Proc. 2nd Caribbean Conf. Combinatorics Computing, Univ. of the West Indies, Barbados, 1977, pp. 191-208.

See pages 199

ON THE COMPUTER CALCULATION OF THE NUMBER OF NONSEPARABLE GRAPHS*

P. Butler and R.W. Robinson

Computing Centre and Mathematics Department,
University of Newcastle,
New South Wales, 2308,
AUSTRALIA.

ABSTRACT

Computation of the number b_p of unlabeled nonseparable graphs on p points has been carried out for $p \leq 18$ by one of the authors, based on equations derived by the other author. The number $b_{p,q}$ of those with q lines has similarly been obtained for $p \leq 11$. The numerical results are reported, and aspects of the computation are discussed.

The method of counting nonseparable graphs involves finding the sum of the cycle indices of the automorphism groups for all graphs, then the cycle index sum for the connected graphs and finally the cycle index sum for the nonseparable graphs. Extracting the cycle index sum of the connected graphs from the cycle index sum for all graphs is based on a principle which can be applied in a number of situations. Of three methods tested for implementing this principle, the clear winner in practice was not the one with the apparent computational advantages.

* The paper was presented by the second author.

Both authors acknowledge the support of the Australian Research Grants Committee.

1. INTRODUCTION

The evaluation by computer of the number of unlabeled nonseparable graphs is reported on in this paper. A graph is *nonseparable* if it is connected and has no point whose deletion results in an empty or disconnected graph. The nonseparable graphs other than the one with $p = 2$ points are just the 2-connected graphs. Nonseparable graphs have been called *blocks* by some graph theorists and *star graphs* by theoretical physicists. The graphs with which we deal are finite and undirected with no loops or multiple lines.

Calculation of the number b_p of unlabeled nonseparable graphs is difficult because it requires the simultaneous determination of the additional information contained in the sum of the cycle indices of the automorphism groups of the graphs being counted. Thus as we shall see b_p is obtained as the sum of rational numbers, one associated with each partition of p. The computation is recursive, requiring the numbers associated with the partitions less than p in order to determine those associated with the partitions of p. In addition the computation requires the analogous information for connected graphs, which in turn must be extracted from that for unrestricted graphs.

The formulas for counting unlabeled nonseparable graphs were developed in [8] and received a detailed exposition in the book by Harary and Palmer [3, Chapter 8]. The first computer implementation was carried out by L. Osterweil, who found b_p for $p \leq 9$ as reported in [8]. The values of b_p for $p \leq 18$ are presented in Section 3 after cycle index sum counting methods are introduced and the basic equations presented in Section 2. For comparison we also give the numbers c_p of unlabeled connected graphs and g_p of all unlabeled graphs on p points, for $p \leq 18$.

Including the number of lines as an enumeration parameter in counting nonseparable graphs is conceptually straightforward. However it expands considerably the storage required as well as the number of arithmetic operations. In Section 3 we give the numbers $b_{p,q}$ of unlabeled nonseparable graphs with p points and q lines for $p \leq 11$. For comparison we include the corresponding numbers $c_{p,q}$ and $g_{p,q}$ of connected and unrestricted unlabeled graphs.

There are several alternative strategies available for solving the basic equations which determine the cycle index sum for nonseparable graphs. The choice of strategy for computer implementation is discussed in the final section. In particular, three different ways of extracting the cycle index sum of the connected graphs from that of the unrestricted graphs were tested. The best method in practice was not the one which appeared to be most clever and efficient beforehand. The principle involved in this case can be expected to have wide application in unlabeled graph counting problems. It is in fact applied to an important part of the extraction of the cycle index sum of the nonseparable graphs from that of the connected graphs.

2. COUNTING WITH CYCLE INDEX SUMS

In this section we introduce cycle indices and cycle index sums for sets of graphs, then present the equations which are the basis for counting nonseparable graphs.

If g is any permutation on a finite set it can be expressed uniquely as a product of disjoint cyclic permutations. Let j(n;g) denote the number of these cycles which have length n. Then the *cycle type* of g is defined by

(1) $$Z(g) = \prod_n s_n^{j(n;g)},$$

where $s_1, s_2, s_3, \ldots$ are some set of independent commuting variables. If G is a finite permutation group then its *cycle index* Z(G) is the arithmetic mean of the cycle types of its elements. That is

(2) $$Z(G) = \frac{1}{|G|} \sum_{g \in G} Z(g),$$

where |G| denotes the order of G. Often the cycle index is written $Z(G; s_1, s_2, s_3, \ldots)$ to display the variables and to provide a convenient form for indicating substitutions for the variables.

As an example, let S_3 be the symmetric group on the object set {1,2,3}. The identity permutation (1)(2)(3) has cycle type s_1^3. The permutations (1)(23), (2)(13) and (3)(12) all have cycle type $s_1 s_2$. Then the cycles (123) and (132) have cycle type s_3. Thus we have

$$Z(S_3) = \tfrac{1}{6}s_1^3 + \tfrac{1}{2}s_1 s_2 + \tfrac{1}{3}s_3.$$

In general, let S_p denote the symmetric group of degree p and order p!.

It is well known that

$$Z(S_p) = \sum_{|(j)|=p} \prod_n s_n^{j_n} \Big/ \prod_n j_n!\, n^{j_n},$$

where the sum is over sequences

$$(j) = (j_1, j_2, \ldots, j_p)$$

of non-negative integers with weight

$$|(j)| = \sum_{n=1}^{p} n j_n$$

equal to p. Thus the sum of these cycle indices takes the elegant form

$$\sum_{p=0}^{\infty} Z(S_p) = \exp \sum_{1}^{\infty} s_n/n. \tag{3}$$

It is interesting to note that if s_n is the n^{th} power sum $\Sigma\alpha_i^n$ of variables $\alpha_1, \alpha_2, \alpha_3, \ldots$ then $Z(S_p)$ is the homogeneous product sum of order p in these variables. In this context the usual notation for $Z(S_p)$ would be h_p, and (3) expresses a familiar form of the relations between these symmetric functions [5, p.7].

Every graph has an automorphism group, which we consider to be a permutation group on the point set. For any set A of graphs we denote by Z(A) the sum of the cycle indices of the automorphism groups of the members of A. The case of the set H_3 of all four graphs on $p = 3$ points is illustrated in Figure 1, where each graph is shown with its automorphism group. The cycle index sum is

$$\begin{aligned} Z(H_3) &= 2(\tfrac{1}{6}s_1^3 + \tfrac{1}{2}s_1 s_2 + \tfrac{1}{3}s_3) + 2(\tfrac{1}{2}s_1^3 + \tfrac{1}{2}s_1 s_2) \\ &= \tfrac{4}{3}s_1^3 + 2s_1 s_2 + \tfrac{2}{3}s_3. \end{aligned} \tag{4}$$

$Z(S_3)$ $s_1Z(S_2)$ $s_1Z(S_2)$ $Z(S_3)$

FIGURE 1

The four graphs on $p = 3$ points.

Note that the number 4 of graphs on $p = 3$ points is obtained as the sum $4 = \frac{4}{3} + 2 + \frac{2}{3}$ of the coefficients of $Z(H_3)$. In general, for any set A of graphs the number with p points can always be found from Z(A) by summing the coefficients of the terms of weight p. This is because the coefficient sum in any cycle index must be 1.

Now let H_p denote the set of all graphs on p points. The cycle index sum $Z(H_p)$ can be readily computed in terms of the number c(j) of cycles of lines induced on the complete graph by a point permutation with cycle structure $(j) = (j_1, j_2, \ldots, j_p)$. The formula is

(5)
$$Z(H_p) = \sum_{|(j)|=p} 2^{c(j)} \prod_n s_n^{j_n} \Big/ \prod_n j_n! \, n^{j_n}.$$

This follows from a slight generalisation of Burnside's Lemma, in the spirit of Redfield's Decomposition Theorem [7, p.445], as reported in [8, Theorem 2]. One need only observe that $c(3,0,0) = 3$, $c(1,1,0) = 2$ and $c(0,0,1) = 1$ in order to evaluate formula (5) for $p = 3$. The answer is easily seen to agree with the sum given in (4). For arbitrary (j) the number of induced line-cycles is given [3, equation 4.1.9] by

(6)
$$c(j) = \sum_n n\left\{\binom{j_n}{2} + j_{2n+1} + j_{2n}\right\} + \sum_{m<n} (m,n) j_m j_n,$$

where (m,n) is the greatest common divisor of m and n.

Let H denote the set of all graphs, C the set of all connected graphs, and B the set of all nonseparable graphs. The cycle index sum Z(H) is determined by (5) and (6) as

$$Z(H) = \sum_{p=0}^{\infty} Z(H_p).$$

Of course in any actual computation there is an upper bound $\overline{p}$ on the number of points to be considered.

To express Z(C) in terms of Z(H) we need the notion of substituting Z(C) for cycle variables. This follows the rule

$$s_n[Z(C)] = Z(C; s_n, s_{2n}, s_{3n}, \ldots), \tag{7}$$

which also applies to the substitution of any cycle index sum. The relation can now be written

$$Z(H) = \exp \sum_1^{\infty} \frac{1}{n} s_n[Z(C)]. \tag{8}$$

In Section 4 we will compare three different ways of solving (8) for Z(C) in terms of Z(H). It can be seen that (8) is a generalisation of the usual relation [3, equation 4.2.3] between the ordinary generating functions $g(x) = \sum_0^{\infty} g_p x^p$ and $c(x) = \sum_1^{\infty} c_p x^p$, which follows at once from Pólya's Hauptsatz [6, p.163]. Indeed (8) follows from a direct lifting of Pólya's Hauptsatz to cycle index sums in place of ordinary generating functions [8, equation (9)].

As intermediaries in the computation of Z(B) the cycle index sums Z(C') and Z(B') of the rooted connected graphs C' and the rooted nonseparable graphs B' will be needed. A *rooted graph* is a graph in which one point is given the special status of *root point*. The root point must be left fixed by every automorphism of a rooted graph, and is omitted from consideration in the cycle index sum. It is easy to obtain Z(C') from Z(C) since

$$Z(C') = \frac{\partial Z(C)}{\partial s_1}. \tag{9}$$

This follows from a general result [8, Theorem 1] which was first observed by G.W. Ford, and remains true if C is replaced by B.

The relation determining Z(B') from Z(C') is

$$Z(C') = \exp \sum_1^{\infty} \frac{1}{n} s_n[Z(B')[s_1 Z(C')]]. \tag{10}$$

Structural considerations due to R.Z. Norman are used in the proof [8, Theorem 4]. Since (9) applies with B in place of C we can integrate to obtain

$$Z(B) = \int_0^{s_1} Z(B')ds_1 + Z(B;0,s_2,s_3,\ldots) . \tag{11}$$

The fixed-point-free portion of Z(B) is then determined by

$$Z(C;0,s_2,s_3,\ldots) = Z(B;0,s_2,s_3,\ldots)[s_1 Z(C')]. \tag{12}$$

Further consideration of the structure of a connected graph with respect to its maximal nonseparable subgraphs is required to justify this relation [8, Theorem 5].

Relations (5) through (12) serve to determine Z(B). The number b_p of unlabeled nonseparable graphs on p points is obtained from Z(B) by adding up the coefficients of all terms of weight p.

The numbers $b_{p,q}$ of unlabeled nonseparable graphs with p points and q lines are counted in a parallel manner using an expanded version of the cycle index sum. For any set A of graphs, let Z(y:A) denote the sum over the graphs in A of y^q times the cycle index of the automorphism group, where q is the number of lines. Then $b_{p,q}$ is obtained from Z(y:B) by summing the coefficients of the terms having the factor y^q and weight p in the point-cycle variables.

The expanded cycle index sum Z(y:B) is determined from Z(y:H) and Z(y:C) by modified forms of equations (6) and (8) through (12). The modification is simply to replace Z( ) by Z(y:) at every occurrence. To compute Z(y:H) initially one modifies equation (5) by replacing $2^{c(j)}$ with the polynomial

$$\prod_n (1+y^n)^{j_{2n}+[\frac{n-1}{2}]j_n+n\binom{j_n}{2}} \prod_{m<n} (1+y^{[m,n]})^{(m,n)j_m j_n}. \tag{13}$$

Here [m,n] is the least common multiple of m and n, and $[\frac{n-1}{2}]$ is the greatest integer less than or equal to $\frac{n-1}{2}$. This corresponds [8, equation (7)] to a factor of $(1+y^n)$ for each induced line-cycle of length n. The only other change needed is to generalise equation (7) to

$$s_n[Z(y:A)] = Z(y^n:A;s_n,s_{2n},s_{3n},\ldots). \tag{14}$$

3. NUMERICAL RESULTS

The numbers b_p of unlabeled nonseparable graphs on p points computed on the basis of equations (5) through (12) for $p \leq 18$ are presented in Table 1. Development of the programs was the work of the first author. Some of the experimentation concerning choice of methods for solving the equations is reported in the next section. The calculation was performed on a PDP 11/45 with a 56 kilobyte parity core memory and a twin RK05 fixed disk system for secondary storage. The programs were written in FORTRAN with the exception of some sections of the multiple precision integer arithmetic routines, which were written in the assembly language. They were compiled and run using the DOS/BATCH operating system. Because the disks provide essentially permanent storage of results, it was possible to compute Z(H), Z(C), and Z(B) in order, each through terms of weight $\overline{p} = 18$. Approximate times for these runs are 1/4 hour, 3 hours and 21 hours respectively, with an additional 1/4 hour for printing out all of the 1596 different terms needed for Z(B). Storage requirements were met with a single disk, having 2.4 megabytes total capacity including .5 megabytes taken up with the operating system.

Also provided in Table 1 for purposes of comparison are the numbers g_p and c_p of unlabeled unrestricted and connected graphs, respectively. C. King had computed g_p for $p \leq 25$ as reported in [3, Table A3]. Stein and Stein [9] had computed $c_{p,g}$ for $p \leq 18$, from which one can obtain c_p by summing q from 0 to $\binom{p}{2}$. Our own computations confirm the values of g_p and c_p reported in Table 1. It will be seen that our results for b_p are in agreement with those of L. Osterwiel for $p \leq 9$, as reported in [3, Table A3].

It is shown in [3, Section 9.4] that $b_p \sim c_p \sim g_p$. From Table 1 it appears that c_p/g_p approaches 1 much more rapidly than b_p/c_p.

The numbers $b_{p,q}$ of unlabeled nonseparable graphs with p points and q lines computed on the basis of equations (5) through (12) modified by (13) and (14) for $p \leq 11$ are presented in Tables 2, 3 and 4. Program development and implementation for the determination of Z(y:H), Z(y:C) and Z(y:B) through terms of point weight $\overline{p} = 11$ followed the same pattern as for Z(B). The additional line parameter increased storage and time demands so much that in spite of the reduction of $\overline{p}$ from 18 to 11, two disks were required for storage and the total running time rose from 27 hours to 60 hours.

N1155 – omit

N 649

- enter S, T card
- name card OK
- need new ref card

ref: TAMS 78 459 55. ST1.
CA2 2 199 77.

N479 – omit

New ref card needed for N649
For instructions see opposite page

have N 1155
enter N 649 = 1349
have N 479

p	b_p / c_p / g_p	p	b_p / c_p / g_p
	0		9 743 542
1	1	10	11 716 571
	1		12 005 168
	1		900 969 091
2	1	11	1 006 700 565
	2		1 018 997 864
	1		153 620 333 545
3	2	12	164 059 830 476
	4		165 091 172 592
	3		48 432 939 150 704
4	6	13	50 335 907 869 219
	11		50 502 031 367 952
	10		28 361 824 488 394 169
5	21	14	29 003 487 462 848 061
	34		29 054 155 657 235 488
	56		30 995 890 806 033 380 784
6	112	15	31 397 381 142 761 241 960
	156		31 426 485 969 804 308 768
	468		63 501 635 429 109 597 504 951
7	853	16	63 969 560 113 225 176 176 277
	1 044		64 001 015 704 527 557 894 928
	7 123		244 852 079 292 073 376 010 411 280
8	11 117	17	245 871 831 682 084 026 519 528 568
	12 346		245 935 864 153 532 932 683 719 776
	194 066		1 783 160 594 069 429 925 952 824 734 641
9	261 080	18	1 787 331 725 248 899 088 890 200 576 580
	274 668		1 787 577 725 145 611 700 547 878 190 848

TABLE 1

The numbers of unlabeled graphs, connected graphs and nonseparable graphs on $p \leq 18$ points.

Again, Tables 2, 3 and 4 are supplied with the numbers $c_{p,q}$ and $g_{p,q}$ of unlabeled connected and unrestricted graphs with p points and q lines for $p \leq 11$ for purposes of comparison. These numbers were calculated by computer for $p \leq 18$ by Stein and Stein [9], in what seems to have been the first major graphical enumeration project undertaken on a computer. Our own results serve to verify the accuracy of theirs for $p \leq 11$. In computing $g_{p,q}$, it is necessary to compute every term of $Z(y{:}H)$ in the process; however, these can be summed in running totals for each $p \leq \bar{p}$ and $q \leq \binom{p}{2}$. Since only the totals $g_{p,q}$ need to be kept in order for the $c_{p,q}$ to be determined, the requirements for storage space and consequent retrieval time are much less than for our procedure, which used all of $Z(y{:}H)$ to determine all of $Z(y{:}C)$ for $p \leq \bar{p}$. The latter was necessary for obtaining $Z(y{:}B)$ and indeed all terms of $Z(y{:}B)$ for $p < \bar{p}$ were needed in order to compute those terms of the highest order $p = \bar{p}$.

Omit

p \ q	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28
1	0																												
	1																												
	1																												
2	0	1																											
	0	1																											
	1	1																											
3	0	0	0	1																									
	0	0	1	1																									
	1	1	1	1																									
4	0	0	0	0	1	1	1																						
	0	0	0	2	2	1	1																						
	1	1	2	3	2	1	1																						
5	0	0	0	0	0	1	2	3	2	1	1																		
	0	0	0	0	3	5	5	4	2	1	1																		
	1	1	2	4	6	6	6	4	2	1	1																		
6	0	0	0	0	0	0	1	3	9	14	12	8	5	2	1	1													
	0	0	0	0	0	6	13	19	22	20	14	9	5	2	1	1													
	1	1	2	5	9	15	21	24	24	21	15	9	5	2	1	1													
7	0	0	0	0	0	0	0	1	4	20	50	82	94	81	59	38	20	10	5	2	1	1							
	0	0	0	0	0	0	11	33	67	107	132	138	126	95	64	40	21	10	5	2	1	1							
	1	1	2	5	10	21	41	65	97	131	148	148	131	97	65	41	21	10	5	2	1	1							
8	0	0	0	0	0	0	0	0	1	6	40	161	429	780	1076	1197	1114	885	622	386	215	112	55	24	11	5	2	1	1
	0	0	0	0	0	0	0	23	89	236	486	814	1169	1454	1579	1515	1290	970	658	400	220	114	56	24	11	5	2	1	1
	1	1	2	5	11	24	56	115	221	402	663	980	1312	1557	1646	1557	1312	980	663	402	221	115	56	24	11	5	2	1	1

TABLE 2

The numbers of (p,q)-graphs which are nonseparable, connected, and unrestricted for $p \leq 8$.

$b_{9,q}$	$c_{9,q}$	$g_{9,q}$	q	$b_{10,q}$	$c_{10,q}$	$g_{10,q}$
0	0	1	0	0	0	1
0	0	1	1	0	0	1
0	0	2	2	0	0	2
0	0	5	3	0	0	5
0	0	11	4	0	0	11
0	0	25	5	0	0	26
0	0	63	6	0	0	66
0	0	148	7	0	0	165
0	47	345	8	0	0	428
1	240	771	9	0	106	1 103
7	797	1 637	10	1	657	2 769
70	2 075	3 252	11	9	2 678	6 759
433	4 495	5 995	12	121	8 548	15 772
1 729	8 404	10 120	13	1 034	22 950	34 663
4 796	13 855	15 615	14	5 898	53 863	71 318
9 981	20 303	21 933	15	23 370	112 618	136 433
16 542	26 631	27 987	16	69 169	211 866	241 577
22 844	31 400	32 403	17	162 593	361 342	395 166
27 015	33 366	34 040	18	317 364	561 106	596 191
27 837	31 996	32 403	19	530 308	795 630	828 728
25 350	27 764	27 987	20	774 876	1 032 754	1 061 159
20 570	21 817	21 933	21	1 004 519	1 229 228	1 251 389
14 971	15 558	15 615	22	1 167 116	1 343 120	1 358 852
9 842	10 096	10 120	23	1 224 430	1 348 674	1 358 852
5 885	5 984	5 995	24	1 166 153	1 245 369	1 251 389
3 210	3 247	3 252	25	1 012 187	1 057 896	1 061 159
1 621	1 635	1 637	26	803 138	827 086	828 728
765	770	771	27	583 958	595 418	596 191
342	344	345	28	389 779	394 820	395 166
147	148	148	29	239 362	241 428	241 577
63	63	63	30	135 571	136 370	136 433
25	25	25	31	70 999	71 293	71 318
11	11	11	32	34 548	34 652	34 663
5	5	5	33	15 729	15 767	15 772
2	2	2	34	6 743	6 757	6 759
1	1	1	35	2 763	2 768	2 769
1	1	1	36	1 100	1 102	1 103
			37	427	428	428
			38	165	165	165
			39	66	66	66
			40	26	26	26
			41	11	11	11
			42	5	5	5
			43	2	2	2
			44	1	1	1
			45	1	1	1

TABLE 3

The number of (p,q)-graphs which are nonseparable, connected and unrestricted, for $p = 9$ and 10.

TABLE 4

The number of (p,q)-graphs which are nonseparable, connected, and unrestricted, for p = 11.

q	$b_{11,q}$	$c_{11,q}$	$g_{11,q}$
0	0	0	1
1	0	0	1
2	0	0	2
3	0	0	5
4	0	0	11
5	0	0	26
6	0	0	67
7	0	0	172
8	0	0	467
9	0	0	1 305
10	0	235	3 664
11	1	1 806	10 250
12	11	8 833	28 259
13	189	33 851	75 415
14	2 242	109 844	192 788
15	17 491	313 670	467 807
16	94 484	803 905	1 069 890
17	380 528	1 870 168	2 295 898
18	1 212 002	3 978 187	4 609 179
19	3 194 294	7 775 398	8 640 134
20	7 197 026	14 013 042	15 108 047
21	14 185 903	23 350 556	24 630 887
22	24 865 489	36 052 412	37 433 760
23	39 222 782	51 662 585	53 037 356
24	56 168 906	68 803 769	70 065 437
25	73 507 286	85 251 441	86 318 670
26	88 352 143	98 355 794	99 187 806
27	97 907 571	105 723 785	106 321 628
28	100 320 075	105 925 685	106 321 628
29	95 254 046	98 945 882	99 187 806
30	83 948 028	86 182 087	86 318 670
31	68 750 894	69 994 055	70 065 437
32	52 365 761	53 002 668	53 037 356
33	37 116 865	37 417 977	37 433 760
34	24 492 244	24 624 123	24 630 887
35	15 051 469	15 105 276	15 108 047
36	8 618 401	8 639 030	8 640 134
37	4 601 228	4 608 750	4 609 179
38	2 293 093	2 295 733	2 295 898
39	1 068 920	1 069 824	1 069 890
40	467 472	467 781	467 807
41	192 671	192 777	192 788
42	75 372	75 410	75 415
43	28 243	28 257	28 259
44	10 244	10 249	10 250
45	3 661	3 663	3 664
46	1 304	1 305	1 305
47	467	467	467
48	172	172	172
49	67	67	67
50	26	26	26
51	11	11	11
52	5	5	5
53	2	2	2
54	1	1	1
55	1	1	1

NUMERICAL METHODS

The polynomials Z(H), Z(C), Z(B) of Section 2 are specified by an array of coefficients and an array of partitions. In calculating Table 1 an array of the 1596 partitions of the numbers ≤ 18 is required. All the polynomials in the calculation are referenced through this partition array so it was held in core throughout. The rational coefficients, however, were held as direct access disk files since the fractions had quite large numerators and several separate arrays were required.

The partitions were precalculated in lexicographic order and were stored so that if the partition of p being considered was $p = \Sigma p_i n_i$, where $p_i, n_i > 0$, then p_i and n_i were packed into one word. An array of pointers marked the beginning and end of each partition. Another array of pointers marked the beginning and end of the terms for each value of p. For the enumeration by lines shown in Tables 2, 3 and 4 the only difference in the storage arrangments was that arrays of fractions were held instead of single fractions.

Three methods were considered for the calculation of Z(C) from Z(H) by solving equation (8). These were tested for $p \leq 15$ which required 684 terms. The best of these methods was chosen for the enumeration up to $\overline{p} = 18$, which required 1596 terms. Immediately from (8) we have

$$\sum_{k=1}^{\infty} \frac{1}{k} s_k[Z(C)] = \log Z(H).$$

This can be solved using a method suggested by Cadogan [1] based on Möbius inversion, giving

$$Z(C) = \sum_{k=1}^{\infty} \frac{\mu(k)}{k} s_k[\log Z(H)]. \tag{15}$$

This equation is the point of departure for the first and third of our methods. In the first, or "dumb", method log Z(H) was replaced by

$$\sum_{i=1}^{\infty} \frac{(-1)^{i+1}}{i} \left(Z(H) - 1\right)^i.$$

The successive powers were evaluated as economically as possible and then summed. The time taken for this method was $5\frac{1}{2}$ hours up to $\overline{p} = 15$.

The second method was described in [8, p.339] and seemed attractive enough to be dubbed the "clever" method initially. Let $H^{(i)}$ be the set of graphs with all components having order at least i and let C_j be the set of connected graphs on j points. Then $Z(H^{(1)}) = Z(H)$, and $Z(C)$ can be read off by selecting the terms of weight 1. Next, $Z(H^{(2)})$ is calculated from the relation

$$Z(H^{(2)}) = Z(H^{(1)}) \exp - \sum_{1}^{\infty} \frac{1}{k} s_k[Z(C_1)].$$

Now we can read off $Z(C_2) + Z(C_3)$, denoted for brevity by $Z(C_{2-3})$, since the smallest nonempty disconnected graph with components all of order at least 2 must itself have order at least 4. In similar fashion

$$Z(H^{(4)}) = Z(H^{(2)}) \exp - \sum_{1}^{\infty} \frac{1}{k} s_k[Z(C_{2-3})]$$

from which one can read off $Z(C_{4-7})$. Finally

$$Z(H^{(8)}) = Z(H^{(4)}) \exp - \sum_{1}^{\infty} \frac{1}{k} s_k[Z(C_{4-7})]$$

giving $Z(C_{8-15})$. In this process the exponentiation was computed by the recursive scheme described for the third method below. Extensive use was made of an additional array of flags to indicate whether a number in a disk held array was zero or not; the point was to save disk access time where possible. The time taken for calculation up to $\bar{p} = 15$ was $3\frac{1}{2}$ hours.

The third or "best" method differed from the first in that $\log Z(H)$ was calculated recursively. The recurrence is obtained from the following general relation by replacing v_i with $Z(C_i)$ and E_i with the terms of weight i in $Z(H)$. If

$$\sum_{i=0}^{\infty} E_i x^i = \exp \sum_{i=1}^{\infty} v_i x^i,$$

then $E_0 = 1$, and

$$E_n = v_n + \frac{1}{n} \sum_{k=1}^{n-1} E_k v_{n-k} (n-k)$$

for $n > 0$. In the theory of symmetric functions this is equivalent to the well-known recurrence relation between the s_i and the h_j [4, p.83] which can be derived from (3) by differentiation. The idea has been used in counting labeled connected graphs by Gilbert [2], and in counting unlabeled connected graphs by Stein and Stein [9, p.9] and Cadogan [1, p.195].

Clearly the recursion will serve to calculate either exponentials or logarithms. The time taken to calculate Z(C) up to $\overline{p} = 15$ was $1\frac{1}{4}$ hours, so that the third method was by far the quickest. It was used to extend the results up to $\overline{p} = 18$, and took some 3 hours for the calculation.

The calculation of Z(B) from Z(C) falls naturally into two parts as described in Section 2; the calculation of Z(B') using relation (10) and the calculation of $Z(B;0,s_2,s_3,\ldots)$ using (12). It is relatively trivial to find Z(C') by differentiating Z(C) as indicated in (9). Then $Z(B')[s_1Z(C')]$ is obtained from Z(C') in exactly the same way that Z(C) was obtained from Z(H), using of course the third method for the computation. From this Z(B') is extracted recursively. One keeps a running total of the contributions by the terms of Z(B') obtained from composing over $s_1Z(C')$ up to, say, weight i. Subtraction of this total from $Z(B')[s_1Z(C')]$ leaves just the terms of weight i+1 in Z(B'), which are used to update the running total so that the process can be continued.

It is worth pointing out here that the composition $Z(B')[s_1Z(C')]$ involves the substitution, for each factor s_i in each term of Z(B'), of the polynomial $s_1Z(C')$ with all the subscripts multiplied by i, that is, *inflated* by i. The consequences of this are that for $p \leq 18$, up to 18 inflated forms of $s_1Z(C')$ will be required, raised to powers up to 18. Varying combinations of these polynomials are then multiplied together (up to 5 factors in all for $p \leq 18$) to complete the substitution. The very great deal of computation here was economised by precomputing all the inflated powers, saving only nonzero coefficients, and setting up an array of pointers. The total disk space required for $p \leq 18$ was only 4 times that required to store just $s_1Z(C')$. The multiplication together of the polynomials was performed by nested loops. Extracting $Z(B;0,s_2,s_3,\ldots)$ from $Z(C;0,s_2,s_3,\ldots)$ using equation (12) follows exactly the same recursive method, but is much less demanding of time since the number of nonzero terms is relatively small. The final step of integrating Z(B') and adding $Z(B;0,s_2,s_3,\ldots)$ to obtain Z(B) is quite straightforward.

Times taken for the various calculations up to $\overline{p} = 18$ were:

(a) 3 hours for determining $Z(B')[s_1Z(C')]$,

(b) 7 hours for precomputation of the inflated powers of $s_1 Z(C')$,

(c) 21 hours for extracting $Z(B')$,

(d) $1\frac{1}{4}$ hours for extracting $Z(B;0,s_2,s_3,\ldots)$,

(e) $\frac{1}{2}$ hour for the integration of $Z(B')$ and its collation with $Z(B;0,s_2,s_3,\ldots)$.

Calculation of $Z(y:B)$ followed the same pattern as used for $Z(B)$, but did require considerable modification of the programs. Extension was necessary to deal with the fact that the coefficients of the partition arrays are now polynomials in y, rather than simple fractions. This means for instance that computing up to $\overline{p} = 11$ requires up to 55 fractional coefficients for each partition. The longest single step in the computation was the analogue of step (c) above, which took 50 hours up to $\overline{p} = 11$. The whole of the free space on two 2.4 megabyte disks was required for working storage.

REFERENCES

[1] C.C. Cadogan, The möbius function and connected graphs, *J. Combinatorial Theory* 11B (1971) 193-200.

[2] E.N. Gilbert, Enumeration of labelled graphs, *Canad. J. Math.* 8 (1956) 405-411.

[3] F. Harary and E.M. Palmer, *Graphical Enumeration.* Academic Press, New York, 1973.

[4] D.E. Littlewood, *The Theory of Group Characters.* 2nd ed., Oxford, 1950.

[5] P.A. MacMahon, *Combinatory Analysis.* Cambridge; reprinted by Chelsea, New York, 1960.

[6] G. Pólya, Kombinatorische Anzahlbestimmungen fur Gruppen, Graphen und chemische Verbindungen, *Acta Math.* 68 (1937) 145-254.

[7] J.H. Redfield, The theory of group-reduced distributions, *Amer. J. Math.* 49 (1927) 433-455.

[8] R.W. Robinson, Enumeration of non-separable graphs, *J. Combinatorial Theory* 9 (1970) 327-356.

[9] M.L. Stein and P.R. Stein, Enumeration of linear graphs and connected linear graphs up to p = 18 points. Los Alamos Scientific Laboratory of the University of California, Los Alamos, N.M., 1967, report LA-3775.