

## PARTITIONS APPROXIMATED BY FINITE COSINE-SERIES

HARVEY DUBNER

**ABSTRACT.** By using a specialized numerical Laplace transform inversion technique the number of partitions are expressed by a finite cosine-series. The accuracy of the results is only limited by the accuracy of computation and available computer time. The method is general and is applicable to all generating functions.

### 1. INTRODUCTION

In the late 1960s two papers appeared about numerical inversion of Laplace transforms [3, 1]. Both papers were aimed at the engineering and probability world, where functions encountered were reasonably simple and 6-digit accuracy was more than adequate. Now, with the ready availability of powerful PC computers and excellent public-domain software, it was decided to reevaluate the utility of the power series expansion technique by applying it to the complicated generating functions found in partition theory.

The number of partitions can be expressed as a finite cosine-series with the accuracy of the results being limited only by the accuracy of computation. Because of this inherent accuracy it seems probable that there is some underlying theory yet to be discovered which would explain this behavior.

### 2. OUTLINE OF THE METHOD

Our objective is to find the coefficients of the power series represented by a given generating function. The outline of the theory of [1] which we will use is as follows:

1. From a given generating function  $f(z)$  we define an auxiliary function  $g(t)$  such that for integral values of  $t$ ,  $g(n)$  gives the coefficients of the power series of the generating function.
2. The Laplace transform,  $G(s)$ , of  $g(t)$  is obtained trivially from  $g(t)$ .
3. The inverse Laplace transform,  $g(t)$ , of  $G(s)$  is approximated by an infinite cosine-series.
4. Since only integral values of  $t$  are needed, the infinite cosine-series can be converted to a finite cosine-series by using various symmetry and periodicity properties and introducing the generalized Riemann zeta function.

---

Received January 15, 1991; revised March 28, 1991.

1991 *Mathematics Subject Classification.* Primary 11P83; Secondary 41A58.

©1992 American Mathematical Society  
0025-5718/92 \$1.00 + \$.25 per page

5. Although the Riemann zeta function consists of an infinite number of terms, it is expandable asymptotically so that it can be calculated efficiently to a high degree of accuracy.

6. This finite cosine-series can be used to numerically approximate the coefficients in the power series of the generating function. The accuracy of the results is completely determined by the arithmetic accuracy of the computations and by the time available for the calculations.

### 3. LAPLACE TRANSFORM OF A GENERATING FUNCTION

We are given a generating function  $f(z)$  which has a power series expansion,

$$(1) \quad f(z) = \sum_{n=0}^{\infty} A(n)z^n.$$

Define a continuous function  $g(t)$  such that  $g(n) = A(n-1)$  with a straight line between  $g(n)$  and  $g(n+1)$ , and  $g(t) = 0$  for  $t < 0$ . The Laplace transform of  $g(t)$  is

$$(2) \quad G(s) = s^{-2}(1 - e^{-s})^2 \sum_{n=0}^{\infty} A(n)e^{-ns}.$$

Perhaps the easiest way to see that  $G(s)$  is the Laplace transform of  $g(t)$  is to consider  $g(t)$  as the sum of a series of triangles with coordinates  $(n, 0)$ ,  $(n+1, A(n))$ ,  $(n+2, 0)$ . The Laplace transform of each triangle is easy to derive, and (2) is the sum of these transforms. However, from (1),

$$\sum_{n=0}^{\infty} A(n)e^{-ns} = f(e^{-s});$$

therefore,

$$(3) \quad G(s) = s^{-2}(1 - e^{-s})^2 f(e^{-s}).$$

This determines the Laplace transform of  $g(t)$  trivially from the given function  $f(z)$  by the transformation  $z = \exp(-s)$ . Then, having obtained  $G(s)$ , we invert it to find the function  $g(t)$ , which by virtue of its definition determines the coefficients

$$(4) \quad A(n) = g(n+1).$$

### 4. INVERTING THE LAPLACE TRANSFORM

For generating functions encountered in combinatorial theory it is almost always impossible to find an exact inversion for (3). However, the methods of [3, 1] can be used to numerically invert these functions, but the question is whether the accuracy of the method is adequate when applied to complicated partition functions. What follows is a short summary of the inversion theory.

An approximation formula for the Laplace transform inversion integral is given by the following infinite series:

$$(5) \quad g(t) = \frac{2e^{at}}{N} \left[ \frac{1}{2} \operatorname{Re}\{G(a)\} + \sum_{n=0}^{\infty} \operatorname{Re} \left\{ G \left( a + \frac{n\pi i}{N} \right) \right\} \cos \frac{n\pi}{N} t \right].$$

The error which results from replacing the integral by the infinite series is of the order  $\exp(-aN)$ , provided we restrict  $t$  to the interval  $0 < t < N/2$ . In practice, selecting the largest value of  $t$  determines  $N$ , which together with the desired error limit determines  $a$ .

There still is the difficulty that the series extends to infinity. Truncation error can be large and is usually beyond our control. However, by only requiring answers for integral values of  $t$  and using various periodicity properties, the infinite series can actually be summed completely by making use of the generalized Riemann zeta function.

The relevant equations are

$$(6) \quad g(m) = \frac{2 \exp(am)}{N} \sum_{n=0}^{2N-1} C_n \cos \frac{n\pi m}{N},$$

$$(7) \quad s = a + \frac{n\pi}{N} i,$$

$$(8) \quad C_n = \operatorname{Re}[S_n(a, N)(1 - e^{-s})^2 f(e^{-s})],$$

$$(9) \quad S_n(a, N) = \sum_{k=0}^{\infty} \frac{1}{(s + 2k\pi i)^2} - \frac{\delta_{0n}}{2a^2},$$

where  $\delta_{0n}$  is the Kronecker delta function,

$$(10) \quad N \geq 2m,$$

and

$$(11) \quad a = \frac{E}{N}.$$

In [1] it is shown how (9) is related to the generalized Riemann zeta function, which in turn can be expanded in an asymptotic series involving the Bernoulli numbers,

$$(12) \quad z(2, w) = \sum_{k=0}^{L-1} \frac{1}{(k+w)^2} + \frac{1}{(L+w)} + \frac{1}{2(L+w)^2} + \frac{1}{6(L+w)^3} \\ - \frac{1}{30(L+w)^5} + \frac{1}{42(L+w)^7} \pm \dots - \frac{3617}{510(L+w)^{17}}$$

with an error in the order of  $50/(L+w)^{19}$ . For  $L = 1000$ , this gives the generalized zeta function to about 55 digits while only requiring the summation of about 1000 terms. Equation (9) can be rewritten as

$$(13) \quad S_n(a, N) = -\frac{z(2, w)}{4\pi^2} - \frac{\delta_{0n}}{2a^2},$$

where

$$(14) \quad w = \frac{n}{2N} - \frac{a}{2\pi} i.$$

In practice,  $N$  is selected based on (9);  $E$  and  $L$  are chosen based on error considerations (see below). Equation (8), with (11), (7), (12), (13), and (14), is used to calculate the coefficients  $C_n$  of the cosine-series. Equations (6) and

(4) are used to calculate the coefficients  $A(n)$  of the power series of the given generating function.

### 5. ACCURACY OF CALCULATION

There are three sources of error in the above calculations:

1. *The theoretical error inherent in the inversion method.* One of the strong points of the method is its tight error control. This error is about  $\exp(-E)$ , and there is no theoretical limit for  $E$ . For  $E = 70$ , the error is about  $10^{-30}$ .

2. *The error in calculating the generalized zeta function.* As shown above, this error can be kept quite small without using excessive computer time.

3. *Error due to arithmetic precision limitations.* This method requires high-precision calculations on complex numbers. Fortunately, there is a public-domain program for PC's, called UBASIC, ideally suited to this problem. It can process up to 2600-digit numbers, and no special handling is necessary for the complex domain. It is also aimed at the number theory world with appropriate built-in functions. For a discussion of UBASIC see [2]. Since any desired precision is available, this error can be ignored.

In theory,  $E$  can be made as large as one wants, making the error due to this source effectively zero. This would indeed be true except that there is an interaction between errors. In (6), the multiplying term  $\exp(am) = \exp(Em/N)$  grows exponentially with  $E$ . Since the answer must remain virtually the same as  $E$  grows, the cosine-series sum must decrease exponentially with  $E$ . Thus, the errors introduced by the zeta function become proportionately larger.

Searching for maximums resulted in the following table based on the accuracy of the term corresponding to  $m = N/2$  in (6):

$E$	$L$	Accuracy
70	100	25 digits
90	500	34 digits
105	2000	39 digits
120	6000	45 digits

That is, if 25-digit accuracy were required then  $L = 100$ , and  $E = 70$  would be an optimum choice. For  $L = 100$ , increasing  $E$  to 100 decreases the accuracy to 19 digits. Dropping  $E$  to 50 decreases the accuracy to 18 digits.

### 6. ATTENUATION

When experimenting with this method, it soon becomes apparent that more accuracy is obtained when the target coefficients are more uniform. Often, a simple transformation can add several digits of accuracy with little computational penalty. In (1), replace  $z$  by  $tz$ , so that

$$(15) \quad f'(z) = \sum_{n=0}^{\infty} [A(n)t^n]z^n.$$

For  $t < 1$ , the new coefficient in brackets has been attenuated by the factor  $t^n$ . To compensate,  $g(m)$  in (6) must be divided by  $t^n$ . One approach is to do a

fast calculation at low accuracy, determine  $t$  so that the answers will be more uniform, then do the final, more accurate calculation.

7. UNRESTRICTED PARTITIONS

The generating function for the number of unrestricted partitions of  $n$  is [4, pp. 273–296],

$$(16) \quad F(x) = \frac{1}{(1-x)(1-x^2)(1-x^3)\dots} = 1 + \sum_1^{\infty} p(n)x^n.$$

If the maximum  $n$  of interest is  $m$ , then the infinite product in the denominator can be terminated after  $m$  terms. The degree of this polynomial is  $m(m-1)/2$ . For  $m = 500$ , the degree is about 125,000! We applied our theory to compute  $p(n)$  up to 500.

From  $m = 500$ ,  $N$  should be 1000, and therefore 2000 terms in the cosine-series are required. Knowing that  $p(500)$  has 22 digits, we set  $E = 70$ ,  $L = 100$ , and attenuation = 0.9 (since  $(0.9)^{500} \approx 10^{-23}$ ). The results are shown in Table 1. It took about one hour to do the calculations on a PC386, running at 20 MHz.

TABLE 1  
*Unrestricted partitions*

2000 Terms E=70 L=100		
96-digit arithmetic Attenuation=.9		
n	Value	Error
0	1	2.8 * 10 <sup>-38</sup>
50	204226	7.8 * 10 <sup>-36</sup>
100	190569292	8.9 * 10 <sup>-34</sup>
150	40853235313	5.8 * 10 <sup>-30</sup>
200	3972999029388	3.9 * 10 <sup>-26</sup>
250	230793554364681	2.5 * 10 <sup>-22</sup>
300	9253082936723602	1.6 * 10 <sup>-18</sup>
350	279363328483702152	1.0 * 10 <sup>-14</sup>
400	6727090051741041926	6.7 * 10 <sup>-11</sup>
450	134508188001572923840	4.3 * 10 <sup>-7</sup>
500	2300165032574323995027	2.7 * 10 <sup>-3</sup>

TABLE 2  
*Cosine-coefficients for unrestricted partitions*

40 Terms E=70 L=100 96-digit arithmetic Attenuation=1.0			
n	C <sub>n</sub>	n	C <sub>n</sub>
0	0.0155834820	20	-0.0146691001
1	0.0550938561	21	-0.0170261156
2	0.0541519010	22	-0.0190597232
3	0.0524163367	23	-0.0207842276
4	0.0499495263	24	-0.0222179896
5	0.0468339096	25	-0.0233824799
6	0.0431664802	26	-0.0243013537
7	0.0390529749	27	-0.0249995712
8	0.0346023496	28	-0.0255025890
9	0.0299219682	29	-0.0258356452
10	0.0251137590	30	-0.0260231517
11	0.0202714114	31	-0.0260882070
12	0.0154785584	32	-0.0260522300
13	0.0108077954	33	-0.0259347124
14	0.0063203534	34	-0.0257530816
15	0.0020662418	35	-0.0255226606
16	-0.0019152981	36	-0.0252567095
17	-0.0055951495	37	-0.0249665300
18	-0.0089535687	38	-0.0246616146
19	-0.0119793746	39	-0.0243498208

**TABLE 3**  
*Unrestricted partitions*

40 Terms E=70 L=100  
96-digit arithmetic Attenuation=1.0

n	Value	Error
0	1.00000	2.0 * 10 <sup>-38</sup>
1	1.00000	1.6 * 10 <sup>-37</sup>
2	2.00000	4.4 * 10 <sup>-36</sup>
3	3.00000	1.4 * 10 <sup>-34</sup>
4	5.00000	4.6 * 10 <sup>-33</sup>
5	7.00000	1.5 * 10 <sup>-31</sup>
6	11.00000	5.0 * 10 <sup>-30</sup>
7	15.00000	1.6 * 10 <sup>-28</sup>
8	22.00000	5.5 * 10 <sup>-27</sup>
9	30.00000	1.8 * 10 <sup>-25</sup>
10	42.00000	7.7 * 10 <sup>-24</sup>
11	56.00000	1.6 * 10 <sup>-21</sup>
12	77.00000	1.2 * 10 <sup>-18</sup>
13	101.00000	1.1 * 10 <sup>-15</sup>
14	135.00000	9.9 * 10 <sup>-13</sup>
15	176.00000	8.6 * 10 <sup>-10</sup>
16	231.00000	7.5 * 10 <sup>-7</sup>
17	297.00065	
18	385.57174	correct answer=385
19	980.00000	correct answer=490
20	422830.76599	correct answer=627

For small values of  $n$ , the calculations are accurate to about 40 significant digits. The accuracy systematically decreases to 24 digits as  $n$  increases to 500. Although the numbers change with the problem, the printout always looks like Table 1 when the answers are known to be integers. We repeated the same calculation with  $E = 105$  and  $L = 2000$ . The accuracy at  $n = 500$  increased to 43 significant digits. However, the computer time increased to nine hours.

For illustrative purposes, the same calculation was made but with only 40 cosine terms. The cosine-coefficients (truncated) are listed in Table 2, and the partitions are shown in Table 3. With 40 terms, high accuracy is assured only up to  $n = 10$ , but we extended the calculations to 20, with typical results. The cosine printout is also typical, but is not predictable (yet).

## 8. MORE COMPLICATED FUNCTION

There are more direct and faster methods for enumerating unrestricted partitions. However, this cosine-series method can be the most effective way to numerically evaluate other partition and combinatorial functions. For example, consider the number of ways  $n$  similar objects can be distributed into  $r$  different cells, including unrestricted partitions of the objects in each cell, a problem from graph theory. This is equivalent to the  $r$ -fold convolution of unrestricted partitions of  $n$  objects in one cell.

The generating function for this is simply equation (16) raised to the  $r$ th power,

$$(17) \quad F(x) = \left( \frac{1}{(1-x)(1-x^2)(1-x^3)\dots} \right)^r.$$

TABLE 4  
Unrestricted partitions into 100 cells

n	Number of Partitions	Zeros After Decimal Point
0	1.0000000000	61
1	100.0000000000	59
2	5150.0000000000	58
3	181800.0000000000	57
4	4941425.0000000000	55
5	110162620.0000000000	54
6	2095845950.0000000000	53
7	34962917400.0000000000	52
8	521585614000.0000000000	52
9	7062897346700.0000000000	51
10	87828985857380.0000000000	49
11	1012388182996200.0000000000	47
12	10900114289905025.0000000000	46
13	110320020147886800.0000000000	44
14	1055240483760370850.0000000000	43
15	9583361158333259760.0000000000	41
16	82961603673953999475.0000000000	39
17	686960321073922256900.0000000000	38
18	5457597221300936398800.0000000000	36
19	41711405649406950264000.0000000000	35
20	307419968899118268037235.0000000000	33
21	2189601546878826111425200.0000000000	32
22	15100553174874832471229300.0000000000	30
23	101012304110994536378731800.0000000000	28
24	656445643900578541883392625.0000000000	27
25	4150449074229675657676600124.0000000000	25
26	25564584111094035554187276350.0000000000	24
27	153588326674996231668155611600.0000000000	22
28	901032337861237426226358548275.0000000000	21
29	5166958969734300590443691341700.0000000000	19
30	28990702304337209570816847100720.0000000000	17
31	159293636023107930969242863847000.0000000000	16
32	857855565668207446850114468890875.0000000000	14
33	4531498857844668075446982931497600.0000000000	13
34	23496084642238776329323210516601900.0000000000	11
35	119665533605139086403027931040903800.0000000000	10
36	599016111555008151654846267275495275.0000000034	8
37	2948920386550115622669826037009108500.0000001264	6
38	14285215637926456305408040948501468200.0000046986	5
39	68130236400679543757786293964095809800.0001745764	3
40	320066066129413431354631633486042155940.0064863304	2

In the cosine method, this is accomplished by adding one instruction to the program, resulting in a negligible amount of additional computer time. The results are shown in Table 4 for  $r = 100$ .

#### 9. DISCUSSION

This method for numerically evaluating generating functions by expansion into a finite cosine-series produces remarkably accurate results. It is even more impressive when one realizes that determining the number of unrestricted partitions of 500 is equivalent to calculating the five-hundredth derivative of a function, which includes a 125,000th-degree polynomial, with 25-digit accuracy.

As a special case, this method gives a closed-form expression for the number of partitions, restricted or unrestricted. Although this is an approximation, in

the limit as one parameter  $E$  becomes infinite, the expression becomes exact. This may be a fertile area for developing new reasonably simple expressions for partition functions.

#### BIBLIOGRAPHY

1. J. Abate and H. Dubner, *A new method for generating power series expansions of functions*, SIAM J. Numer. Anal. **5** (1968), 102–112.
2. C. Caldwell, *The near repdigit primes*, J. Recreational Math. **22** (1990), 101–109.
3. H. Dubner and J. Abate, *Numerical inversion of Laplace transforms by relating them to the finite Fourier cosine transform*, J. Assoc. Comput. Mach. **15** (1968), 115–123.
4. G. H. Hardy and E. M. Wright, *An introduction to the theory of numbers*, 5th ed., Oxford Univ. Press, Oxford, 1979.

449 BEVERLY ROAD, RIDGEWOOD, NEW JERSEY 07450  
*E-mail address*: 70372.1170@compuserve.com