Fuchsian differential equations from modular arithmetic

Iwan Jensen

Abstract. Counting combinatorial objects and determining the associated generating functions can be computationally very difficult and expensive when using exact numbers. Doing similar calculations modulo a prime can be orders of magnitude faster. We use two simple polygon models to illustrate this: we study the generating functions of (singly) punctured staircase polygons and imperfect staircase polygons, counted by their extent along the main diagonal. For the former model this is equivalent to counting by the half-perimeter of the outer staircase polygon. We derive long series for these generating functions modulo a single prime, and then proceed to find Fuchsian ODEs satisfied by the generating functions, modulo this prime. Knowledge of a Fuchsian ODE modulo a prime will generally suffice to determine exactly its singular points and the associated characteristic exponents. We also present a procedure for the efficient reconstruction of the exact ODE, using results from multiple mod-prime calculations. Finally, we demonstrate how modular calculations can be used to factor Fuchsian differential operators.

1. Introduction

Counting the number of combinatorial objects $p_n$ of size $n$ (say, the number of self-avoiding polygons of perimeter $2n$, on the square lattice), and determining the corresponding generating function $F(x) = \sum p_n x^n$, is a fundamental pursuit of algebraic and enumerative combinatorics. A time-honored approach is to generate the series numerically (i.e., calculate $p_n$ using a computer) and then try to guess the generating function using a symbolic package such as Gfun [SZ]. For typical combinatorial problems the number of objects grows exponentially with the size $n$, so computationally it can be extremely expensive to calculate $p_n$. Furthermore, various off-the-shelf search programs can quickly become stuck because of the often very large integer coefficients involved ($2^n$ quickly becomes huge), and programs relying on exact formal calculations may easily run into memory barriers as well as time constraints.

In this article we review recent work, in which we have undertaken such calculations modulo a prime. This can be many orders of magnitude faster, and since the
size of the coefficients is fixed and quite small, the problem of exhausting available memory is also somewhat alleviated. Specifically, our approach is to start from a series \( F(x) \) known modulo a single prime, and then search for a linear differential equation with polynomial coefficients which has \( F(x) \) as a solution (see Section 2). We illustrate the use of our methods by studying in Section 3 the generating functions \( P(x) \) and \( I(x) \) of punctured and imperfect staircase polygons on the square lattice. We calculate long series for these generating functions modulo a single prime, and then proceed to find ODEs modulo this prime. (That is, we calculate the coefficients of the polynomials in ODEs which \( P(x) \) and \( I(x) \) satisfy, modulo the prime.) We demonstrate in Section 3.2 how knowledge of an ODE modulo a single prime suffices to determine exactly its singular points and the associated characteristic exponents. That is, we need not know the ODE in exact arithmetic in order to find them exactly.

For \( P(x) \), the mod-prime ODE is sufficiently simple that we decided additionally to generate the exact series and from this calculate an exact ODE, which serves as a powerful check on the validity of the preceding modular calculations. (See Section 3.3.) For \( I(x) \), the mod-prime ODE is more complicated, so we developed a procedure outlined in Section 4 for the efficient reconstruction of an exact ODE using results from multiple mod-prime calculations. For this particular problem the multiple mod-prime approach is a factor of about 1000 times faster than first calculating the series exactly to the required order, and then finding an exact ODE using exact arithmetic.

In Section 5, we demonstrate how modular calculations can also be used to factor the differential operators in the calculated ODEs, in particular the order-11 one in the minimal ODE satisfied by \( P(x) \). Again, symbolic packages exist which do a decent job on ODEs of moderate size. The MAPLE package \texttt{DEtools} is very advanced and contains many useful routines, but for this order-11 ODE it simply fails, by running out of memory and time (the ‘black-box’ factorization routine \texttt{DFactor} quickly consumes many gigabytes of memory).

The use of modular calculations as outlined in this article promises to make it possible to find exact solutions, through numerical means, to many combinatorial problems which have hitherto been considered too computationally difficult. The prime example of this is probably our recent work on the 5-particle contribution \( \tilde{\chi}^{(5)} \) to the square lattice Ising model [BGHJ]. The \( n \)-particle contributions \( \tilde{\chi}^{(n)} \), which are combinatorial generating functions, can be expressed in terms of quite involved \((n - 1)\)-fold integrals over algebraic functions. The research outlined here stems from the studies of \( \tilde{\chi}^{(3)} \) and \( \tilde{\chi}^{(4)} \) by Zenine et al. [ZBHM1, ZBHM2, ZBHM3], where they found that these contributions are the solutions of quite high order (7 and 10, respectively) Fuchsian ODEs. Guttmann and Jensen [GJ1, GJ2] then proceeded to show that the generating functions for the combinatorial problems of punctured and three-choice polygons are solutions of 8th order Fuchsian ODEs. These studies needed moderately long series (a few hundred terms or so) in order to find the ODE, which was done using the exact series. This research has since progressed via the study of \( \tilde{\chi}^{(5)} \) [BGHJ], which we found to be a solution of a high order ODE (the minimal order being 33), by using at least some 7400 terms of its series. The coefficients of the series and of the ODE were calculated modulo a single prime, and even attempting to generate the exact series (let alone finding an exact ODE) is completely beyond current computational resources.
The final piece of research reported on here, very briefly in Section 6, is our recent study \([BBGH]\) of the factorization properties of the differential operator in the minimal-order ODE satisfied by \(\tilde{\chi}(5)\). We can factor such ODE operators, using modular calculations informed by knowledge of their characteristic exponents.

2. Fuchsian differential equations

The starting point of our approach is to calculate using an appropriate computer program a long series expansion of some function \(F(x)\), which could be, for example, the generating function of a combinatorial problem. With the series coefficients known up to some order \(N\), we look for a linear differential equation of order \(M\) such that \(F(x)\) is an (approximate) solution, i.e.,

\[
\sum_{k=0}^{M} P_k(x) \frac{d^k}{dx^k} F(x) = 0 + O(x^{N+1}),
\]

where the \(P_k(x)\) are polynomials. In what follows we shall always consider linear ODEs and generally leave out the word ‘linear’. In order to make things as simple as possible, we limit the search to Fuchsian ODEs \([In]\). Such ODEs have only regular singular points. There are several reasons for searching for a Fuchsian ODE, rather than a more general differential equation. Computationally the Fuchsian assumption simplifies the search for a solution. From the general theory of Fuchsian equations \([In]\) it follows that the degree of \(P_k(x)\) is at most \(D - M + k\), where \(D\) is the degree of \(P_M(x)\). Thus any differential equation of Fuchsian type is constrained by two parameters, namely the order \(M\) and the degree \(D\) of the head polynomial \(P_M(x)\). One may also argue, less precisely, that for most ‘sensible’ combinatorial models one would expect Fuchsian equations, as irregular singular points are characterized by explosive, super-exponential behavior. Such behavior is not normally characteristic of combinatorial problems. The point at infinity may be an exception to this somewhat imprecise observation. Recent work by Bostan et al. \([BBHM]\) has thrown more light on why Fuchsian ODEs are so common in problems arising in statistical mechanics and enumerative combinatorics. One of their main observations is that many of the functions of interest can be written as \(n\)-fold integrals of algebraic integrands (they are, as pure mathematicians say, ‘derived from geometry’). As shown in \([BBHM]\), these functions are therefore necessarily solutions of ODEs which have rational coefficients, and moreover, are Fuchsian.

The Fuchsian condition on the ODE satisfied by \(F(x)\) requires that all singular points be regular, and specifically that \(x = 0\) and \(x = \infty\) be regular. A form for the ODE that automatically satisfies this condition is \(L_{MD}(F(x)) = 0\), where

\[
L_{MD} = \sum_{i=0}^{M} \left( \sum_{j=0}^{D} a_{ij} \cdot x^j \right) \cdot \left( x \frac{d}{dx} \right)^i, \quad a_{M0} \neq 0, \quad a_{MD} \neq 0.
\]

The condition \(a_{M0} \neq 0\) (resp. \(a_{MD} \neq 0\)) ensures that \(x = 0\) (resp. \(x = \infty\)) is regular. Note that it is the use of the operator \(x \frac{d}{dx}\) (sometimes called Euler’s operator), rather than just \(d/dx\), which leads to the preceding simple conditions guaranteeing regularity of \(x = 0\) and \(x = \infty\), and to the equality of the degrees of the polynomials in front of the derivatives. A simple rearrangement of terms casts
$L_{MD}(F(x)) = 0$ into the form $\hat{L}_{MD}(F(x)) = 0$, where

$$
(2.3) \quad \hat{L}_{MD} = \sum_{i=0}^{M} \left( \sum_{j=0}^{D} \hat{a}_{ij} \cdot x^{j+i} \right) \cdot \left( \frac{d}{dx} \right)^{i}
$$

and the coefficients $\hat{a}_{ij}$ are linear combinations of the $a_{ij}$.

The only major difference between $L_{MD}$ and $\hat{L}_{MD}$ is the change from using Euler’s operator $x \frac{d}{dx}$ to the standard operator $\frac{d}{dx}$. Either form can be used, but the form (2.2) has certain computational advantages. However, the form (2.3) was used in the original (and many subsequent) articles by Zenine et al. [ZBHM1, ZBHM2, ZBHM3]. It is also the form used by Guttmann and Jensen in their study of punctured and three-choice polygons [GJ1, GJ2]. The Fuchsian character of (2.3) is reflected in the decreasing degrees of the polynomials in front of successive derivatives. Both $L_{MD}$ and $\hat{L}_{MD}$ contain $(M+1)(D+1)$ unknown coefficients.

It should be noted that there is no unique ODE for a given series. The series can be annihilated by many differential operators, i.e., be a solution of many ODEs. But among these ODEs there is one of minimal order $m$ and degree $D_m$, and this ODE is unique. In terms of differential operators, the minimal-order differential operator appears as a right factor of any non-minimal-order differential operator. The minimal-order ODE may have a large number of ‘apparent’ singular points and can thus only be determined from a large number of series coefficients (generally speaking, $(m+1)(D_m+1)$ terms are needed). Other (non-minimal-order) ODEs, because they involve polynomials of smaller degrees, may require fewer series coefficients in order to be obtained. For any $M > m$, an ODE annihilating $F(x)$ (i.e., $L_{MD}(F(x)) = 0$), can be found for $D$ sufficiently large, and if $M$ is small enough we can choose $M$ and $D$ such that $(M+1)(D+1) < (m+1)(D_m+1)$. Among the non-minimal ODEs there will generally be one requiring the minimum number of terms. In a computational sense, one may view this as the ‘optimal’ ODE.

In previous articles such as [ZBHM1, ZBHM2, ZBHM3, GJ1, GJ2], the search for an ODE was done using exact series coefficients. The technique is to vary $M$ and $D$ until an $\hat{L}_{MD}$ is found with $\hat{L}_{MD}(F(x)) = 0 + O(x^{N+1})$, where from now on we shall drop the $O(x^{N+1})$. This condition leads to a set of linear equations for the unknown coefficients $\hat{a}_{ij}$. Naturally, one can always find such an ODE if the ODE is allowed to have at least $N$ coefficients. The real problem is to find an ODE of order $M$ and degree $D$ with $F(x)$ (known only via the first $N$ terms of its series) as a solution, such that the number of coefficients $K = (M+1)(D+1)$ of the ODE is less than $N$ (and generally significantly less). The $N-K$ series terms not required in order to find the ODE are thus a strong check on (though obviously not a proof of) the correctness of the ODE. The only major computational trick used in these previous articles was that rather than solving the set of linear equations for the $\hat{a}_{ij}$ using exact arithmetic, it was done using floating-point arithmetic with very high precision (say, up to 1000 digits). Once a solution was found, the floating-point numbers were turned into rational numbers. Obviously the set of linear equations always yields a solution, that is, one can always find an ODE annihilating the first $K = (M+1)(D+1)$ terms of the series. To check if the ODE is actually correct (i.e., annihilates the full series) one can check numerically, at the floating-point level, whether $\hat{L}_{MD}(F(x)) = 0$. However, a strong indication that the ODE is correct is for its coefficients to be fairly simple rational numbers. One accordingly factors $P_M(x)$ and checks whether the numbers of digits in the numerators of these
coefficients are ‘small,’ say, \( \ll 1000 \). One can perform a final check, using the ODE one has just obtained in exact arithmetic, to confirm that \( \hat{L}_{MD}(F(x)) = 0 \).

Finding the ODE, if its size is large, can be very time consuming both in generating the series for \( F(x) \) and in searching through values of \( M \) and \( D \), looking for the ODE. In recent work [BGHJ], we have adopted a different and much more efficient strategy. Rather than performing the search using the exact series, we search only for a solution modulo a specific prime (in practice, we use the prime \( p_r = 32749 = 2^{15} - 19 \)). The advantages of this are obvious. Firstly we only need generate a long series modulo a single prime (at least initially), and secondly, solving the system of linear equations determined by (2.2) largely amounts to finding whether or not the system has a zero determinant, which is easily done using Gaussian elimination. If a zero determinant is found, one can proceed to solve the system, which yields the ODE modulo the prime \( p_r \). In [BGHJ] we were quite remarkably able to work with a series of 10000 terms for which we found several mod-prime ODEs requiring the determination of more than 7400 unknown coefficients. (As in the exact procedure, the remaining 2600 or so terms served as a powerful check on the correctness of the obtained ODE.) We emphasize that for this problem it would be quite impossible to calculate the exact series coefficients, since they become huge, growing as \( 2^n \). Due to the size of the coefficients it would also be impossible to solve the set of linear equations for the coefficients of the ODE in either floating-point or exact arithmetic. In theory one must worry about possible false positive results, but we have never encountered this situation in practice (and in most cases, one can confirm any results by using a different prime). Below we give some further details of the procedure developed in [BGHJ].

To determine the unknown coefficients \( a_{kj} \) of the polynomials in (2.2), we arrange the set of linear equations \( (L_{MD}(F(x)) = 0) \) in a specific order (see [BGHJ] for details). There exists a nontrivial solution if the determinant of the matrix of the system of \((M + 1)(D + 1)\) linear equations vanishes. We test this by standard Gaussian elimination, creating an upper triangular matrix \( U \) in the process. If we find that a diagonal element \( U(K, K) = 0 \) for some \( K \), then a nontrivial solution exists. If \( K < K_{MD} := (M + 1)(D + 1) \), we set to zero all \( a_{kj} \) in the ordered list beyond \( K \). Of the remaining \( a_{kj} \) we set \( a_{M0} = 1 \), thus guaranteeing that \( x = 0 \) is a regular singular point, and determine the remaining coefficients by back-substitution. The \( K \) for which \( U(K, K) = 0 \) is the minimum number of series coefficients needed to find the ODE within the constraint of a given \( M \) and \( D \). Obviously, \( K \leq K_{MD} := (M + 1)(D + 1) \). Henceforth, \( D \) will always refer to the minimum degree \( D \) for which a solution was found for a given \( M \). Then, for example, we can define a unique non-negative deviation \( \Delta \) by \( K = K_{MD} - \Delta = (M + 1)(D + 1) - \Delta \). In [BGHJ] we made a very striking empirical observation, namely that the numbers \( K \) depend on \( M \) and \( D \) via the simple linear relation

\[
K = A \cdot M + B \cdot D + C = (M + 1)(D + 1) - \Delta,
\]

where \( A, B, \) and \( C \) are constants depending on the particular series. We have no proof of this formula, but it has been found to work for all the problems we have studied. Furthermore, we have found that the constant \( B \) is reliably related to the minimum order possible for the ODE. In most cases the \( B \) found from (2.4) equals the order \( m \) of the minimal-order ODE, although there are exceptions to this rule; e.g., when a constant function is a solution of the ODE, so that \( f_0(x) = 0 \). If this is the case, then \( B \) equals \( m - 1 \), and one can obviously extend this to cases
where several of the lowest order polynomials are 0. The order $m$ of the minimal-order ODE can thus be determined from non-minimal-order ODEs by using the remarkable formula (2.4). The constant $A$ is the minimal possible degree of the ODE, i.e., the number of singular points, counted with multiplicity but excluding any ‘apparent’ singular points, and also excluding $x = 0$.

3. Punctured and imperfect staircase polygons

In recent articles, Guttmann and Jensen studied the problems of punctured staircase polygons [GJ2] and three-choice polygons [GJ1], and found in each case that the perimeter generating function can be expressed as the solution of a 8th order Fuchsian ODE.

A staircase polygon can be viewed as the intersection of two directed walks starting at the origin, moving only to the right or up, and terminating once the walks join at a vertex [GP]. The perimeter length of a staircase polygon is even. Let us denote by $c_n$ the number of staircase polygons of perimeter $2n$. It is well known that $c_{n+1} = \frac{1}{n+1} \binom{2n}{n}$, i.e., that $c_{n+1} = C_n$, the $n$’th Catalan number. The associated half-perimeter generating function is

$$P(x) = \sum_{n=2}^{\infty} c_n x^n = \frac{1 - 2x - \sqrt{1 - 4x}}{2} \sim \text{const} \times (1 - \mu x)^{2-\alpha},$$

exhibiting both the connective constant $\mu = 4$, which determines the exponential growth of $c_n$ as $n \to \infty$, and the critical exponent $\alpha = 3/2$. Punctured staircase polygons [GJWE] are staircase polygons with internal holes which are also staircase polygons (i.e., the polygons are mutually- as well as self-avoiding). In [GJWE] it was proved that the connective constant $\mu$ of $k$-punctured staircase polygons (i.e., ones with $k$ holes) is the same as for unpunctured staircase polygons. Numerical evidence clearly indicates that the critical exponent $\alpha$ increases by $3/2$ per puncture. The closely related model of punctured discs was considered in [JvRW]. If punctured discs are counted by area, it was proved that the critical exponent increases by 1 per puncture. Here we study only the case of a single hole (see Figure 1), and we refer to these objects as punctured staircase polygons. The perimeter length of any staircase polygon is even, and thus the total perimeter of any punctured one

**Figure 1.** Examples of the types of polygons studied in this article.
(i.e., its outer perimeter plus that of the hole) is also even. We denote by \( p_n \) the number of punctured staircase polygons of perimeter \( 2n \). The results of [GJWE] imply that the half-perimeter generating function \( \sum_n p_n x^n \) has a simple pole at \( x = x_c = 1/\mu = 1/4 \), though the analysis in [GJWE] indicates that the critical behavior is more complicated than a simple algebraic singularity, and that logarithmic corrections to the dominant singular behavior are to be expected.

This was confirmed by a detailed analysis of the local solutions of the corresponding ODE, given in [GJ2]. Near the dominant singular point \( x = x_c = 1/4 \) the following singular behavior was found:

\[
(3.1) \quad A(x)(1 - 4x)^{-1} + B(x)(1 - 4x)^{-1/2} + C(x)(1 - 4x)^{-1/2} \log(1 - 4x),
\]

where \( A(x) \), \( B(x) \), and \( C(x) \) are analytic in a neighbourhood of \( x_c \). The ODE has other singular points. Near the singularity \( x = x_\pm = -1/4 \) on the negative \( x \)-axis, the behavior \( D(x)(1 + 4x)^{13/2} \) was found, where \( D(x) \) is analytic near \( x_- \). Similar behavior is expected near the pair of singularities \( x = \pm i/2 \), and finally at the roots of \( 1 + x + 7x^2 \) the behavior \( E(x)(1 + x + 7x^2)^2 \log(1 + x + 7x^2) \) is expected.

Three-choice self-avoiding walks on the square lattice were introduced by Manna [Ma], and can be defined as follows: Starting from the origin one can step in any direction; after a step upward or downward one can head in any direction (except backward); after a step to the left one can step forward or head downward, and similarly after a step to the right one can continue forward or turn upward. Alternatively put, one cannot make a right-hand turn after a horizontal step. As usual, one can define a polygon version of the walk model by requiring that the walk return to the origin. So a three-choice polygon [GPO] is simply a three-choice self-avoiding walk which returns to the origin, but has no other self-intersections. There are two distinct classes of three-choice polygons. The three-choice rule leads either to staircase polygons or to imperfect staircase polygons [CGD] (see Figure 1). Here we shall focus only on the case of imperfect staircase polygons, which as indicated in Figure 1 have exactly one notch or indentation. The ODE for the perimeter generating function of three-choice polygons has the same set of singularities as that for punctured staircase polygons, and the behavior at these singular points is almost the same, the only difference being that the first term in (3.1) is missing.

In the work reviewed below, the punctured and imperfect models are studied using an alternative counting variable, namely the ‘length’ (extent along the main diagonal) of the polygons. This ‘length’ is equal to the sum of the \( x \)- and \( y \)-coordinates of the point of the polygon furthest from the origin. For punctured staircase polygons this is equivalent to counting according to the half-perimeter of the outer staircase polygon (rather than the total perimeter of the outer and inner polygons combined). We denote the resulting generating functions for punctured and imperfect staircase polygons by \( \mathcal{P}(x) \) and \( \mathcal{I}(x) \), respectively.

### 3.1. Computer enumeration.

The algorithms that we used to count the number of punctured and imperfect staircase polygons by ‘length’ are modified versions of the algorithm of Conway, Guttmann, and Delest [CGD] for the enumeration of imperfect staircase polygons. The two problems are similar, and consequently there are only minor differences between the algorithms. A detailed description of the algorithm we used to count imperfect staircase polygons can be found in [GJ1], and the minor changes required to count punctured ones are outlined in [GJ2]. The algorithms are based on transfer matrix techniques. This
Table 1. Number of terms of the series that are needed, to find an ODE of the form (2.2) or (2.3), modulo the prime $p_r = 2^{15} - 19$. $M$ is the chosen order of the ODE, $D$ is the degree of each of its polynomial coefficients, $K_{MD} = (M+1)(D+1)$, $K$ is the number of terms predicted by (2.4) to be needed, and $\Delta = K_{MD} - K$.

<table>
<thead>
<tr>
<th>$M$</th>
<th>$D$</th>
<th>$K_{MD}$</th>
<th>$K$</th>
<th>$\Delta$</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>53</td>
<td>648</td>
<td>646</td>
<td>2</td>
</tr>
<tr>
<td>12</td>
<td>31</td>
<td>416</td>
<td>413</td>
<td>3</td>
</tr>
<tr>
<td>13</td>
<td>23</td>
<td>336</td>
<td>334</td>
<td>2</td>
</tr>
<tr>
<td>14</td>
<td>20</td>
<td>315</td>
<td>310</td>
<td>5</td>
</tr>
<tr>
<td>15</td>
<td>17</td>
<td>288</td>
<td>286</td>
<td>2</td>
</tr>
<tr>
<td>16</td>
<td>16</td>
<td>289</td>
<td>284</td>
<td>5</td>
</tr>
<tr>
<td>17</td>
<td>15</td>
<td>288</td>
<td>282</td>
<td>6</td>
</tr>
<tr>
<td>18</td>
<td>14</td>
<td>285</td>
<td>280</td>
<td>5</td>
</tr>
<tr>
<td>19</td>
<td>13</td>
<td>280</td>
<td>278</td>
<td>2</td>
</tr>
<tr>
<td>20</td>
<td>13</td>
<td>294</td>
<td>287</td>
<td>7</td>
</tr>
<tr>
<td>21</td>
<td>13</td>
<td>308</td>
<td>296</td>
<td>12</td>
</tr>
<tr>
<td>22</td>
<td>12</td>
<td>299</td>
<td>294</td>
<td>5</td>
</tr>
</tbody>
</table>

...entails bisecting the polygons by a line, and enumerating the number of polygons by moving the line ‘forward’ one step at a time. To count by length, all we need note is that the length is equal to the number of iterations or steps of the transfer matrix algorithm. This in fact makes these enumerations technically a little simpler than the original problem. The computed sequences of coefficients of $P(x)$, $I(x)$ are available from the author. They appear as A173408, A173409 in the OEIS [Slo].

3.2. ODEs for $P(x)$ and $I(x)$ modulo a prime. Starting with a series of 1000 terms for $P(x)$ and one of 1500 terms for $I(x)$, we were able to find underlying ODEs modulo the prime $p_r = 32749 = 2^{15} - 19$. In Table 1 we list the number of terms $K$ of each series, which were required in order to find an ODE of order $M$ and (minimal) degree $D$. One can easily check that the numbers follow the formula (2.4), the constants for $P(x)$ being $A = 9$, $B = 11$, $C = -36$, and for $I(x)$ being $A = 13$, $B = 13$, $C = -78$. However, for $I(x)$ the resulting ODE always has $P_0(x) = 0$, which means that in this case the minimal order $M$ equals not $B$ but rather $B + 1$, i.e., 14, and the ODE can be found using a reduced-size $M \times (D + 1)$ matrix. In the rightmost column of Table 1, we indicate in parentheses the reduced value of the deviation $\Delta$ if this option is used. For both $P(x)$ and $I(x)$, we clearly see the substantial decrease in the number of terms $K$ required to find the ‘optimal’ ODE ($K = 278$ and 481, respectively), as compared to the number of terms needed to find the minimal-order ODE ($K = 646$ and 1300). This illustrates the great utility of (2.4). From this formula we immediately get the minimal order, and we easily find the value of $K$ for the minimal-order and optimal ODEs. This is very important in minimizing computational effort.

The possible singular points of the differential equation are given by the roots of the head polynomial $P_M(x)$, which factors into a polynomial $Q(x)$ exhibiting the true singular points as well as (in most cases) a polynomial exhibiting apparent
ones. As mentioned earlier, most of the calculated ODEs (including the minimal-order ODE) have many apparent singular points. The true ones can be found easily by calculating ODEs of different orders, factoring the polynomials \( P_M(x) \) modulo \( p_r \), and then deducing the singularity polynomials from the common factors. For punctured staircase polygons, we find (using the prime \( p_r = 32749 = 2^{15} - 19 \)) that the common factors after factoring mod \( p_r \) are

\[
(x + 8187)^4(x + 32748)^2(x + 26199)(x + 24562)(x + 10234),
\]

where we recognize that since \(-1/4 \text{ mod } 32749 = 8187\), the first factor is just \((1 - 4x)^4\). Continuing in this way, one can easily deduce that the exact singularity polynomial is

\[
(3.2) \quad Q_P(x) = (1 - 16x)(1 - 5x)(1 - 4x)^4(1 - x)^2(1 + 4x).
\]

For imperfect staircase polygons we similarly find that

\[
(3.3) \quad Q_J(x) = (1 - 16x)^2(1 - 5x)(1 - 4x)^6(1 - x)^2(1 + 4x).
\]

The growth constant \( \mu \) is given by the reciprocal of the position of the singular point closest to the origin, \( x = 1/16 \). So the coefficients in the power series for \( \mathcal{P}(x) \) and \( \mathcal{J}(x) \) grow asymptotically as \( 16^n \), where \( n \) is ‘length’; whereas if one counts polygons by total perimeter \([GJ1, GJ2]\), the coefficients grow as \( 4^n \).

It is possible from the method of Frobenius \([In]\) to obtain from the indicial equation the characteristic exponents at the singular points. The indicial polynomial \( P_I(\lambda) \) at the singular point \( x = x_s \) can be written as

\[
(3.4) \quad P_I(\lambda) = \sum_{k=0}^{M} \lim_{x \to x_s} \frac{(x - x_s)^{M-k}P_k(x)}{P_M(x)} \prod_{j=1}^{k} (\lambda - j + 1),
\]

with the exponents at \( x_s \) obtained from the roots of \( P_I(\lambda) \). These calculations can be carried out using modular arithmetic, and we can thus find the exponents exactly (provided of course that they are integers or simple rational numbers). A simple \textsc{Maple} worksheet for calculating singular points and exponents for ODEs known modulo a prime is available from the author.

For example, for the minimal (order-11) ODE satisfied by \( \mathcal{P}(x) \), we find at the singular point \( x = 1/4 \) that the indicial polynomial factored modulo \( p_r \) is

\[
\lambda(\lambda + 1)(\lambda + 16373)(\lambda + 16374)(\lambda + 16375) \times (\lambda + 32743)(\lambda + 32744)(\lambda + 32745)(\lambda + 32746)(\lambda + 32747)(\lambda + 32748).
\]

Its roots are either integers or correspond to simple rational numbers; e.g., the factor \((\lambda + 16373)\) comes from \( \lambda = 3/2 \text{ mod } 32749 \). In Tables 2 and 3 we list the exponents of the minimal-order ODEs for \( \mathcal{P}(x) \) and \( \mathcal{J}(x) \).

The results presented in this section cannot be guaranteed to be correct. After all, the calculations have been performed modulo only a single prime. However, due to the simplicity of the results for the singularities and exponents it is unlikely (though not impossible) that they are incorrect. To obtain a higher degree of confidence in such results, one can repeat the calculations for a different prime; and if one gets the same results, it is hard to imagine that they are not correct. In the following subsection we go one step further and find one of the two ODEs in exact arithmetic, thus confirming our results beyond any reasonable doubt.
performed using modular arithmetic up to length 750, exactly. The numbers become very large, so the calculation was modulo several prime numbers digits, using an array of size 1300 Xeon processor. The calculation of the ODE was done with each prime used about 200MB of memory and 6 minutes of CPU time on a 2.8 GHz form were needed to represent the integer coefficients correctly. The calculation for thus have complete confidence that the modular method for finding and analysing and we found all the singular points and associated exponents to be correct. We tion took about 40 minutes and used some 450MB of memory. Having found the 1000 digit accuracy on a 2.5 GHz PowerPC G5 Macintosh computer. The calcula-

\begin{table}[h]
\centering
\caption{Exponents of the singular points of the minimal (order-11) Fuchsian ODE satisfied by $P(x)$.}
\begin{tabular}{|l|l|}
\hline
Singular point & Characteristic exponents \\
\hline
$x = 0$ & $1, 2, 3, 3, 10/3, 11/3, 4, 4, 5, 6, 8$ \\
$x = 1/16$ & $0, 1, 2, 3, 4, 5, 6, 13/2, 7, 8, 9$ \\
$x = 1/5$ & $0, 1/2, 1, 2, 3, 4, 5, 6, 7, 8, 9$ \\
$x = 1/4$ & $-1, -1/2, 0, 1/2, 1, 3/2, 2, 3, 4, 5, 6$ \\
$x = 1$ & $-2, -3/2, 0, 1, 2, 3, 4, 5, 6, 7, 8$ \\
$x = -1/4$ & $0, 1, 2, 3, 4, 5, 6, 13/2, 7, 8, 9$ \\
$x = \infty$ & $-3, -13/6, -2, -2, -11/6, -3/2, -5/4, -1, -3/4, 0, 1$ \\
\hline
\end{tabular}
\end{table}

\begin{table}[h]
\centering
\caption{Exponents of the singular points of the minimal (order-14) Fuchsian ODE satisfied by $J(x)$.}
\begin{tabular}{|l|l|}
\hline
Singular point & Characteristic exponents \\
\hline
$x = 0$ & $0, 1, 2, 2, 7/3, 8/3, 3, 3, 3, 4, 5, 6, 7, 8$ \\
$x = 1/16$ & $0, 1, 2, 3, 4, 5, 6, 13/2, 7, 8, 9, 10, 11$ \\
$x = 1/5$ & $0, 1/2, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12$ \\
$x = 1/4$ & $-1, -1/2, 0, 1/2, 1, 3/2, 2, 5/2, 3, 7/2, 4, 5, 6, 7$ \\
$x = 1$ & $-2, -3/2, -1, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10$ \\
$x = -1/4$ & $0, 1, 2, 3, 4, 5, 6, 13/2, 7, 8, 9, 10, 11, 12$ \\
$x = \infty$ & $-2, -3/2, -7/6, -1, -1, -5/6, -1/4, 0, 0, 1/4, 1, 2, 3, 4$ \\
\hline
\end{tabular}
\end{table}

3.3. The ODE for $P(x)$ in exact arithmetic. The minimal-order ODE for $P(x)$ is small enough that the original method for finding the ODE in exact arithmetic should work. So we calculated the number of punctured staircase polygons up to length 750, exactly. The numbers become very large, so the calculation was performed using modular arithmetic [Kn]. This involves performing the calculation modulo several prime numbers $p_i$ and then reconstructing the exact series coefficients at the end, by using the Chinese remainder theorem. We used primes of the form $p_i = 2^{30} - r_i$, where $r_i$ is a small positive integer. Almost 100 primes of this form were needed to represent the integer coefficients correctly. The calculation for each prime used about 200MB of memory and 6 minutes of CPU time on a 2.8 GHz Xeon processor. The calculation of the ODE was done with Mathematica using 1000 digit accuracy on a 2.5 GHz PowerPC G5 Macintosh computer. The calculation took about 40 minutes and used some 450MB of memory. Having found the exact ODE, we checked the results of the calculations summarized in Section 3.2; and we found all the singular points and associated exponents to be correct. We thus have complete confidence that the modular method for finding and analysing ODEs can yield exact results.

4. Reconstructing the exact ODE for $J(x)$ from modular results

Calculating the exact minimal-order ODE for $J(x)$ using the exact series coefficients would be a much more difficult task than it is for $P(x)$. Since the size of the coefficients grows as $10^k$, we would have to handle integers with some 1600 digits, using an array of size $1300^2$ to solve the set of linear equations arising from
FUCHSIAN DIFFERENTIAL EQUATIONS FROM MODULAR ARITHMETIC

equation (2.2). We would thus expect to use more than four times the memory used to find the ODE for \( \mathcal{P}(x) \), and an even larger multiple of the CPU time (and 1000 digits would, most likely, not provide sufficient accuracy to perform the floating-point calculations). So, such a calculation would probably stretch the capacity of that type of algorithm beyond its limit. Instead, we decided to use a different and (as we shall see) much more efficient approach. It is possible to reconstruct the exact ODE using the results from several mod-prime calculations. Depending on the precise approach taken, the ODE can be reconstructed exactly: using 18 primes in one case, but only 10 primes using a more efficient and better informed approach. Here we schematically outline the procedure for finding the exact minimal-order ODE. A MAPLE worksheet is available from the author.

**Procedure for ODE reconstruction:**

1. Generate a long series modulo a single prime.
2. Find ODEs of different orders and identify the constants \( A, B, \) and \( C \) of (2.4).
3. Use this formula to identify both the minimal-order ODE and the optimal ODE requiring the least number of terms.
4. Generate series for more primes \( p_i \), long enough to find the optimal ODEs.
5. Turn these ODEs into recurrences and generate longer series.
6. Use these series to find the minimal-order ODEs \( \text{mod} p_i \).
7. Combine these modular results to obtain the exact minimal-order ODE, by using an algorithm for rational reconstruction [Wa, Mo, CE] to calculate the rational coefficients \( a_{ij} \) in (2.2).

In step (7) we used the built-in MAPLE routine `iratrecon`, which is based on work of Wang and Monagan [Wa, Mo]. This requires us first to use the Chinese remainder theorem to reconstruct the integers \( b_{ij} := a_{ij} \text{ mod } P \), where \( P = \prod p_i \). The exact rational coefficients \( a_{ij} \) can then be calculated via the call \( a_{ij} = \text{iratrecon}(b_{ij}, P) \). In this fashion one can find the rational number \( a_{ij} = r/s \), provided that \( 2 |r| s \leq P \).

We managed to reconstruct the exact ODE for \( \mathcal{I}(x) \) using 18 primes of the form \( p_j = 2^{10} - r_j \). Reconstructing the exact series coefficients using the Chinese remainder theorem up to the length needed to find the exact ODE by the original approach would have required about 10 times as many primes as the method used above. This is because in general, \( P = \prod p_i \) must be larger than any of the integer coefficients we are trying to reconstruct; and since the coefficients in the series for \( \mathcal{I}(x) \) grow as \( 16^n \), the number of primes required to reconstruct a series of length \( N \) would be \( 4N/30 \), or about 174 for \( N = 1300 \). We note that it only takes a few minutes to find the ODE modulo any given prime, and it also takes only a few minutes to reconstruct the exact ODE coefficient from the mod-prime ODEs.

Fewer than 18 primes are actually needed. In all the problems we have studied, the numerators happen to be much larger than the denominators. This means that we can modify the call to be \( r/s = a_{ij} = \text{iratrecon}(b_{ij}, P, R, S) \), where \( R \) and \( S \) are positive integers such that \( |r| \leq R \) and \( 0 < s \leq S \), with \( 2RS \leq P \). If we assume that \( s < \sqrt{r} \) then we may choose \( S = \sqrt{P} \) and \( R = P/(2S) \), and we can then find the \( a_{ij} \) using only 12 primes. This latter refinement clearly relies on the empirical observation that the denominators tend to be quite small, and this obviously need not be the case for all problems.
A further refinement is possible by generating the $a_{ij}$ starting from $a_{MD}$. We then multiply all the residues by the denominator of $a_{MD}$ modulo the respective primes. We go through the remaining coefficients by decreasing first $j$ so as to generate all $a_{Mj}$. Whenever a non-integer rational number is encountered, we multiply all residues by its denominator (after this we found that the only remaining denominator was 9). We then repeat for $i = M - 1$, and so on until all $a_{ij}$ have been exhausted. After this, the modified residues for the $a_{ij}$ will be representations of integer coefficients, which we then reconstruct. This procedure can generate the exact integer coefficients of the ODE using only 10 primes.

We note that for $J(x)$, this new procedure for finding the exact minimal-order ODE, based on multiple mod-prime calculations, is at least 1000 times faster than the original procedure described in Section 2 and used in Section 3.3 to find the exact ODE for $P(x)$.

5. Factoring differential operators modulo a prime

Finally, we demonstrate how modular calculations can be used to factor linear differential operators. We use a method developed in [BBGH], which is similar to a method proposed by van Hoeij in [vHo]. The basic approach is to ‘follow’ the series associated to a specific characteristic exponent at a given singular point. Linear combinations of series with different exponents can be studied as well. The modular nature of our calculation is of great help in this, since with the series being known modulo a prime, the coefficient in the linear combination can take only a finite number of integer values; so ‘guessing’ the correct combination can be done by exhaustive search. For each series used as a candidate to ‘break’ the differential operator under consideration, we compute three (or more) ODEs, and from the ODE formula (2.4) we infer the minimal order.

For an ODE of minimal order $m$, let $L_m$ denote the corresponding operator (henceforth the subscript on an operator will always denote its order). Consider a singular point $x = x_s$, and assume the exponents at this point are

$$\lambda_1^{k_1} < \lambda_2^{k_2} < \cdots < \lambda_p^{k_p}, \quad \sum_{j=1}^p k_j = m,$$

where the superscript $k_j$ on an exponent $\lambda_j$ simply denotes its multiplicity. In our cases the exponents are either integers or rational numbers. Here we utilize only solutions which are analytic at the expansion point (say, $x = x_s = 0$). So in what follows we consider only integer exponents, and we denote them by $n_j$ (with $n_1 < n_2 < \cdots < n_p$). Focusing on the highest exponent $n_p$, let us plug the series

$$S_p(x) = x^{n_p} + \sum_{k=n_p+1}^{\infty} a_k x^k$$

into the ODE. Requiring that $L_m(S_p(x)) = 0$ will determine all coefficients $a_k$. After calculating sufficiently many terms, we can find a new ODE satisfied by the particular series solution $S_p(x)$, which is by construction a solution of $L_m$. This new ODE will either have order $m$ or have order $m_1 < m$. In the first case, either $L_m$ is irreducible or the factor ‘responsible’ for annihilating the solution $S_p(x)$ is not the rightmost factor of $L_m$. In the second case, we must have a factorization

$$L_m = L_{m-m_1} \cdot L_{m_1}.$$
To summarize, the series associated to the highest exponent leads either to the original ODE (in the first case), or to a ‘breaking’ of it (in the second).

If the series $S_{p-1}(x)$ leads to the original ODE, we turn to the second-highest exponent $n_{p-1}$. In this case, a series $S_{p-1}(x)$ starting with $x^{n_{p-1}} + \cdots$, plugged into the original ODE, i.e., $L_m(S_{p-1}(x)) = 0$, will yield the expansion

\[(5.4) \quad S_{p-1}(x) = x^{n_{p-1}} + \sum_{k=n_{p-1}+1}^{n_{p-1}+1} a_k x^k + a_{n_p} x^{n_p} + \sum_{k=n_{p-1}+1}^{\infty} c_k x^k,\]

where all $a_k$ with $k < p$ are fixed and the $c_k$’s depend linearly on the free coefficient $a_{n_p}$, i.e., $S_{p-1}$ is a one-parameter solution. This series $S_{p-1}(x)$ is a sum of a series starting with $x^{n_{p-1}} + \cdots$ and the series $a_{n_p} S_p(x)$, and is thus a linear combination of two formal series solutions, starting with $x^{n_{p-1}} + \cdots$ and $x^{n_p} + \cdots$, respectively. For generic values of the rational coefficient $a_{n_p}$, the series $S_{p-1}(x)$ will give rise to the original ODE. But for some values of the coefficient $a_{n_p}$, $S_{p-1}(x)$ may be the solution of an ODE of order less than $m$. If so, a factorization of $L_m$ will be obtained.

To demonstrate how this procedure works in practice, we consider the order-11 operator of the minimal-order ODE for $P(x)$ (denoted $L_{11}$). In this case we are fortunate to have access to both the exact ODE (through the calculation summarized in Section 3.3), and the exact series to any required length (in linear time). It is thus not a problem to generate the mod-prime series for any prime $p_r$ we want. To find a ‘breaking’ linear combination we first form the series $S_{p-1}(x)$ for a given value of the free rational coefficient $a_{n_p}$, and then find the ODE annihilating $S_{p-1}(x)$. This procedure is carried out for all possible values of $a_{n_p}$; but since our calculations are done modulo $p_r$, this just means for all values in the interval $[1, p_r - 1]$. So when searching for a ‘breaking’ linear combination it is advisable to take the prime to be quite small, thus limiting the number of times the ODE solver must be used. (We use primes just above 1000 for this.) However, when reconstructing any factor exactly, larger primes are preferred, thus limiting the number of mod-prime ODEs required to calculate the factor in exact arithmetic. (In practice, as above we use primes just below $2^{36}$ for this.)

As displayed in Table 2, at the singular point $x = 0$ the exponents of the minimal ODE satisfied by $P(x)$ are, with multiplicity,

\[(5.5) \quad 1, 2, 3, 10/3, 11/3, 4, 5, 6, 8.\]

Substituting into the ODE the series that starts with $x^8 + \cdots$, i.e.,

\[(5.6) \quad S_8(x) = x^8 + \sum_{k=9}^{\infty} a_k x^k,\]

will determine all its coefficients. In fact, this procedure yields the expansion at $x = 0$ of $P(x)$, thus leading to the original, minimal-order ODE. Of course this is not surprising, since the series for $P(x)$, used to ‘generate’ the differential operator $L_{11}$, starts with $x^8$. The unique series $S_8(x)$ must be $P(x)$ itself. So in this case, the series associated to the highest exponent cannot be used to ‘break’ $L_{11}$, since $L_{11}$ is the minimal-order operator annihilating $P(x)$.
Consider next a series that starts with $x^6 + \cdots$, i.e., is associated to the second-highest exponent, namely
\begin{equation}
S(x) = x^6 + \sum_{k=7}^{\infty} a_k x^k.
\end{equation}
We insert this series into the exact ODE for $P(x)$ and then solve (term by term) the equations arising from $L_{11}(S(x)) = 0$. Doing this, we find that the coefficient $a_7$ is fixed, while the coefficient $a_8$ is undetermined and hence enters the series as a free parameter. The remaining coefficients are all expressed in terms of $a_8$. The resulting series is, after multiplication by 996 to ensure integer coefficients,
\begin{align*}
S(x) &= 996x^6 + 6915x^7 + 996a_8x^8 + (13944a_8 - 333315)x^9 \\
&\quad + (126492a_8 - 3797208)x^{10} + (954168a_8 - 31074358)x^{11} + \ldots.
\end{align*}
The terms in $S(x)$ proportional to the free coefficient $a_8$ are the coefficients of the series $996S_8(x) = 996P(x)$. We define $S_6(x)$ to be the series obtained from $S(x)$ by setting $a_8 = 0$. In order to break the operator $L_{11}$, we look at linear combinations $S_\alpha(x) = S_8(x) + \alpha S_6(x)$, where $\alpha$ is a rational number. For generic values of $\alpha$ the series $S_\alpha(x)$ is annihilated only by the full ODE of order 11. However, it is possible that for special values of $\alpha$ the series $S_\alpha(x)$ may be the solution of an ODE of order less than 11.

We do not know how such ‘splitting’ values of $\alpha$ can be obtained, except by exhaustive search. The use of modular calculations is very useful in searching for splitting values. The series $S_8(x)$ and $S_6(x)$ can be obtained modulo any prime $p_r$, and in the modular calculations $\alpha$ will thus take a value in the finite range $[1, p_r - 1]$. If a rational splitting value of $\alpha$ exists, it can be found by looking for an underlying ODE of order less than 11 annihilating the series $S_\alpha(x)$. In the search we used the ‘optimal’ ODE, which (from Table 1) is of order 19 and degree 13, with $K = 278$. We used the prime $p_r = 1009 = 2^{10} - 15$ in our search. For each $\alpha \in [1, 1009]$ we calculated the series modulo $p_r$, and then looked for an annihilating ODE of order 19 and degree 13. For any value of $\alpha$ such an ODE exists, and for almost all values, $K = 278$. However, for the special values $\alpha = 18$, 132, and 962, we have $K = 274, 256,$ and 138, respectively. The decrease in $K$ is a sure sign that a simpler ODE annihilates $S_\alpha(x)$. In this particular case we find that the ODEs for $\alpha = 18$ and 132 are of order 10, while for $\alpha = 962$ the ODE is of order 3.

Working with the optimal ODE rather than the minimal-order ODE may appear counterintuitive, since we are attempting to find a factorization of $L_{11}$ and not the order-19 operator $O_{19}$ of the optimal ODE. However, in [BBGH] it is demonstrated that one can work with non-minimal-order ODEs because the minimal-order ODE is a right factor of the non-minimal-order ODE. In our case we have that there exists an 8th order operator $O_8$ such that $O_{19} = O_8 \cdot L_{11}$, and hence any right factor of $O_{19}$ is also a right factor of $L_{11}$. We can confirm that the procedure works, by directly checking that the various factors obtained in exact arithmetic do indeed right-divide $L_{11}$.

We focus here only on the case $\alpha = 962$. The fact that the series is annihilated by a 3rd order operator means that we have found a right factor of $L_{11}$ of order 3, and we thus have that $L_{11} = L_8 \cdot L_3$, where $L_8$ and $L_3$ are differential operators of orders 8 and 3, respectively. Going through the same procedure for a few more primes (5 in all), we find that in fact, $\alpha = 1/36646$. Having determined $\alpha$ exactly,
we then proceed to produce series for $S_\alpha(x)$ modulo several primes of the form $p_j = 2^{30} - r_j$, and calculate the minimal-order ODE and reconstruct $L_3$ exactly using the procedure described in Section 4, obtaining

\[(5.8)\]

\[L_3 = \sum_{i=0}^{3} P_i(x) \left( \frac{d}{dx} \right)^i\]

with

\[(5.9a)\]

\[P_3 = x^3(1 - 4x)^2(1 - x)^2(6640 - 115940x + 745572x^2 - 160044x^3 - 3904387x^4 + 32245387x^5 - 81483743x^6 + 102350592x^7 - 63271990x^8 + 15728047x^9 - 1896930x^{10}),\]

\[(5.9b)\]

\[P_2 = x^2(1 - 4x)(1 - x)(-39840 + 997500x - 10171096x^2 + 49748712x^3 - 65361298x^4 - 592676545x^5 + 3944925464x^6 - 12014127235x^7 + 21682463258x^8 - 23961110611x^9 + 15690708326x^{10} - 5774145597x^{11} + 1279604150x^{12} - 132785100x^{13}),\]

\[(5.9c)\]

\[P_1 = 2x(39840 - 1141920x + 13498966x^2 - 77265957x^3 + 109315350x^4 + 1533773209x^5 - 12778856043x^6 + 52423258998x^7 - 136624376823x^8 + 238960509306x^9 - 281352576354x^{10} + 218137005338x^{11} - 108012316206x^{12} + 34030365096x^{13} - 6798582940x^{14} + 607017600x^{15}),\]

\[(5.9d)\]

\[P_0 = 79680 + 2283840x - 26828612x^2 + 149243694x^3 - 144204350x^4 - 3690114536x^5 + 28993250188x^6 - 117936034106x^7 + 308125615678x^8 - 542161753964x^9 + 642034367908x^{10} - 499556504280x^{11} + 248708837380x^{12} - 80255885640x^{13} + 16105328880x^{14} - 1365789600x^{15}.\]

Next we use the command `rightdivision` from the MAPLE package `DEtools` to find exactly the factor $L_8 = \text{rightdivision}(L_{11}, L_3)$, and we then calculate the exponents of $L_8$ at the singularities of $L_{11}$. They are listed in Table 4. It should be noted firstly that the ODE corresponding to $L_8$ does not seem to be singular at $x = 1/5$ and $x = -1$. This can be confirmed by factoring the head polynomial of $L_8$, so $L_3$ is solely responsible for the singular behavior of $L_{11}$ at these points. Secondly, it should be noted that all exponents of $L_8$ are rational.

The ODE corresponding to $L_3$ can be solved exactly, and it has the three independent solutions

\[(5.10a)\]

\[F_1(x) = \frac{210x - 1718x^2 + 4271x^3 - 2836x^4 - 914x^5 + 1392x^6}{(1 - 4x)(1 - x)^2},\]

\[(5.10b)\]

\[F_2(x) = \frac{-2x + 10x^2 - 17x^3 + 18x^4}{\sqrt{1 - 4x}(1 - x)^2},\]

\[(5.10c)\]

\[F_3(x) = \frac{\cos(3 \arctan(1 - 5x))}{(1 - 4x)(1 - x)^{3/2} \sqrt{1 - 5x}}.\]
Having found these simple algebraic solutions it is natural for us to check if they appear as part of a direct sum decomposition of \( L_{11} \). We do this by forming the series for the linear combinations \( \mathcal{P}(x) + \alpha_l F_i(x) \), and then doing a search as above, to find the value (if any) of \( \alpha_l \), which makes the linear combination a solution of a lower order (actually order-10) ODE. We find \( \alpha_1 = 1/253 \) and \( \alpha_2 = 1/6 \), but we find no value \( \alpha_3 \); so perhaps \( \alpha_3 \) is not a rational number. We note that in addition to the solutions (5.10abc), \( L_{11} \) also has two simple polynomial solutions

\[
(5.11) \quad Q_1(x) = x(1 - 16x^2) \quad \text{and} \quad Q_2(x) = x^2(1 - 4x).
\]

Now, let \( L_9 \) denote the order-9 differential operator annihilating the ‘reduced’ series \( \mathcal{P}_R(x) = \mathcal{P}(x) + F_1(x)/253 + F_2(x)/6 \). The exponents at \( x = 0 \) of the corresponding ODE are

\[
(5.12) \quad 1, 2, 3^2, 10/3, 11/3, 4^2, 5.
\]

By following the two solutions with exponents 5 and 4 we find a value of the combination coefficients that splits \( L_9 \) into a product of operators of order 7 and 2, \( L_9 = L_7 \cdot L_2 \). The operator \( L_2 \) can be found exactly using the method of Section 4, and hence solved. However, the two solutions are just \( F_3(x) \) and \( Q_1(x) - 3Q_2(x) \).

By acting with \( L_2 \) on \( \mathcal{P}_R(x) \), we produce a series that is annihilated by the differential operator \( L_7 \). The exponents of \( L_7 \) at \( x = 0 \) are

\[
(5.13) \quad 2, 3^2, 10/3, 11/3, 4, 5.
\]

Again we try to use the two highest exponents to split \( L_7 \), but this does not work in this case. When we plug the series starting with \( x^4 + \cdots \) into the ODE, the coefficient of \( x^5 \) is a constant, so this series cannot be annihilated by the ODE. This is probably because the exponent 4 has multiplicity 2 in \( L_{11} \) (see Table 2), so there must be a solution of the form \( x^4 \log(x)(1 + a_1 x + \cdots) \); and this logarithmic solution survives in \( L_7 \), which is why there is no solution of the form \( x^4 + a_5 x^5 + \cdots \). Instead, we try using the series starting with \( x^3 + \cdots \), and with this we find an order-1 factor such that \( L_7 = L_6 \cdot F_1^{(a)} \), and going one step further we find a second order-1 factor so that \( L_6 = L_5 \cdot L_1^{(b)} \). The ODE corresponding to \( L_1^{(a)} \) has the exact solution

\[
(5.14) \quad F(x) = \frac{x \mathcal{P}(x)}{(1 - x)^3}.
\]

<table>
<thead>
<tr>
<th>Singular point</th>
<th>Characteristic exponents</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x = 0 )</td>
<td>3, 3, 10/3, 11/3, 4, 5, 8</td>
</tr>
<tr>
<td>( x = 1/16 )</td>
<td>0, 1, 2, 3, 7/3, 4, 5, 6</td>
</tr>
<tr>
<td>( x = 1/5 )</td>
<td>0, 1, 2, 3, 4, 5, 6, 7</td>
</tr>
<tr>
<td>( x = 1/4 )</td>
<td>-1/2, 0, 1/2, 1, 2, 3, 4, 5</td>
</tr>
<tr>
<td>( x = 1 )</td>
<td>0, 1, 2, 3, 4, 5, 6, 7</td>
</tr>
<tr>
<td>( x = -1/4 )</td>
<td>0, 1, 2, 3, 7/2, 4, 5, 6</td>
</tr>
<tr>
<td>( x = \infty )</td>
<td>-103/6, -17, -101/6, -65/4, -16, -63/4, -15, -14</td>
</tr>
</tbody>
</table>

Table 4. Exponents of the left factor \( L_8 \) of \( L_{11} \) (the order-11 operator in the minimal Fuchsian ODE satisfied by \( \mathcal{P}(x) \)), at the singular points of \( L_{11} \).
with
\[ P(x) = 1 - 23x + 281x^2 - 2595x^3 + 16426x^4 - 63202x^5 + 140322x^6 - 171410x^7 + 101936x^8 - 15904x^9. \]

Now \( F(x) \) is not a solution of \( L_{11} \), because the factor \( L_4^{(a)} \) is not a right factor of \( L_9 \). However, we can use it to construct a solution \( F_4(x) = x^3 + a_4x^4 + \cdots \) by solving \( L_2(F_4(x)) = F(x) \), term by term. We find that in \( L_2(F_4(x)) \) the coefficient \( a_4 \) is a free parameter, and by fixing it at the value \( a_4 = -9/2 \) we obtain a new first order ODE, the solution of which,

\[
(5.15) \quad F_4(x) = \frac{x^2(1 - 6x - 27x^2 + 140x^3 - 36x^4)}{(1 - 4x)(1 - x)^2},
\]
is linearly independent of the previous solutions. Similarly, we find that \( L_4^{(b)} \) has a polynomial solution, which turns out to be just the previous solution \( Q_2(x) \).

We find that the remaining operator \( L_5 \) cannot be factored by the method used above. To proceed further (if possible), we would have to use solutions associated to non-integer exponents at \( x = 0 \), or use expansions about other singularities such as \( x = 1/4 \) or \( x = 1/16 \). But this is work which we shall leave for the future.

6. Conclusion and outlook

In this article we have reported on recent progress in the problem of finding, numerically, the exact ODE underlying a series expansion of a given combinatorial generating function. We have shown how an ODE can be obtained from a series that is known modulo a single prime, and how the singular points and characteristic exponents can be obtained exactly (provided they are integers or simple rational numbers) from an ODE known modulo a single prime. The obvious advantage of this approach is that we need only calculate the series modulo one prime, thus saving a lot of CPU time; and opening up the possibility of studying large and complicated combinatorial problems such as that of the \( \chi^{(5)} \) contribution to the susceptibility of the square lattice Ising model [BBGH, BGHJ].

We described in Section 4 a procedure for the efficient reconstruction of the exact ODE using results from multiple mod-prime calculations. For the particular problem of the generating function \( \mathcal{J}(x) \) of imperfect staircase polygons, this approach is a factor of about 1000 times faster than first trying to generate the exact series and then finding the exact minimal-order ODE.

Additionally, we demonstrated in Section 5 how modular calculations can be used to factor linear differential operators. We showed that the differential operator \( L_{11} \) in the minimal ODE satisfied by the generating function \( \mathcal{P}(x) \) can be factored as \( L_{11} = L_8 \cdot L_3 \) with \( L_3 \) being exactly solved, thus yielding three exact solutions. Two of these solutions can be subtracted from \( \mathcal{P}(x) \) to leave us with a reduced series \( \mathcal{P}_R(x) = \mathcal{P}(x) + F_1(x)/253 + F_2(x)/6 \), which is the solution of an ODE based on a 9th order operator \( L_9 \), which in turn can be factored as \( L_9 = L_5 \cdot L_4^{(b)} \cdot L_4^{(a)} \cdot L_2 \). It is worth noting that \( L_9 \) is a right factor of \( L_{11} \). The factorization carried out in Section 5 utilized only solutions analytic at \( x = 0 \), i.e., local solutions at \( x = 0 \) associated to the exponent zero. In future work we plan to extend the study of the ODEs for \( \mathcal{J}(x) \) and \( \mathcal{P}(x) \) by looking at local solutions associated to nonzero exponents, and local solutions around other singular points. It
may also be possible to utilize logarithmic solutions, such as those at $x = 0$, where two of the exponents of the ODE for $P(x)$ have multiplicity two. (See Table 2.)

In this article we wanted to demonstrate how modular calculations can be used to speed up the search for the linear combinations of local solutions that split the differential operator into simpler factors. This may not be the most efficient approach. For example, for $P(x)$ the only local solution at $x = 1/5$ that is singular, i.e., the only one that is associated to a non-integer exponent, starts with $(1 - 5x)^{1/2}$, and if we were to ‘follow’ this solution we would most likely find $L_3$ (or a simpler factor) directly, without the need for any guesswork. The approach of following singular local solutions has been used by B. Nickel to find many factors of the operator annihilating $\tilde{\chi}^{(5)}$ (several of these are reported in [BBGH]), and these factors were obtained without resorting to the type of guessing used in Section 5.

In Section 5, we had the luxury of being able to work with the exact minimal-order ODE satisfied by $P(x)$, and thus with exactly known series. In the article [BBGH] the ODE for $\tilde{\chi}^{(5)}$ was only calculated modulo a few primes (and only one was really used for most calculations); and since the minimal-order ODE for $\tilde{\chi}^{(5)}$ requires almost 50000 terms while the ‘optimal’ ODE requires about 7400, it was impossible to work directly with the former. (Indeed, we would never have been able even to generate directly a series of 50000 terms.) Instead, this minimal-order ODE was always represented in the calculations by the optimal ODE or a nearly optimal one, and the factorization was carried out on these higher order ODEs, with the minimal order inferred from (2.4). These calculations of [BBGH] demonstrated the important fact that one can work at the level which is computationally most efficient.

The use of modular calculations as outlined in this article thus promises to make it possible to find exact solutions through numerical means to many combinatorial problems which have hitherto been considered too computationally difficult. The only real drawback (if one can call it that) is that a priori there is generally no way of knowing whether or not a series is a solution of an underlying (Fuchsian) ODE; and if so, how many terms one would need to uncover it.

Acknowledgements. The author would like to thank his co-workers S. Boukraa, A. J. Guttmann, S. Hassani, B. Nickel, J.-M. Maillard and N. Zenine for their many and varied contributions to this research. The calculations performed for this work used the computational resources of the Australian Partnership for Advanced Computing as well as the Victorian Partnership for Advanced Computing.

References


FUCHSIAN DIFFERENTIAL EQUATIONS FROM MODULAR ARITHMETIC


DEPARTMENT OF MATHEMATICS AND STATISTICS, THE UNIVERSITY OF MELBOURNE, VICTORIA 3010, AUSTRALIA

E-mail address: I.Jensen@ms.unimelb.edu.au

URL: http://www.ms.unimelb.edu.au/~iwan

This is a free offprint provided to the author by the publisher. Copyright restrictions may apply.