Acta Bot. Neerl. 22(6), December 1973, p. 645–654.

AN ALGOL-60 PROGRAM FOR COMPUTATION AND GRAPHICAL REPRESENTATION OF POLLEN ANALYTICAL DATA

A. VOORRIPS

Instituut voor Prae- en Protohistorie, Universiteit van Amsterdam

SUMMARY

This paper describes an ALGOL-60 computer program for computation and graphical representation of pollen analytical data. The program offers the possibility of computing and drawing relative frequencies, values per unit of sediment and values per unit of sediment per unit of time. An input description is included.

1. INTRODUCTION

Pollen analytical data are usually presented by means of pollen diagrams in which the values for the various pollen types, computed in one way or another, are plotted on a linear scale. These pollen diagrams generally serve two purposes. Firstly they may be used for the correlation of lithostratigraphic or biostratigraphic features of an area. In order to keep the pollen diagrams comparable the computations are to be performed on a similar basis. Secondly, pollen diagrams may be used for palaeo-ecological investigations. Pollen types may be plotted in groups showing the ecological affinities of the taxa involved (Janssen 1967). In this way the composition and succession of the vegetation may be visualized more accurately. Moreover, the increasingly refined distinctions that are made between the various pollen types tend to result in rather extensive diagrams, so grouping of the pollen types will promote the readability. In both cases it may be useful to calculate not only the relative frequencies of both the different pollen types and groups of pollen types, but also the pollen densities, either as the values per unit of sediment (Aario 1943; Erdtman 1943; Davis 1965) or as the values per unit of sediment per unit of time (Davis 1967). Therefore there is a need for pollen diagrams composed and computed in different ways. Computing and plotting of these diagrams is a time-consuming process. Moreover, compiling a pollen diagram for publication is a very accurate job but not an inspiring one.

For these reasons a computer program has been written that can perform a number of computations and presents the results in tables and graphs. The program language is ALGOL-60. In order to keep the speed of execution high and the cost low, the output is on the lineprinter. The program was originally written for the (Philips) Electrologica X8. The version presented here is an adaptation for IBM 360.

2. POSSIBILITIES IN COMPUTING AND PRINTING

The program offers the following possibilities which can be combined in different ways at each run.

a. Tables and/or graphs of the raw values.

b. Tables and/or graphs of the relative frequencies. As a basis for the relative frequencies any pollen sum can be selected but it is fixed for each run. For the pollen types not included in the sum two ways of computation are possible. If N is the number of pollen grains of such a type and S the pollen sum, then the relative frequency R_N can be computed either as $R_N = 100 \times N/S$, the normal way, or as $R_N = 100 \times N/(S+N)$, individual inclusion in the pollen sum (TROELS-SMITH 1955). The latter means that the values for R_N never exceed 100%.

c. Tables and/or graphs of the data, corrected for the concentration of pollen grains per unit of sediment. For this purpose the method of adding a standard solution of exotic pollen grains – or any other kind of object of similar

	Column 1	Column 2	Column 3	Columns 4-9
				type or group indices type or group names
2 a	spectrum index			raw values of types or groups
2 b	spectrum index	pollensum (SUM)		raw values of types or groups relative frequen-cies of types or groups
2 c	spectrum index	total number per unit of volume or weight (C SUM)	number of counted exotics (EXO)	concentration values of types or groups
2 d	spectrum index	total number per unit of volume per unit of time (I SUM)	AR/Ts x 100 (CA x 100)	influx values of types or groups

Fig. 1. Scheme of the possibilities for printing tables.

size and specific gravity – to a standard volume or weight of the sediment has been selected (BENNINGHOF 1962; HARTMAN 1968). The numbers of fossil and exotic grains that are to be counted to obtain reasonable confidence intervals can be found by using the equations by Mosimann or the nomograms by Maher (MOSIMANN 1965; MAHER 1972). The concentration per unit of sediment, C, follows from C = F × A/E × V, in which F is the number of counted fossil grains, A is the number of added exotic grains, E is the number of counted exotic pollen grains, and V is the volume or weight of the sample. The unit of measure in which C is expressed is in the same scale as that of V.

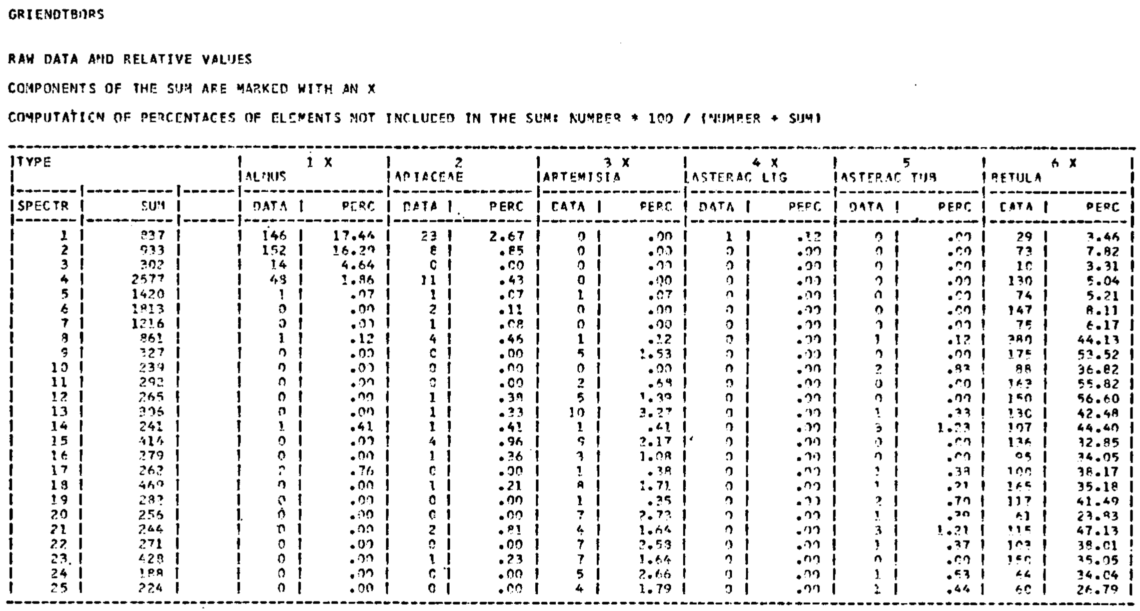
d. Tables and/or graphs of the influx values. The influx value can be defined as the number of pollen grains per unit of sediment per unit of time. This means that the accumulation rate of the sediment must be known. Time can be checked by ^{14}C assays, dendrochronological data or by archaeological data. The influx value, Iv, follows from Iv = C × Ar/Ts, in which C is the concentration value as calculated under 2.c, Ar is the accumulation rate of the sediment, its value depending on the actual form of the time curve (DAVIS 1967) and Ts is the vertical extension of the sample, expressed in a unit of measure that is in the same scale as that of C. In calculating the influx value, V – see 2.c – has to be the volume of the sample.

3. DESCRIPTION OF THE TABLES

All the tables are similarly composed. An example is shown in *fig. 2.* A scheme of the various possibilities is presented in *fig. 1.* The maximum number of printed characters of the names of pollen types or -groups at the head of the columns is 15. In case 2 b the type index is followed by an 'X' if the type is included in the pollen sum.

4. DESCRIPTION OF THE GRAPHS

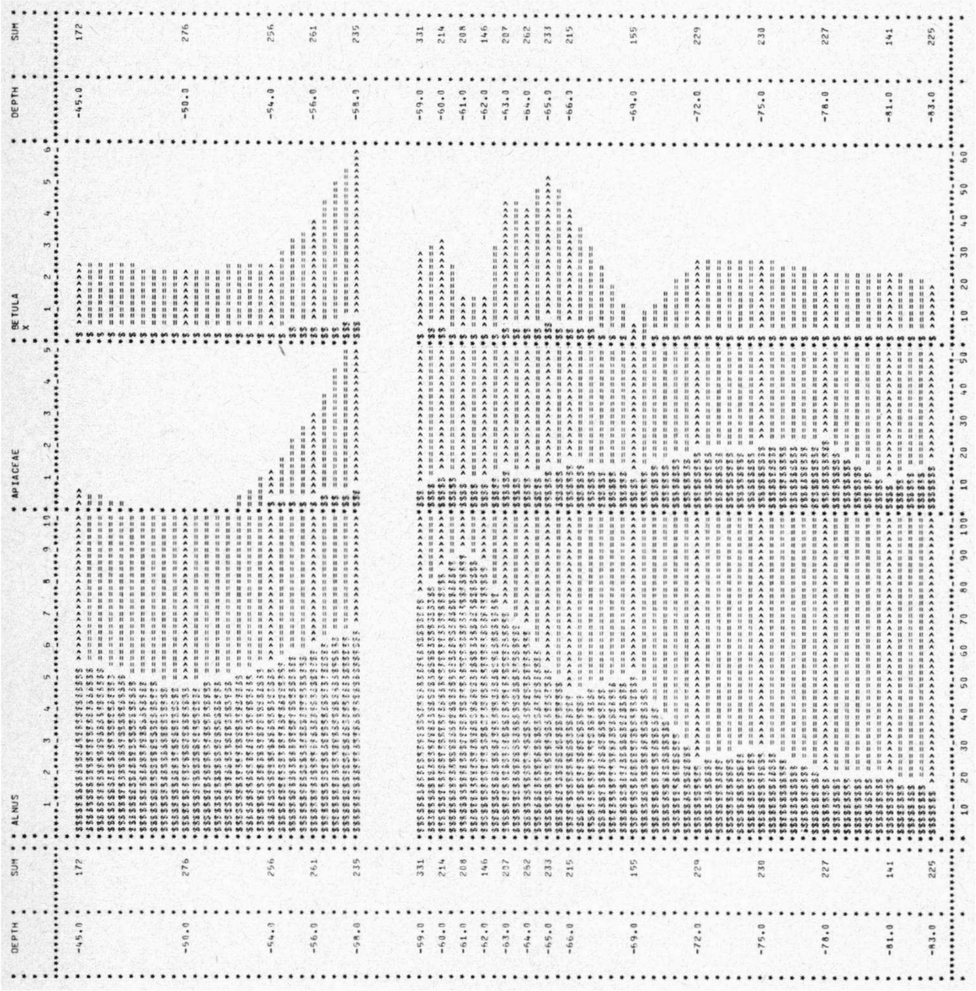
The graphs, an example is shown in *fig. 3,* have a maximum width of 50 characters. Variations in scale, both vertically and horizontally, are possible. Each graph shows two contours. The grey one corresponds to the scale at the top of the graph, the black one to that at the bottom. The black lines in the grey contour show the levels of the samples. The scale at the bottom is that of the top, multiplied by an integer positive number. This number has the name 'scale reduction' in the input description. In practice a scale reduction between 5 and 20 will do best. If the printing of a reduced value – the black contour – would need more than 50 characters, only 44 characters are printed. The value is printed as a number in the remaining six positions. However, when more than half of all the reduced values would need more than 50 characters, a procedure is started in which both the normal and the reduced scale are reduced 5, 10, 20, 40,..... times, till the point is reached when less than half of the reduced values need more more than 50 characters. It is also possible to vary the number of lines between the spectra. This number between two spectra p and q, LN_{pq},



```
GRIENDTBORS

RAW DATA AND RELATIVE VALUES

COMPONENTS OF THE SUM ARE MARKED WITH AN X

COMPUTATICN OF PERCENTACES OF ELEMENTS NOT INCLUCED IN THE SUM: NUMBER * 100 / (NUMBER + SUM)

------------------------------------------------------------------------------------------------------------------------------------
ITYPE                    I       1 X       I       2         I       3 X       I       4 X       I       5         I       6 X       I
I                        IALNUS            IAPIACEAE         IARTEMISIA        IASTERAC LIG      IASTERAC TUB      IBETULA           I
I-------I---------I------I-----------------I-----------------I-----------------I-----------------I-----------------I-----------------I
ISPECTR I     SUM I      I DATA I     PERC I DATA I     PERC I DATA I     PERC I DATA I     PERC I DATA I     PERC I DATA I     PERC I
I-------I---------I------I-----------------I-----------------I-----------------I-----------------I-----------------I-----------------I
I     1 I     837 I      I  146 I    17.44 I   23 I     2.67 I    0 I      .00 I    1 I      .12 I    0 I      .00 I   29 I     3.46 I
I     2 I     933 I      I  152 I    16.29 I    8 I      .85 I    0 I      .00 I    0 I      .00 I    0 I      .00 I   73 I     7.82 I
I     3 I     302 I      I   14 I     4.64 I    0 I      .00 I    0 I      .00 I    0 I      .00 I    0 I      .00 I   10 I     3.31 I
I     4 I    2577 I      I   48 I     1.86 I   11 I      .43 I    0 I      .00 I    0 I      .00 I    0 I      .00 I  130 I     5.04 I
I     5 I    1420 I      I    1 I      .07 I    1 I      .07 I    1 I      .07 I    0 I      .00 I    0 I      .00 I   74 I     5.21 I
I     6 I    1813 I      I    0 I      .00 I    2 I      .11 I    0 I      .00 I    0 I      .00 I    0 I      .00 I  147 I     8.11 I
I     7 I    1216 I      I    0 I      .00 I    1 I      .08 I    0 I      .00 I    0 I      .00 I    0 I      .00 I   75 I     6.17 I
I     8 I     861 I      I    1 I      .12 I    4 I      .46 I    1 I      .12 I    0 I      .00 I    1 I      .12 I  380 I    44.13 I
I     9 I     327 I      I    0 I      .00 I    0 I      .00 I    5 I     1.53 I    0 I      .00 I    0 I      .00 I  175 I    53.52 I
I    10 I     239 I      I    0 I      .00 I    0 I      .00 I    0 I      .00 I    0 I      .00 I    2 I      .83 I   88 I    36.82 I
I    11 I     292 I      I    0 I      .00 I    0 I      .00 I    2 I      .68 I    0 I      .00 I    0 I      .00 I  163 I    55.82 I
I    12 I     265 I      I    0 I      .00 I    1 I      .38 I    5 I     1.89 I    0 I      .00 I    0 I      .00 I  150 I    56.60 I
I    13 I     306 I      I    0 I      .00 I    1 I      .33 I   10 I     3.27 I    0 I      .00 I    1 I      .33 I  130 I    42.48 I
I    14 I     241 I      I    1 I      .41 I    1 I      .41 I    1 I      .41 I    0 I      .00 I    3 I     1.23 I  107 I    44.40 I
I    15 I     414 I      I    0 I      .00 I    4 I      .96 I    9 I     2.17 I    0 I      .00 I    0 I      .00 I  136 I    32.85 I
I    16 I     279 I      I    0 I      .00 I    1 I      .36 I    3 I     1.08 I    0 I      .00 I    0 I      .00 I   95 I    34.05 I
I    17 I     262 I      I    2 I      .76 I    0 I      .00 I    1 I      .38 I    0 I      .00 I    1 I      .38 I  100 I    38.17 I
I    18 I     469 I      I    0 I      .00 I    1 I      .21 I    8 I     1.71 I    0 I      .00 I    1 I      .21 I  165 I    35.18 I
I    19 I     282 I      I    0 I      .00 I    0 I      .00 I    1 I      .35 I    0 I      .00 I    2 I      .70 I  117 I    41.49 I
I    20 I     256 I      I    0 I      .00 I    0 I      .00 I    7 I     2.73 I    0 I      .00 I    1 I      .39 I   61 I    23.83 I
I    21 I     244 I      I    0 I      .00 I    2 I      .81 I    4 I     1.64 I    0 I      .00 I    3 I     1.21 I  115 I    47.13 I
I    22 I     271 I      I    0 I      .00 I    0 I      .00 I    7 I     2.58 I    0 I      .00 I    1 I      .37 I  103 I    38.01 I
I    23 I     428 I      I    0 I      .00 I    1 I      .23 I    7 I     1.64 I    0 I      .00 I    0 I      .00 I  150 I    35.05 I
I    24 I     188 I      I    0 I      .00 I    0 I      .00 I    5 I     2.66 I    0 I      .00 I    1 I      .53 I   64 I    34.04 I
I    25 I     224 I      I    0 I      .00 I    0 I      .00 I    4 I     1.79 I    0 I      .00 I    1 I      .44 I   60 I    26.79 I
------------------------------------------------------------------------------------------------------------------------------------
```

Fig. 2. Example of a table according to 2b.

Fig. 3.
Example of a diagram according to 2b.



follows from $LN_{pq} = D_{pq}/MIN$ in which D_{pq} is the distance between the samples, and MIN is a real positive number in the scale of the unit of measure of D_{pq}. If $D_{pq} \leqslant 1.5 \times MIN$, two spectra will be printed on consecutive lines. If $D_{pq} > 1.5 \times MIN$, LN_{pq} lines of characters will be printed between the spectra, the numbers of characters being interpolated between the spectra values, in order to produce a smooth graph. When this interpolation is to be suppressed, however, for instance in the case of discontinuity in the sedimentation, the adding of a special code to the input – see nr. 55 of the input description – will result in the printing of five blank lines between the corresponding spectra, as is shown in *fig. 3* in the upper half of the diagram.

Two columns are printed together with each kind of graph, comprising the distances to some constant level of each of the samples and the pollen sum in case 2b, the total numbers of pollen grains per unit of sediment in case 2c, the total numbers of pollen grains per unit of sediment per unit of time in case 2d.

5. INPUT DESCRIPTION

The input description is in the form of a key. The abbreviations between brackets point to the text of the program. The numbers between brackets point to the first place in the input description where an abbreviation has been used. If not stated otherwise the numbers to be punched are of integer type.

1. How many data sets are to be treated sequentially? Punch that number (njobs)
   2. Punch the name of the data set to be computed and printed between apostrophes (title)
   3. Punch the number of pollen types (ntypes)
   4. Punch the number of pollen spectra (nspectr)
   5. Punch the number of types included in the pollen sum (ninsum)
   6. Punch the number of groups of pollen types (ngroup)
   – mark that at 5 and 6 you may punch a 0 –
   7. ninsum (5) = 0? then proceed to 9
   8. Punch a 1 when at the calculation the types excluded from the pollen sum are to be included individually, if not then punch a 0 (locin) – see also 2.b –
   9. If concentration values are to be calculated then punch a 1 else a 0 (corconc)
   10. If influx values are to be calculated then punch a 1 else a 0 (corinf)
   11. If you want for the types the tables of the raw values and –in the case ninsum (5) ≠ 0 – of the relative frequencies then punch a 1 else a 0 (tabtraw)
   12. If you want graphs of the raw values of the types then punch a 1 else a 0 (grtraw)
   13. ninsum (5) = 0? then proceed to 15
       14. If you want graphs of the relative frequencies of the types then punch a 1 else a 0 (grtperc)
   15. grtraw (12) = 0? then proceed to 17
       16. Punch the scale reduction for the graphs of the raw values of the types (redtraw)
   17. You punched nothing or a 0 for grtperc (14)? then proceed to 19
       18. Punch the scale reduction for the graphs of the relative frequencies of the types (redtperc)
   19. corconc (9) = 0? then proceed to 24
       20. If you want tables of the concentration values of the types then punch a 1 else a 0 (tabtconc)
       21. If you want graphs of the concentration values of the types then punch a 1 else a 0 (grtconc)

```
    22. grtconc (21) = 0? then proceed to 24
        23. Punch the scale reduction for the graphs of the concentration values of the types (redtconc)
24. corinf = 0? then proceed to 29
    25. If you want tables of the influx values of the types then punch a 1 else a 0 (tabtinf)
    26. If you want graphs of the influx values of the types then punch a 1 else a 0 (grtinf)
    27. grtinf = 0? then proceed to 29
        28. Punch the scale reduction for the graphs of the influx values of the types (redtinf)
29. ngroup (6) = 0? then proceed to 48
    30. If you want tables of the raw data of the groups and – in the case ninsum (5) $\neq$ 0 – of the relative frequencies then punch a 1 else a 0 (tabgraw)
    31. If you want graphs of the raw data of the groups then punch a 1 else a 0 (grgraw)
    32. ninsum (5) = 0? then proceed to 34
        33. If you want graphs of the relative frequencies of the groups then punch a 1 else a 0 (grgperc)
    34. grgraw (31) = 0? then proceed to 36
        35. Punch the scale reduction for the graphs of the raw data of the groups (redgraw)
    36. You punched nothing or a 0 for grgperc (33)? then proceed to 38
        37. Punch the scale reduction for the graphs of the relative frequencies of the groups (redgperc)
    38. corconc (9) = 0? then proceed to 43
        39. If you want tables of the concentration values of the groups then punch a 1 else a 0 (tabgconc)
        40. If you want graphs of the concentration values of the groups then punch a 1 else a 0 (grgconc)
        41. grgconc (40) = 0? then proceed to 43
            42. Punch the scale reduction for the graphs of the concentration values of the groups (redgconc)
    43. corinf (10) = 0? then proceed to 48
        44. If you want tables of the influx values of the groups then punch a 1 else a 0 (tabginf)
        45. If you want graphs of the influx values of the groups then punch a 1 else a 0 (grginf)
        46. grginf (45) = 0? then proceed to 48
            47. Punch the scale reduction for the graphs of the influx values of the groups (redginf)
48. ninsum (5) = 0? then proceed to 50
    49. Punch the indices of the types included in the pollen sum (insum)
50. ngroup (6) = 0? then proceed to 52
    51. For each group punch the number of types in that group, followed by the in dices of these types (ingroup)
52. If you do not want any graphs of the types then proceed to 54
    53. For all the types punch a 1 if you want its graph, else a 0 (skip)
54. If you don't want any graph at all then proceed to 57
    55. Punch the distance to some constant level of each of the samples, beginning with the uppermost. Numbers of real type can be used. If there are levels between which you want the graphs to be discontinued, then punch +9999 between these levels (depth)
    56. Punch the maximum distance between the samples at which spectra still should be printed on consecutive lines. This distance must be > 0. A number of real type can be used (min)
57. corconc (9) = 0? then proceed to 61
    58. For each sample, beginning with the uppermost, punch the number of exotic pollen grains you added (stand)
```

59. For each sample, beginning with the uppermost, punch the volume or weight. Numbers of real type can be used (vol)
60. For each sample, beginning with the uppermost, punch the number of exotic pollen grains you counted (exo)
61. corinf (10) = 0? then proceed to 63
62. For each sample, beginning with the uppermost, punch the value of Ar/Ts – see 2d –. Numbers of real type can be used (cort)
63. Punch .001 – this has the function of a check on the foregoing part –
64. ngroup (6) = 0? then proceed to 66
65. Punch the names of the groups, each between apostrophes (grname)
66. You don't want any tables or graphs of the types? then proceed to 68
67. Punch the names of the types, each between apostrophes (name)
68. Punch ++ – this is a check on the namelist –
69. For each spectrum, beginning with the uppermost, punch the raw values of the types. Punch –1 at the end of each spectrum –this is a check on the number of types per spectrum –
70. If you have another data set then start again at 2.

6. DESCRIPTION OF INPUT/OUTPUT PROCEDURES DECLARED WITH 'CODE'[1]

1. 'real' 'procedure' READ; 'code';
The value of READ is the value of the next following number at the input medium, while the end of this number is to be shown by a number delimiter that can be any symbol except the characters for · or ' or a digit or only one space character or a sign character.
2. 'procedure' LINE (dn, n); 'value' dn, n; 'integer' dn, n; 'code'; LINE gives n transfers to a new line.
3. 'procedure' BLANK (dn, n); 'value' dn, n; 'integer' dn, n; 'code'; BLANK gives n blank spaces, a transfer to a new line included, if needed. On input n positions are skipped.
4. 'procedure' TEXT (dn, s); 'value' dn; 'integer' dn; 'string' s; 'code';
TEXT outputs the string s, starting on the first free position, if necessary continuing on new lines. No blanks are given behind the last character of the string.
5. 'procedure' FIX (dn, n, m, x); 'value' dn, n, m, x; 'integer' dn, n, m; 'real' x; 'code';
FIX outputs x in fixed point notation with sign (immediatly preceding the first non-zero digit or the decimal point), n digits before and m digits behind the decimal point, which is suppressed when m = 0. Leading zeros are replaced by blanks except when m = 0 and x = 0, in which case a single zero is produced, right aligned in the field, without a sign. Behind the last digit a number of blanks is added, the default value of this number being 2 in most implementations.
6. 'procedure' AFIX (dn, n, m, x); 'value' dn, n, m, x; 'integer' dn, n, m; 'real' x; 'code';
AFIX gives the same results as FIX with the only difference that instead of a sign a blank is produced.
7. 'procedure' DEF (dn, p, q); 'value' dn, p, q; 'integer' dn, p, q; 'code'; DEF opens the data-set; it sets the line-length at p and the page-length at q, if q ≠ 0. Setting q at 0 – as has been done in this program – requires provisions in job-control statements (IBM System/360 OS ALGOL, 1970).

7. REQUIREMENTS

The required storage for the program is about 90K bytes, using an IBM 360–65.

Required storage for the data-set (RSD) follows from:

[1] According to VAN WINGEN (1972)

RSD = (52 + nspectr) × ntypes + 4× nspectr + (if ninsum > 0 then 2 × ntypes × nspectr + ninsum else 0) + (if ngroup > 0 then (3× nspectr + ntypes + 50) × ngroup else 0) + (if corconc then 4 × nspectr else 0) + (if corinf then nspectr else 0) + ca. 100,

in which

the used numbers stand for words of minimally 20 bits.

8. AVAILABILITY OF THE PROGRAM

The program is available in versions for IBM 360 and for EL X8, while a version for CDC 6000 and 7000 series being in preparation. It can be obtained from the author as a paper tape (EL X8 version), as a card deck, or on magnetic tape that should be submitted (IBM 360 version, the described I/O procedures will be included).

ACKNOWLEDGEMENTS

The author is indebted to Dr. C. R. Janssen (Laboratory of Palaeobotany and Palynology of the State University of Utrecht), Mr. C. van de Wijgaart (Technical Centre, Department of Social Sciences of the University of Amsterdam), and to Mr. J. W. van Wingen (Dutch Institute for Preventive Medicine, Leiden) for their valuable advices. Computer time was placed at disposal by the Laboratory of Palaeobotany and Palynology of the State University of Utrecht, by the Technical Centre, Department of Social Sciences of the University of Amsterdam and by the Institute for Prae- and Protohistory of the University of Amsterdam.

REFERENCES

AARIO, L. (1943): Über die Wald- und Klimaentwicklung an der lappländischen Eismeerküste in Petsamo, mit einem Beitrag zur nord- und mitteleuropäischen Klimageschichte. *Ann. Bot. Soc. Zool. Bot. Fennicae Vanamo* **19**: 1–55.

BACKUS, J. W. e.a. (1963): Revised report on the algorithmic language, ALGOL-60. *The Communications of the ACM* **6**: 1–17.

BENNINGHOF, W. S. (1962): Calculation of pollen and spores density. *Pollen et Spores* **4**: 332–333.

DAVIS, MARGARET B. (1965): A method for determination of absolute pollen frequency. In: *Handbook of palaeontological techniques* (B. Kummel and D. Raup, ed.). Freeman, San Francisco, Calif.: 674–686.

— (1967): Pollen accumulation rates at Rogers Lake, Connecticut, during late- and post-glacial time. *Rev. Palaeobotan. Palynol.* **2**: 219–230.

ERDTMAN, G. (1943): *An Introduction to pollenanalysis.* The Ronald Press Company, New York: 199–202.

HARTMAN, A. A. (1968): A study on pollen dispersal and sedimentation in the western part of the Netherlands. *Acta Bot. Neerl.* **17**: 506–549.

IBM (1970): *IBM System*/360 Operating System. ALGOL programmers guide. GC 33–4000–2.

JANSSEN, C. R. (1967): Stevens Pond: A postglacial pollendiagram from a small Typha swamp in northwestern Minnesota, interpreted from pollen indicators and surface samples. *Ecological Monographs* **37**: 145–164.

MAHER, Jr. LOUIS J. (1972): Nomograms for computing 0.95 confidence limits of pollen data. *Rev. Palaeobotan. Palynol.* **13**: 85–93.

MOSIMANN, J. E. (1965): Statistical methods for the pollen analyst: multinomial and negative multinomial techniques. In: *Handbook of palaeontological techniques* (B. KUMMEL & D. RAUP, Eds.). Freeman, San Francisco, Calif.: 636–673.

TROELS-SMITH, J. (1955): Pollenanalytische Untersuchungen zu einigen schweizerischen Pfahlbauproblemen. In: *Das Pfahlbauproblem* (W.U. Guyan, ed.). Birkhäuser Verlag, Basel: 14–57.

WINGEN, J. W. VAN (1972): A proposal for a simple input-output system in ALGOL-60. *ALGOL Bulletin* 33.3.1.