity [Sha89, Sal91, RSSvdH96, Sha96, Gru96, vdH97]. A *transseries* is formed in a similar way, except that, under certain support conditions, we allow additions to be replaced by infinite summations. An example of a transseries is

$$1!\, n^{-1}\, \mathrm{e}^{n + \frac{n}{\log n} + \frac{n}{\log^2 n} + \cdots} + 2!\, n^{-2}\, \mathrm{e}^{n + \frac{n}{\log n} + \frac{n}{\log^2 n} + \cdots} + 3!\, n^{-3}\, \mathrm{e}^{n + \frac{n}{\log n} + \frac{n}{\log^2 n} + \cdots} + \cdots.$$

The field of formal transseries at infinity is stable under many operations, such as differentiation, integration, composition and functional inversion. However, transseries are formal objects and often divergent. Techniques for the computation with transseries can be found in [Éca92, RSSvdH96, vdH97, vdH06c, vdH08].

The aim of this paper is not to go into details about the computational aspects of transseries. A heuristic approach, which is particularly well adapted to the present context, is described in [vdH08]. From the computational point of view, we recall that a transseries f is approximated by an infinite sum

$$f = \sum_{k=0}^{\infty} f_k, \tag{15}$$

where each f_k is a *finite* linear combination of transmonomials. Each transmonomial is either an iterated logarithm or the exponential of another transseries. Furthermore, it is described in [vdH08] how to perform common operations on transseries, such as differentiation, integration, composition, as well as discrete summation Δ^{-1}. Finally, we notice that the technique of summation up to the least term [Poi93] can be used recursively for the approximate numeric evaluation of (15), even if f is divergent. Whenever $\tilde{g}$ extrapolates $g = \Delta f$, this allows us in particular to compute $\Delta_{f_N;N}^{-1}(\tilde{g}) = \Delta^{-1}(\tilde{g}) + a$, with $a = f_N - \Delta^{-1}(\tilde{g})(N)$.

Returning to our asymptotic extrapolation algorithm, it follows that the inverses of the atomic transformations from section 2 can systematically be applied to transseries instead of numeric sequences. In other words, given the recipe Ψ of an asymptotic extrapolation, we simply compute $\Psi(0)$ as a transseries. If we want an exp-log extrapolation, then we truncate $f \approx f_0 + \cdots + f_k$ for the order k which yields the best numeric extrapolation. Of course, the truncations are done recursively, for all transseries whose exponentials occur as transmonomials in the expansions of $f_0, \ldots, f_k$.

Several techniques can be used to enhance the quality of the returned exp-log extrapolations. Indeed, due to truncation and the process of summation up to the least term, the accuracies of the numeric parameters in the exp-log extrapolation may be deteriorated with respect to those in the original recipe. Let $\tilde{f}_{a_1, \ldots, a_k}$ be an exp-log extrapolation depending on parameters $a_1, \ldots, a_k$. One way to enhance the quality of our extrapolation is by iterative improvement. For instance, we may use Newton's method for searching the zeros of the function

$$\begin{pmatrix} a_1 \\ \vdots \\ a_k \end{pmatrix} \mapsto \begin{pmatrix} \tilde{f}_{a_1, \ldots, a_k}(N-k+1) - f_{N-k+1} \\ \\ \tilde{f}_{a_1, \ldots, a_k}(N) - f_N \end{pmatrix}.$$

Alternatively, we may perform a least square fit on the range $\{L, \ldots, N\}$. Another approach to *a posteriori* enhancement is to search closed form expressions for the parameters $a_1, \ldots, a_k$, such as integers, rational numbers or simple linear combinations of known constants π, e, etc. Such closed form expressions can be guessed by continuous fraction expansion or using the LLL-algorithm [LLL82]. In cases of success, it may be worthy to adjust the set of atomic transformations and rerun the extrapolation algorithm. For instance, if we find an extrapolation such as

$$f \approx 1.5904324561\, n^{1.5000000032} + \cdots,$$

then we may add $\times_{n^{-3/2}}$ as an atomic transformation. This spares the determination of one continuous parameter, which increases the accuracy of the successive transformed sequences.

5. Examples

In this section, we will present the results of running our algorithm on a series of explicit examples, as well as an example from [CGJ+05] suggested by one of the referees. We have tried several sets of atomic transformations $\mathbf{\Phi}$, using various criteria and thresholds. In what follows, we systematically take $L = \lfloor N/2 \rfloor$, and the algorithm works as follows:

1. If $\varepsilon_{\operatorname{Log} f, \log n} < 0.25$, $\varepsilon_{n \Delta \operatorname{Log} f, 1} < 0.05$ and $|(\times_{\log^2 n} \circ \Delta \circ \times_n \circ \Delta \circ \operatorname{Log})(f)| < 0.5$ (where norms are taken on $\{L, \ldots, N\}$), we apply the transformation $\Pi = \Delta \circ \times_n \circ \Delta \circ \operatorname{Log}$ in order to detect expansions of the form $f \approx c n^p + \cdots$. Moreover, if the last coefficient p of the sequence $(\times_n \circ \Delta \circ \operatorname{Log})(f)$ is close to an integer $|p - \lfloor p \rceil| < N^{-1/2}$, then we rather apply $\Delta \circ \times_{n^{-p}}$
2. If step 1 does not succeed and $\varepsilon_{f, n^p} < 0.25$ for some $p \in \{-2, -1, 0, 1\}$, then we apply the transformation $\Delta_p = \Delta \circ \times_{n^{-p}}$.
3. If steps 1 and 2 do not succeed and f does not change sign on $\{L, \ldots, N\}$, then we apply $Q = \Delta \circ \operatorname{Log}$.

In other words, we first check whether f has an expansion of the form $f \approx c n^p + \cdots$, with a high degree of certainty. If not, then we still try for $p = \{-2, -1, 0, 1\}$ with a bit less certainty. In cases of panic, we fall back on the transformation Q.

We want to emphasize that the above choices are preliminary. We are still investigating better choices for L and the thresholds, and more particular transformations for the detection of common expansion patterns.

5.1. Explicit sequences

For our first series of examples, we took $N = 1000$ and considered several explicit sequences. The results have been summarized in table 1 below. For all examples we used a precision of 512 bits, except for f_{IV}, in which case we used 4096 bits.

The first sequence $f_{\mathrm{I}} = \exp(n^{-1})$ is an ordinary power series in n^{-1}, which is well recognized (3 decimal digits per iteration, as expected). Notice the remarkable presence of a Δ_4-transformation. The second sequence $f_{\mathrm{II}} = n^{\pi} + n + 1$ demonstrates the ability to detect expansions in n with arbitrary exponents. The third sequence $f_{\mathrm{III}} = n!$ reduces to an ordinary power series expansion after the transformation $\Delta_1 \circ \Delta_0 \circ Q$ and is therefore well recognized. The fourth sequence $f_{\mathrm{IV}} = \zeta(n)$ shows that we are also able to detect expansions in e^{-n} with arbitrary exponents. The next two sequences f_{V} and f_{VI} essentially concern the detection of expansions of the form $\sum_k (a_k \log n + b_k)\, n^{-k}$. This works well for a few iterations, but the presence of $\log n$ tends to slow down the convergence and becomes problematic at higher orders. Finally, we considered an expansion in $(\log n)^{-1}$, which should be hopeless for our algorithm due to the slow increase of $\log n$.

Function	Order	Reductor	Relative error
$f_{\mathrm{I}} = \mathrm{e}^{\frac{1}{n}}$	10	$\Delta_2^7 \circ \Delta_4 \circ \Delta_2 \circ \Delta_0$	$7.284930271 \cdot 10^{-34}$
$f_{\mathrm{II}} = n^{\pi} + n + 1$	6	$\Pi \circ \Delta_2 \circ \Delta_2 \circ \Pi$	$1.079457292 \cdot 10^{-12}$
$f_{\mathrm{III}} = n!$	10	$\Delta_2^6 \circ \Delta_1 \circ \Delta_0 \circ Q$	$8.054709905 \cdot 10^{-21}$
$f_{\mathrm{IV}} = \zeta(n)$	9	$(\Delta_0 \circ Q)^4 \circ \Delta_0$	$3.272054867 \cdot 10^{-350}$
$f_{\mathrm{V}} = n + \log(n+1)$	4	$\Delta_0 \circ \Delta_1 \circ \Delta_2 \circ \Delta_{-1}$	$8.417894011 \cdot 10^{-9}$
$f_{\mathrm{VI}} = n\,\mathrm{e}^{\sqrt{n}}$	7	$Q \circ \Delta_0 \circ \Delta_2 \circ \Pi \circ \Delta_1 \circ Q$	$2.035560080 \cdot 10^{-11}$
$f_{\mathrm{VII}} = \frac{1}{\log n + 1}$	6	$\Delta_1^3 \circ Q \circ \Delta_1 \circ \Delta_0$	$2.481662714 \cdot 10^{-9}$

Table 1. Results when applying our algorithm to several explicit sequences for $N = 1000$ and order 10. In the table, we have shown the order of the best reductor and the corresponding relative error.

Having obtained the recipes for each of the sequences, we next applied the post-treatment of section 4 in order to find asymptotic expansions in the more readable exp-log form. The results are shown below:

$$\begin{aligned}
f_{\mathrm{I}} &\approx 1.000000000 + \frac{1.000000000}{n} + \frac{0.5000000000}{n^2} + \frac{0.1666666667}{n^3} + \\
&\quad \frac{0.04166666667}{n^4} + \frac{0.008333333333}{n^5} + \frac{0.001388888889}{n^6} + \cdots \\
f_{\mathrm{II}} &\approx 1.00000000\, n^{3.14159265} - 0.000001308\, n^{2.14159265} + 0.0451432651\, n^{1.14159265} + \\
&\quad 0.999749239\, n^{0.982199533} - 0.903748123\, n^{0.141592654} + \frac{2.12789974}{n^{0.0178004671}} + \cdots \\
f_{\mathrm{III}} &\approx 2.506628275\, \mathrm{e}^{1.0000000000 n \log(n) - 1.0000000000 n + 0.5000000000 \log(n)} + \\
&\quad \frac{0.2088856896\, \mathrm{e}^{1.0000000000 n \log(n) - 1.0000000000 n + 0.5000000000 \log(n)}}{n} + \cdots \\
f_{\mathrm{IV}} &\approx 1.000000000 + \frac{1.000000000}{\mathrm{e}^{0.6931471806 n}} + \frac{1.000000000}{\mathrm{e}^{1.098612289 n}} + \\
&\quad \frac{1.000000000}{\mathrm{e}^{1.386294361 n}} - \frac{0.5714285714}{\mathrm{e}^{1.504077397 n}} + \frac{0.5714285714}{\mathrm{e}^{1.504077397 n}} + \cdots \\
f_{\mathrm{V}} &\approx 0.0000009101\, n \log(n) + 0.9999911895\, n + 1.002816158 \log(n) - \\
&\quad 0.01863317212 + \frac{0.5014080791 \log(n)}{n} - \frac{0.7614285530}{n} + \cdots
\end{aligned}$$

$$\begin{aligned}
f_{\mathrm{VI}} \;\approx_5\; & 0.9949758928\,\mathrm{e}^{0.9929778942 n^{0.2504215280}+1.003963496\log(n)}+\\
& \frac{0.2405363359\,\mathrm{e}^{0.9929778942 n^{0.2504215280}+1.003963496\log(n)}}{n^{0.7495784720}}-\\
& \frac{0.4994597378\,\mathrm{e}^{0.9929778942 n^{0.2504215280}+1.003963496\log(n)}}{n}+\cdots\\
f_{\mathrm{VII}} \;\approx_5\; & \frac{79.57941000\,n\,\mathrm{e}^{0.02189743996\log(n)^2-1.677515325\log(n)}}{\log(n)^2}+\\
& \frac{2462.217956\,n\,\mathrm{e}^{0.02189743996\log(n)^2-1.677515325\log(n)}}{\log(n)^3}+\\
& \frac{62587.76904\,n\,\mathrm{e}^{0.02189743996\log(n)^2-1.677515325\log(n)}}{\log(n)^4}+\cdots
\end{aligned}$$

The obtained expansions are remarkably accurate in the cases of f_{I}, f_{III} and f_{IV}. We were particularly happy to recover Stirling's formula in a completely automatic numeric-symbolic manner. The expansions for f_{II}, f_{V} and f_{VI} are also close to the correct ones, when neglecting terms with small coefficients. The last expansion is more problematic; errors typically occur when a numerically small expression on $\{L, \ldots, N\}$ is incorrectly assumed to be infinitesimal. Such mistakes may lead to the computation of absurdly complicated transseries expansions, which occasionally fail to terminate. For instance, we only obtained exp-log expansions for extrapolations at order 5 in the last two examples.

5.2. Self avoiding walks on a triangular lattice

Let us now consider the sequence f_n, where f_n is the number of self-avoiding walks of size n on a triangular lattice [CGJ+05]. The terms $f_0, \ldots, f_N$ with $N = 40$ are as follows:

0, 6, 30, 138, 618, 2730, 11946, 51882, 224130, 964134, 4133166, 17668938, 75355206,
320734686, 1362791250, 5781765582, 24497330322, 103673967882, 438296739594,
1851231376374, 7812439620678, 32944292555934, 138825972053046, 584633909268402,
2460608873366142, 10350620543447034, 43518414461742966, 182885110185537558,
768238944740191374, 3225816257263972170, 13540031558144097474,
56812878384768195282, 238303459915216614558, 999260857527692075370,
4188901721505679738374, 17555021735786491637790, 73551075748132902085986,
308084020607224317094182, 1290171266649477440877690,
5401678666643658402327390, 22610911672575426510653226

It is theoretically known that

$$f_n \approx \mu^n\, n^{\frac{11}{32}}\,(a_0 + \cdots) \tag{16}$$

and the authors of [CGJ+05] actually expect an expansion of the type

$$f_n \approx \mu^n\, n^{11/32}\left(a_0 + \frac{a_1}{n} + \frac{a_2}{n^{3/2}} + \frac{a_3}{n^2} + \frac{a_4}{n^{5/2}} + \frac{a_5}{n^3} + \cdots\right). \tag{17}$$

In view of (16), we have applied our asymptotic extrapolation algorithm on f, by forcing the transformation $\mathrm{Log} \circ \times_{n^{-11/32}}$ at the start. This yields the expansion

$$\begin{aligned}
& 0.2852276870\,\mathrm{e}^{1.423300582n} + \frac{0.04609972489\,\mathrm{e}^{1.423300582n}}{n^{1.000000000}} - \frac{0.1672827822\,\mathrm{e}^{1.423300582n}}{n^{1.644753856}} +\\
& \frac{0.2440961517\,\mathrm{e}^{1.423300582n}}{n^{2.000000000}} - \frac{0.3967446855\,\mathrm{e}^{1.423300582n}}{n^{2.644753856}} + \frac{0.2828049628\,\mathrm{e}^{1.423300582n}}{n^{3.000000000}} + \cdots
\end{aligned}$$

which is quite close to the expected expansion. The corresponding reductor is given by

$$\Delta_1 \circ Q \circ \Delta_2 \circ \Delta_2 \circ \Delta_{-1} \circ \mathrm{Log} \circ \times_{n^{-11/32}}$$

As explained in [CGJ+05], the detection of the exponent $-3/2$ for a_2 is somewhat critical due to the possible appearance of an additional term $\tilde{a}_0 n^{-59/32} = \tilde{a}_0 n^{-1.84375}$. The authors argue that it is difficult to confirm or refute the presence of such a term, based on numerical evidence up to order $N = 40$ only. We agree with this analysis. Nevertheless, it is striking that the exponent returned by our algorithm is not far from the mean value -1.671875 of -1.5 and -1.84375. Also, the algorithm failed to detect a clean Π-transformation, which might indicate the presence of two terms with close exponents.

6. Possible improvements

There is still significant room for improving our choice of $\mathbf{\Phi}$ in section 5, and the corresponding criteria for acceptability. Failures to detect the expected form of asymptotic expansion can be due to the following main reasons:

1. Even for $N \to \infty$, the set $\mathbf{\Phi}$ of transforms does not allow for the detection of the correct asymptotic expansion.
2. At a certain step of the extrapolation process, the wrong atomic transformation is applied.
3. A recipe is found for the asymptotic extrapolation, but the corresponding exp-log extrapolation is incorrect or too hard to compute.

In this section, we will discuss the different types of failure in more detail and examine what can be done about them.

6.1. Inadequate transformations

Of course, the class of asymptotic expansions which can be guessed by our extrapolation algorithm is closely related to the choice of $\mathbf{\Phi}$. Asymptotic behaviour for which no adequate atomic transformation is available will therefore not be detected. For instance, none of the transformations discussed so far can be used for the detection of oscillatory sequences such as $f_n = \sin(\alpha n)$.

A more subtle sequence which cannot be detected using $\mathbf{\Phi} = \{\Delta, Q, \times_{n^{-1}}, \times_n, \times_{n^2}\}$ is $f_n = (\log n)^{-1}$. In order to detect $f_n = (\log n)^{-1}$, we might add the transformation $\times_{\log n}$. However, the corresponding criterion should be designed with care: as noticed in section 3, sequences of the form $(\log n)^{\alpha}$ are easily mistaken for constant sequences, when taking $L = \lfloor \lambda N/2 \rfloor$ for a fixed $\lambda \in (0, 1)$. A more robust criterion might require $\varepsilon_{f \log n, 1} < \delta$, $\varepsilon_{n \Delta f^{-1}, 1} < \delta$ as well as $\varepsilon_{n \Delta (f \log n), 1} \not< \delta$ for a suitable threshold δ.

Another more intricate difficulty occurs when the various transformations introduce an infinite number of parasite terms. For instance, consider the expansion

$$f_n \approx \alpha + \frac{\beta}{n} + \mathrm{e}^{-n} + \cdots. \tag{18}$$

When applying one Δ-transformation, we obtain

$$(\Delta f)_n \approx \frac{-b}{n^2} + \frac{b}{n^3} - \frac{b}{n^4} + \cdots + \left(1 - \frac{1}{\mathrm{e}}\right) \mathrm{e}^{-n} + \cdots. \tag{19}$$

The extrapolation of (19) therefore leads to an infinite number of Δ_2-transformations, in order to clear the infinity of terms which were introduced by the first Δ-transformation. In particular, the presence of the e^{-n} will not be detected. The phenomenon aggravates when parasite logarithmic terms are introduced. For instance, if we start the extrapolation of

$$f_n \approx \alpha^n n^\beta \left(c_0 + \frac{c_1}{n} + \frac{c_2}{n^2} + \cdots \right)$$

with the transformation $\times_{n^2} \circ \Delta_{-1} \circ \mathrm{Log}$ instead of Q, then we are lead to the extrapolation of an expansion of the form

$$(\times_{n^2} \circ \Delta_{-1} \circ \mathrm{Log}\, f)_n \approx \beta \log n + \frac{\lambda_1 \log n + \mu_1}{n} + \frac{\lambda_2 \log n + \mu_2}{n^2} + \cdots.$$

We have seen in the previous section (examples f_{V} and f_{VI}) that such expansions are much harder to detect than ordinary power series expansions. This is why we often prefer the Q-transformation over Log.

The real reason behind the above difficulty is that “nice” expansions in exp-log form, such as (18), do not coincide with “nice” expansions constructed using the inverses of operators in Φ. For instance, the expansion

$$f_n \approx -\frac{1}{n} - \frac{1}{2\,n^2} - \frac{1}{6\,n^3} + \frac{1}{30\,n^5} - \frac{1}{42\,n^7} + \frac{1}{30\,n^9} - \frac{5}{66\,n^{11}} + \frac{691}{2730\,n^{13}} - \frac{7}{6\,n^{15}} + \cdots + \mathrm{e}^{-n}$$

is “nice” in the second sense, since

$$\begin{aligned} \Delta f &\approx \frac{1}{n^2} + \left(1 - \frac{1}{\mathrm{e}}\right) \mathrm{e}^{-n} \\ \Delta_2 \Delta f &\approx -\left(\frac{1}{\mathrm{e}} - 1\right)^2 n^2\, \mathrm{e}^{-n} + \cdots \end{aligned}$$

In order to detect “nice” exp-log expansions, we have to adapt the set Δ so as to kill several continuous parameters at a time. In the case of linear combinations (11), this can be done using the transformations of the E-algorithm, modulo prior determination of the functions ψ_i. In general, the required transformations cease to be simple; see the discussion at the end of section 2. One idea might be to use the relation

$$\partial = \log(1 + \Delta),$$

in combination with summation up to the least term, in order to replace the finite difference operator by a genuine derivation.

6.2. Inadequate criteria

One particular strength of our asymptotic extrapolation algorithm is that it is strongly guided by the asymptotic behaviour of the original sequence and its successive transformations. If the asymptotic expansion of the sequence is given by a transseries, then this property is preserved during the extrapolation process. Theoretically, it is therefore possible to always select the right transformation at each stage.

In practice however, the asymptotic regime is not necessarily reached. Furthermore, numerical tests on the dominant asymptotic behaviour of a sequence have to be designed with care. If bad luck or bad design result in the selection of the wrong transformation at a certain stage, then the obtained extrapolation will be incorrect from the asymptotic view. Nevertheless, its numeric relative error on the range $\{L, ..., N\}$ may still be very small, which can make it hard to detect that something went wrong.

Let us first consider the situation in which the asymptotic regime is not necessarily reached. This is typically the case for expansions such as

$$f_n \approx 1 - \frac{1000}{n} + \cdots. \tag{20}$$

For $N \ll 1000$, our algorithm will incorrectly assume that $f_n \approx 1000\, n^{-1}$. The sequence f_{V} from section 5 leads to a similar situation:

$$\begin{aligned} (\Delta_2\, \Delta_1\, \Delta_2\, \Delta_{-1}\, f_{\mathrm{V}}) &\approx \log(n) - 6 - \frac{7\log(n)}{n} + \frac{95}{2\,n} + \frac{31\log(n)}{n^2} - \cdots & (21)\\ ((\Delta_2\, \Delta_1)^2\, \Delta_2\, \Delta_{-1}\, f_{\mathrm{V}}) &\approx -7\log(n) + 62 + \frac{145\log(n)}{n} - \frac{2677}{2\,n} - \frac{1549\log(n)}{n^2} + \cdots & (22)\end{aligned}$$

This explains our problems to detect the right expansion, even for large values of N such as $N \approx \mathrm{e}^6 \approx 403.429$ or $N \approx \mathrm{e}^9 \approx 8103.08$.

The best remedy is to add an atomic transformation which detects the two leading terms together. For instance, we may use the transformation $\Delta_0 \circ \Delta_1$ in (20) and $\Delta_1 \circ \Delta_0$ in (21) and (22). Of course, in order to gain something, we also need to adjust the acceptability criteria. For instance, in the cases of (21) and (22), we may simply require that $\varepsilon_{\Delta_0 f,1} < \delta$ for a suitable threshold δ, instead of the combined requirements $\varepsilon_{f,\log n} < \delta$ (or $\varepsilon_{f,1} < \delta$) and $\varepsilon_{\Delta_0 f,1} < \delta$. Of course, a drawback of our remedy is that there are many possibilities for the subdominant term(s) to be killed. Consequently, we will only be able to correct cancellations on or near the range $\{L, \ldots, N\}$ in a few special cases which occur frequently in practice. For the time being, it seems that polynomials in $\log n$ or n^{-1} deserve special treatment, but more experimentation is needed in order to complete this list.

Another major problem is the determination of a suitable value for L and good thresholds δ for tests $\varepsilon_{f,g} < \delta$ on the relative error. Let us first examine some frequent mistakes, due to inadequate thresholds. As we have seen before, our algorithm frequently assumes sequences of the form $(\log n)^k$ to be constant on the range $\{L, \ldots, N\}$. Fortunately, this is a rather harmless mistake, because we usually want to apply Δ both for 1 and $(\log n)^k$, $k \in \mathbb{N}$. A more annoying mistake is to consider n^{ε} as a constant for small $\varepsilon \neq 0$. This typically happens when δ is too large. Conversely, if δ is too small, then we may fail to recognize that $f_n = a + b\,(\log n)^{-1} \asymp 1$.

An interesting experiment for quantifying the risk of mistakes is to examine how well basic functions, such as n^{α}, $\log n$, $\mathrm{e}^{\alpha n}$, are approximated by other basic functions on the range $\{L, \ldots N\}$. More precisely, given sequences f_n and $g_{1,n}, \ldots, g_{k,n}$, we solve the system

$$\begin{pmatrix} g_{1,n_1} & \cdots & g_{k,n_1}\\ \vdots & & \vdots\\ g_{1,n_k} & \cdots & g_{k,n_k} \end{pmatrix} \begin{pmatrix} \lambda_1\\ \vdots\\ \lambda_k \end{pmatrix} = \begin{pmatrix} f_{n_1}\\ \vdots\\ f_{n_k} \end{pmatrix}$$

for $L = n_1 < \cdots < n_k = N$ and consider the relative error

$$\eta_{f;g_1,\ldots,g_k} = \varepsilon_{f,\lambda_1 g_1 + \cdots + \lambda_k g_k}$$

of the obtained approximation of f by $\lambda_1\, g_1 + \cdots + \lambda_k\, g_k$. Some results are shown in table 2, where we took the n_i equally spaced. The table shows a significant risk of confusions for $L = \lfloor N/2 \rfloor$ and the quasi-absence of confusions for $L = \lfloor \sqrt{N} \rfloor$. Unfortunately, when taking L too small, we often fail to be in the asymptotic regime and we will either need to increase the thresholds δ or improve the detection of several terms in the expansion at once. We may also start check the criteria several times for increasing values of L: transformations which are accepted for smaller values of L can be applied with a greater degree of confidence.

		$\eta_{f;g_1,\ldots,g_k}$ for $N=1024$ and $L=32,\ldots,512$				
f_n	$g_{1,n},\ldots,g_{k,n}$	32	64	128	256	512
$\log n$	$1,n^{-1}$	0.24759	0.15592	0.08616	0.037406	0.009053
$n^{-1/2}$	$1,n^{-1}$	0.33294	0.22222	0.12927	0.058875	0.014939
e^{-n}	$1,n^{-1}$	2.00000	2.00000	2.00000	2.000000	2.000000
n^{-1}	$\log n,1$	1.14526	0.82157	0.50005	0.233013	0.059642
$\log n$	$1,n^{-1}$	0.24759	0.15592	0.08616	0.037406	0.009053
$\log n$	$1,n^{-1},n^{-2}$	1.72035	0.35513	0.08644	0.016122	0.001272
$\log n$	$1,n^{-1},n^{-2},n^{-3}$	1.44191	0.59469	0.10606	0.009455	0.000250
$\log n$	$1,n^{-1},n^{-2},n^{-3},n^{-4}$	2.00000	2.00000	0.20042	0.006680	0.000059
$n^{-1/2}$	$1,n^{-1}$	0.33294	0.22222	0.12927	0.058875	0.014939
$n^{-1/2}$	$1,n^{-1},n^{-2}$	0.29686	0.14843	0.05718	0.014348	0.001399
$n^{-1/2}$	$1,n^{-1},n^{-2},n^{-3}$	2.00000	0.35220	0.05846	0.006534	0.000220
$n^{-1/2}$	$1,n^{-1},n^{-2},n^{-3},n^{-4}$	1.52100	0.57069	0.07160	0.003784	0.000044

Table 2. Approximation error of a given sequence f by a linear combination $\lambda_1 g_1+\cdots+\lambda_n g_n$ of other given sequences $g_1,\ldots,g_k$ on the range $\{L,\ldots,N\}$, as a function of L.

6.3. Multiple acceptable transformations

A final point about the design of a good set $\boldsymbol{\Phi}$ of atomic transformations is whether we allow several transformations to be acceptable for the same sequence (as in our meta-algorithm), or rather insist on immediately finding the right transformation at each step. For instance, in section 6.1, we argued that Q-transformations are usually preferable over Log-transformations. However, in section 5.2, starting with $Q\circ\times_{n^{-11/32}}$ instead of $\mathrm{Log}\circ\times_{n^{-11/32}}$ leads to an extrapolation of inferior quality. More generally, it is not rare that a better extrapolation is found using a slightly different transformation. This might be due to subtle cancellations in the asymptotic expansion which occur only for one of the transformations.

However, a systematic search for magic transformations is quite time-consuming, so we strongly recommend to restrict such searches to the first few iterations. Moreover, one should avoid using the increased flexibility for loosening the acceptability criteria: a better relative error for a numeric extrapolation is no guarantee for a better asymptotic quality. For instance, in the case of an expansion

$$f_n\approx\sum_{k\geqslant 0}\frac{a_{k,5}\,(\log n)^5+\cdots+a_{k,0}}{n^k},$$

with $N=1000$ and $L=500$, we can only hope to gain 3 decimal digits modulo an increase of the extrapolation order by 6. A simple interpolation

$$f_n\approx\sum_{k\geqslant 0}\frac{b_k}{n^k}$$

on the range $\{L,\ldots N\}$ will often yield better results at the same order. Nevertheless, we may expect $(\times_n\circ\Delta)^k f$ to be exceptionally small whenever $k=6,12,18,\ldots$, which should never be the case for transformations of the form $\times_n\circ\Delta\circ(\times_{n^2}\circ\Delta)^k f$. Be guided by the asymptotics at appropriate moments!

6.4. Incorrect exp-log extrapolations

Whenever the asymptotic extrapolation algorithms returns an incorrect answer, the computed exp-log extrapolations quickly tend to become absurd. Typically, we retrieve expansions of the type $\mathrm{e}^{\varepsilon n}$ or n^{ε} instead of $1 + \varepsilon\ n$ or $1 + \varepsilon\ \log\ n$, where $\varepsilon \approx 0$ is a small non-zero numeric parameter, which really should vanish from a theoretic point of view. In section 5.1, we have seen an example f_{VII} of this phenomenon. Actually, the computation of exp-log extrapolations frequently does not terminate at all. We probably should abort computations when the transseries extrapolations involve too many asymptotic scales.

The above phenomenon is interesting in itself, because it might enable us to improve the detection of incorrect expansions: when incorporating the computation of the exp-log extrapolations into the main algorithm, we may check *a posteriori* that the asymptotic assumptions made on the sequence are still satisfied by the resulting extrapolation. In this way, we might get rid of all extrapolations which are good from a numeric point of view on the range $\{L, \ldots, N\}$, but incorrect for large $n \gg N$. However, we have not investigated in detail yet whether our algorithms for the computation with transseries may themselves be a source of errors, especially in the case when we use summation up to the least term.

7. Possible generalizations

Our algorithm may be generalized along several lines. First of all, the restriction to real-valued sequences is not really essential. Most of the ideas generalize in a straightforward way to complex sequences. Only the transformation $f \rightarrow \log f$ requires a bit more care. In order to compute $g = \log f$, we first compute $g_N = \log f_N$ using the principal determination of the logarithm. We next compute the other terms using

$$g_i = g_{i+1} + \log \frac{f_i}{f_{i+1}}.$$

For more information about complex transseries, we refer to [vdH01]. Notice that the present algorithms will only work well for complex transseries expansions which do not essentially involve oscillation. The expansion

$$f = n^{2+\mathrm{i}} + 3\, n^{1+5\mathrm{i}} + n^{-7\mathrm{i}} + \frac{\log n}{n} + \cdots$$

will typically fall in our setting, but our algorithms will fail to recognize a sequence as simple as $f_n = \sin n$. One approach for removing this drawback is discussed in [PF07].

Another direction for generalization is to consider multivariate power series. For instance, assume that the coefficients f_{n_1,n_2} of a bivariate power series are known up to $n_1 + n_2 \leqslant N$. Then one may pick N_1 and N_2 with $N_1 + N_2 = N$ and apply the algorithm to the sequences $f_{N_1+k_1,n_2}$ in n_2 for different small values of $k_1 \in \mathbb{Z}$. One may next try to interpolate the resulting expansions as a function of k_1. Alternatively, one might investigate a generalization of the algorithm where discrete differentiation is applied in an alternate fashion with respect to n_1 and n_2. Of course, in the multivariate case, it is expected that there are several asymptotic regions, according to the choice of (N_1, N_2).

In fact, the dependency of the asymptotic expansion on the choice of (N_1, N_2) does not only occur in the multivariate case: for the algorithms presented here, it would be of interest to study the dependency of the obtained expansion as a function of N. For instance, consider the sequence

$$f_n = \frac{1}{1 - n^{-1} - 100\, n^{-\log n}},$$

which admits a transfinite asymptotic expansion

$$\begin{aligned} f_n \;=\; & 1+\frac{1}{n}+\frac{1}{n^2}+\cdots+\\ & \frac{100}{n^{\log n}}+\frac{100}{n^{\log n+1}}+\frac{100}{n^{\log n+2}}+\cdots+\\ & \frac{10000}{n^{2\log n}}+\frac{10000}{n^{2\log n+1}}+\frac{10000}{n^{2\log n+2}}+\cdots+\\ & \vdots \end{aligned}$$

Depending on the value of n some of the *a priori* invisible terms $100\, n^{-\log n}$, etc. may suddenly become dominant from a numerical point of view. This phenomenon is best illustrated by reordering the terms in the expansion as a function of their magnitude, for different values of n:

$$\begin{aligned} f_{10} &\approx 1+\frac{100}{n^{\log n}}+\frac{10000}{n^{2\log n}}+\frac{10^6}{n^{3\log n}}+\frac{1}{n}+\frac{10^8}{n^{4\log n}}+\frac{100}{n^{\log n+1}}+\frac{10^{10}}{n^{5\log n}}+\frac{10000}{n^{2\log n+1}}+\cdots\\ f_{25} &\approx 1+\frac{1}{n}+\frac{100}{n^{\log n}}+\frac{1}{n^2}+\frac{100}{n^{\log n+1}}+\frac{1}{n^3}+\frac{10000}{n^{2\log n}}+\frac{100}{n^{\log n+2}}+\frac{1}{n^4}+\frac{10000}{n^{2\log n+1}}+\cdots\\ f_{100} &\approx 1+\frac{1}{n}+\frac{1}{n^2}+\frac{1}{n^3}+\frac{100}{n^{\log n}}+\frac{1}{n^4}+\frac{100}{n^{\log n+1}}+\frac{1}{n^5}+\frac{100}{n^{\log n+2}}+\frac{1}{n^6}+\frac{100}{n^{\log n+3}}+\cdots\\ f_{250} &\approx 1+\frac{1}{n}+\frac{1}{n^2}+\frac{1}{n^3}+\frac{1}{n^4}+\frac{100}{n^{\log n}}+\frac{1}{n^5}+\frac{100}{n^{\log n+1}}+\frac{1}{n^6}+\frac{100}{n^{\log n+2}}+\frac{1}{n^7}+\cdots\\ f_{1000} &\approx 1+\frac{1}{n}+\frac{1}{n^2}+\frac{1}{n^3}+\frac{1}{n^4}+\frac{1}{n^5}+\frac{1}{n^6}+\frac{100}{n^{\log n}}+\frac{1}{n^7}+\frac{100}{n^{\log n+1}}+\frac{1}{n^8}+\frac{100}{n^{\log n+2}}+\cdots\\ f_{2500} &\approx 1+\frac{1}{n}+\frac{1}{n^2}+\frac{1}{n^3}+\frac{1}{n^4}+\frac{1}{n^5}+\frac{1}{n^6}+\frac{1}{n^7}+\frac{100}{n^{\log n}}+\frac{1}{n^8}+\frac{100}{n^{\log n+1}}+\frac{1}{n^9}+\cdots\\ f_{10000} &\approx 1+\frac{1}{n}+\frac{1}{n^2}+\frac{1}{n^3}+\frac{1}{n^4}+\frac{1}{n^5}+\frac{1}{n^6}+\frac{1}{n^7}+\frac{1}{n^8}+\frac{100}{n^{\log n}}+\frac{1}{n^9}+\frac{100}{n^{\log n+1}}+\cdots \end{aligned}$$

It may therefore be interesting to apply the asymptotic extrapolation algorithm for different values of N and try to reconstruct the complete transfinite expansion by combining the different results. Here we notice that the detection of a term like $100/n^{\log n}$ in the expansion for $N=10$ may help for the detection of the term $1/n$ in the second expansion for $N=25$, since we may subtract it from the expansion (and *vice versa*).

Acknowledgment. We would like to thank U. FRISCH and W. PAULS for various stimulating conversations and for their comments on a draft of this paper. We also thank the anonymous referees for their remarks and references.

Bibliography

[AH83] G. Alefeld and J. Herzberger. *Introduction to interval analysis*. Academic Press, New York, 1983.

[BL94] B. Beckermann and G. Labahn. A uniform approach for the fast computation of matrix-type Padé approximants. *SIAM J. Matrix Analysis and Applications*, pages 804–823, 1994.

[BZ91] C. Brezinski and R. Zaglia. *Extrapolation Methods. Theory and Practice*. North-Holland, Amsterdam, 1991.

[CDFT01] M. Canalis-Durand, F. Michel F., and M. Teisseyre. Algorithms for formal reduction of vector field singularities. *Journal of Dynamical and Control Systems*, 7(1):101–125, January 2001.

[CGJ+05] S. Caracciolo, A. J. Guttmann, I. Jensen, A. Pelissetto, A. N. Rogers, and A. D. Sokal. Correction-to-scaling exponents for two-dimensional self-avoiding walks. *J. Stat. Phys.*, 120:1037–1100, 2005.

[Der94] H. Derksen. An algorithm to compute generalized padé-hermite forms. Technical Report Rep. 9403, Catholic University Nijmegen, January 1994.

[Éca92] J. Écalle. *Introduction aux fonctions analysables et preuve constructive de la conjecture de Dulac*. Hermann, collection: Actualités mathématiques, 1992.

[FS96] P. Flajolet and R. Sedgewick. *An introduction to the analysis of algorithms*. Addison Wesley, Reading, Massachusetts, 1996.

[GPR03] M. Grimmer, K. Petras, and N. Revol. Multiple precision interval packages: Comparing different approaches. Technical Report RR 2003-32, LIP, École Normale Supérieure de Lyon, 2003.

[Gru96] D. Gruntz. *On computing limits in a symbolic manipulation system*. PhD thesis, E.T.H. Zürich, Switzerland, 1996.

[Har10] G.H. Hardy. *Orders of infinity*. Cambridge Univ. Press, 1910.

[Har11] G.H. Hardy. Properties of logarithmico-exponential functions. *Proceedings of the London Mathematical Society*, 10(2):54–90, 1911.

[JG99] I. Jensen and A. J. Guttmann. Self-avoiding polygons on the square lattice. *J. Phys.*, 32:4867–4876, 1999.

[LLL82] A.K. Lenstra, H.W. Lenstra, and L. Lovász. Factoring polynomials with rational coefficients. *Math. Ann.*, 261:515–534, 1982.

[Moo66] R.E. Moore. *Interval analysis*. Prentice Hall, Englewood Cliffs, N.J., 1966.

[Pau07] W. Pauls. *Complex singularities of ideal incompressibles flows*. PhD thesis, University Nice-Sophia Antipolis, 2007.

[PF07] W. Pauls and U. Frisch. A Borel transform method for locating singularities of Taylor and Fourier series. *J. Stat. Phys*, 127:1095–1119, 2007.

[PMFB06] W. Pauls, T. Matsumoto, U. Frisch, and J. Bec. Nature of complex singularities for the 2D Euler equation. Technical report, 2006.

[Poi93] H. Poincaré. *Les méthodes nouvelles de la mécanique céleste*, volume Tôme II. Gauthier-Villars, Paris, 1893.

[Pól37] G. Pólya. Kombinatorische Anzahlbestimmungen für Gruppen, Graphen und chemische Verbindungen. *Acta Mathematica*, 68:145–254, 1937.

[RSSvdH96] D. Richardson, B. Salvy, J. Shackell, and J. van der Hoeven. Expansions of exp-log functions. In Y.N. Lakhsman, editor, *Proc. ISSAC '96*, pages 309–313, Zürich, Switzerland, July 1996.

[Sal91] B. Salvy. *Asymptotique automatique et fonctions génératrices*. PhD thesis, École Polytechnique, France, 1991.

[Sha89] J. Shackell. A differential-equations approach to functional equivalence. In *Proc. ISSAC '89*, pages 7–10, Portland, Oregon, A.C.M., New York, 1989. ACM Press.

[Sha90] J. Shackell. Growth estimates for exp-log functions. *Journal of Symbolic Computation*, 10:611–632, 1990.

[Sha96] J. Shackell. Limits of Liouvillian functions. *Proc. of the London Math. Soc.*, 72:124–156, 1996. Appeared in 1991 as a technical report at the Univ. of Kent, Canterbury.

[SZ94] B. Salvy and P. Zimmermann. Gfun: a Maple package for the manipulation of generating and holonomic functions in one variable. *ACM Trans. on Math. Software*, 20(2):163–177, 1994.

[vdH97] J. van der Hoeven. *Automatic asymptotics*. PhD thesis, École polytechnique, Palaiseau, France, 1997.

[vdH01] J. van der Hoeven. Complex transseries solutions to algebraic differential equations. Technical Report 2001-34, Univ. d'Orsay, 2001.

[vdH02a] J. van der Hoeven. Relax, but don't be too lazy. *JSC*, 34:479–542, 2002.

[vdH02b] J. van der Hoeven et al. Mathemagix, 2002. `http://www.mathemagix.org`.

[vdH06a] J. van der Hoeven. Algorithms for asymptotic interpolation. Technical Report 2006-12, Univ. Paris-Sud, 2006.

[vdH06b] J. van der Hoeven. Transserial Hardy fields. Technical Report 2006-37, Univ. Paris-Sud, 2006. Accepted for publication.

[vdH06c] J. van der Hoeven. *Transseries and real differential algebra*, volume 1888 of *Lecture Notes in Mathematics*. Springer-Verlag, 2006.

[vdH08] J. van der Hoeven. Meta-expansion of transseries. Technical Report 2008-03, Université Paris-Sud, Orsay, France, 2008. Short version presented at ROGICS 2008, Mahdia, Tunesia.

[Wen01] E. J. Weniger. Nonlinear sequence transformations: Computational tools for the acceleration of convergence and the summation of divergent series. Technical Report math.CA/0107080, Arxiv, 2001.

[Wil04] H.S. Wilf. *Generatingfunctionology*. Academic Press, 2nd edition, 2004. `http://www.math.upenn.edu/~wilf/DownldGF.html`.