A note on the construction of a general survey program in Extended Mercury Autocode

By A. J. B. Anderson*

The paper describes a general program for the analysis of surveys. The program was written in Extended Mercury Autocode for the Rothamsted Orion and is similar to that written earlier for the Elliott 401, but the derived variate instructions are written separately for each survey in EMA by the user. The steering section is also written in EMA, and complicated cycling can therefore be introduced without difficulty. The program provides for the use of magnetic tape for dumps and storage of tables.

1. Introduction

Many general survey programs have been written for computers. Most provide for only the simpler types of operation and cannot adequately handle surveys with a complicated data structure (e.g. data on families and individuals within a family) or deal with manipulation and storage of completed tables. Yates (1960) outlined the requirements of a general program which would cover all the commonly occurring operations needed in the analysis of surveys of the research type. This scheme was implemented as far as was possible on a small computer, the Elliott 401 at Rothamsted (Yates and Simpson, 1960; 1961).

When this machine was replaced by an Orion, it was decided to construct SEP (Survey and Experiments Program), an entirely interpretive statistical autocode for use on the new machine. As often with such a large project, progress was slower than had been hoped so that, as a temporary measure, it became necessary to write a less sophisticated General Survey Program (GSP). This follows the basic outline of the 401 program, the main addition being the provision of storage facilities for tables, and dumps on magnetic tape. The steering section of the Orion program, concerned with the sequence of operations needed for a particular job, is written by the user in Extended Mercury Autocode, thus affording considerable flexibility in analysis. The standard part of the program need not therefore provide any mechanism for handling the complexities involved in multiple cycles with many card types.

Most of the standard section was written in E.M.A., which greatly facilitated the completion of the project (I.C.T., 1964). The scanning facilities described by Gower (1962), which are incorporated into the E.M.A. Compiler for Orion* proved very useful in writing the table manipulation operations. Any machine with an E.M.A. compiler able to handle these functions can therefore be used with GSP, whereas SEP is written in machine code and is therefore restricted to Orion.

2. Basic structure

The flow diagram given in **Fig. 1** shows the organization of the operations required for a simple survey analysis. For a survey program to be useful, the user must be able to specify the required operations in (*a*) to (*e*) in reasonably simple form. Thus for the formation of a table (Section (*c*)) the variate tabulated, and the variates by which this tabulation is to be classified, must be specified. These specifications, or "instructions", can then either be stored in more or less their original form and interpreted by the program each time they are obeyed, or they can be compiled into machine code by some means before the analysis is begun.

The program can notionally be divided into two parts —sections (*a*), (*b*) and (*c*) forming Part 1, and sections

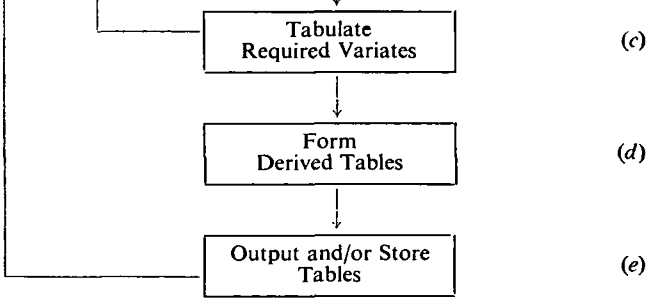

Fig. 1.—Flow diagram for a simple GSP analysis

---

* The E.M.A. version will not scan simultaneously over tables which differ in their factor structure, e.g. **P**, **Q**, **PQ**, or **PQ**, **PR**, **QR**. As Gower pointed out, three scan counts are advantageous for survey work; the lack of them in the E.M.A. version necessitated more involved programming and some restrictions on the table manipulation operations.

* *Agricultural Research Council Scholar, Rothamsted Experimental Station, Harpenden, Herts, and Department of Statistics, Aberdeen University.*

(*d*) and (*e*) Part 2. These are concerned respectively with unit-by-unit input and tabulation, and with manipulation of completed tables. In an analysis of even moderate complexity with many units, Part 1 is traversed thousands of times. Efficiency is therefore likely to be greatly increased if this part is compiled into machine code. Section (*a*) requires some standard card-reading routine. Section (*b*) usually involves operations, such as the evaluation of simple algebraic functions of the variates, for which standard compilers are ideally suited. Unfortunately section (*c*) (in common with Part 2) is essentially an interpretive procedure, in which a great many details must be looked up in indexes to perform the almost trivial operation of tabulation.

Accordingly the following system was adopted. For a particular job, the user provides:

(i) sections (*a*) and (*b*) written in E.M.A. and compiled into Orion Basic Input by the E.M.A. Compiler,
(ii) a series of coded instructions for sections (*c*), (*d*) and (*e*) which are stored and interpreted at run-time by the standard program.

At the required point or points in (*a*) and (*b*), the user enters the standard program (by the E.M.A. instruction DOWN) to perform, say, tabulations. The tabulation instructions coded in (ii) are then interpreted sequentially until a terminator is found, whereupon control returns to the user at the next E.M.A. instruction after DOWN. The fact that the tabulation routine usually need be entered only once per cycle minimizes the relatively slow drum transfers of program. In addition, when successive tabulations involve the same classification set (see Yates and Simpson, 1960), the cell address of the entry is not re-calculated.

3. Storage of tables

To make most use of the storage available in the core during Part 1, tables are formed without spaces for marginal totals. In addition, tables such as counts having only small positive integer entries can be packed with either three or four cells per word, giving limits to cell totals of 65535 and 4095, respectively; a comment is printed each time overflow occurs in any cell.

At the end of Part 1 the completed tables are transferred to the drum store. They can then be called for individually as required in Part 2. When a table is called for it is transferred back to the core, unpacked if packed, and expanded to provide space for marginal totals; these totals are also inserted in the same operation.

An instruction is provided for setting selected tables in their unexpanded form to zero. This is of use when some of the tables are to be formed over several strata. All the tables are returned to the core in unexpanded form at the end of section (*e*), the within-stratum tables being then set to zero.

4. Magnetic-tape operations

The table manipulation operations provided by Part 2 are similar to those of the earlier program (see Yates and Simpson, 1961, Table 1). In addition, magnetic-tape storage instructions are provided, the table store being transferred from core to tape or vice versa in standard E.M.A. blocks of 512 words. This enables later GSP or *ad hoc* E.M.A. programs to make use of the tables produced by GSP. Transfers to magnetic tape are recorded on the printed output as they are made, and any table output by the print instruction is accompanied by information on its location in the core store; thus later retrieval is possible without consulting the original program.

In addition to table storage, the whole data store (i.e. variates, partially formed tables, parameters, etc.) can be dumped at any point in sections (*a*) or (*b*). Then, if a restart is required because of a voluntary or involuntary stop, it is only necessary to restore the last dump and continue from there.

To avoid monopolizing more magnetic-tape decks than are essential, stored tables and dumps are carried on the same reel, interspersed so as to minimize tape winding. As all previous dumps are available, a rerun is possible, starting at any dump.

Since many surveys occupy only a few hundred of the 4096 blocks available on a reel, provision is made for storing the results of several analyses on the same tape. An index is held at the head of the tape, the entry for a particular survey being recognized by its Orion job name. The group of blocks reserved for the particular job is recorded in the index as well as the position and number of the dump immediately preceding the last voluntary stop.

The user always considers these blocks as numbered from zero, the relevant datum being automatically added in by the program. The user can make dumps or store tables only within the section of tape reserved for his job, but can read from anywhere on the reel; this gives access to other surveys or earlier analyses of the same survey, but inhibits inadvertent destruction of material from other jobs.

5. Operating experience

Provision is made for input from magnetic tape as well as from cards. Magnetic-tape input is particularly useful in large jobs where many analyses of the same data may be required. There is the additional advantage that the restart problems associated with card reading are transferred to a separate card-to-tape transcription program; the actual analysis can then be done very quickly, with far less chance of an involuntary stop.

E.M.A. has no standard routines for data input from cards, and restarts caused by peripheral incidents (e.g. card wreck or mis-feed) proved somewhat difficult to implement as peripheral transfers on the Orion are autonomous and some of the programs are branched.

More programming experience with the Orion will be required before E.M.A., GSP and the Orion Monitor Program can be made to handle restart procedures completely satisfactorily.

Improvements in other directions will also be necessary should the program prove of lasting use. Much of the user's E.M.A. section could be standardized; in particular a convenient method of testing whether variates transgress limits should be made available to replace the present somewhat *ad hoc* procedures. Some redundancies in the coded instructions for (*c*), (*d*) and (*e*) should be removed, and provision made for automatic allocation of storage for tables (as in the 401 program). Some loop or jump facility with modification would be useful for the tabulation and table manipulation instructions; this would greatly reduce the number of instructions required.

The increase in speed resulting from using compilation instead of interpretation for sections (*a*) and (*b*) is indicated by a comparison (the only one so far available) with SEP. This job involved six variates with relatively few derived variate operations, and only four tabulations per cycle. SEP performed the analysis at 70 cards/minute as against 130 cards/minute for GSP, using the E.M.A. card-reading routine written by J. A. Nelder of the National Vegetable Research Station, Wellesbourne, Warwick. (The maximum speed of the card reader is 600 cards/minute.) When the GSP tabulation section was used as a routine instead of as a separate chapter (thus avoiding drum transfers of program) the speed increased to 200 cards/minute. With an improved card-reading routine (for numerical material only but with tests for limits incorporated) the speed was further increased to 450 cards/minute. Another job with 460 variates and 20 tabulations per cycle, involving data previously transferred from cards to magnetic tape, was analysed in this way at 1500 cards/minute. This use of the tabulation section as a routine will therefore probably become standard practice.

6. Acknowledgements

The author is indebted to Dr. F. Yates who programmed many of the Part 2 operations, and whose patient help and encouragement made the task seem easy. The author is also grateful to the Agricultural Research Council for providing the studentship under which the work was done.

References

GOWER, J. C. (1962). "The handling of multiway tables on computers," *The Computer Journal*, Vol. 4, p. 280.

INTERNATIONAL COMPUTERS AND TABULATORS LTD. (1964). *I.C.T. Atlas and Orion Computers: Extended Mercury Autocode (EMA)*. I.C.T. publication No. 5052.

YATES, F. (1960). *Sampling methods for censuses and surveys* (3rd edn.), London: Griffin.

YATES, F., and SIMPSON, H. R. (1960). "A general program for the analysis of surveys," *The Computer Journal*, Vol. 3, p. 136.

YATES, F., and SIMPSON, H. R. (1961). "The analysis of surveys: processing and printing the basic tables," *The Computer Journal*, Vol. 4, p. 20.

Book Review

Digital Communications with Space Applications, by S. W. Golomb, *et al.*, 1965; 210 pages. (London: *Prentice-Hall International*, 96s.)

This book is a product of the pioneering space programme of the *Jet Propulsion Laboratory*. It deals with the use of pseudo-random (shift register) sequences in space communication and ranging, and does not give a general treatment of digital communication. There are eight chapters, each almost self-contained, together with four appendices.

The editor's first chapter introduces pseudo-random sequences and indicates their uses. The remaining chapters are the work of four other authors. Up to Chapter 5 the basic ideas and processes involved in the use of pseudo-random sequences for communication are presented. The second chapter gives an interesting account of the methods of generating specified sequences, and includes some fundamental theorems. Chapter 3 deals with the calculation of the power spectra of random sequences from the auto-correlation function, by the standard method.

Various types of orthogonal codes are discussed in Chapter 4 and the use of Hadamard matrices is introduced. Chapter 5 extends the work of Chapter 3 to obtain the power spectra of pseudo-random sequences to specify bandwidth requirements, and concludes by discussing modulation and reception.

Chapters 6 and 8 deal with tracking and synchronization. In Chapter 6 the interest is in ranging rather than communication and the acquisition properties of sequences are investigated; the performance in noise is estimated. The subject of Chapter 8 is the synchronization of pseudo-random communication codes; the 'random source' method in which the value of one digit position is constant, and the 'comma free' method in which no part of two consecutive code words is also a code word, are both considered.

Signalling error probabilities in the presence of noise with optimum detection methods are calculated in Chapter 7 for orthogonal codes; a comparison is made with uncoded communication and the Shannon limit.

Appendix 2 gives a useful list of shift register sequences, while Appendix 3 contains a list of logics and terminating words for counting up to 2047 with these sequences.

The classified origins of much of the work on pseudo-random sequences and their uses has created pockets of knowledge in various organizations, and the dispersion of this knowledge by such books must be welcomed. Although the space application is of limited direct interest the valuable ideas presented have a wide field of interest.

The joint authorship of the book has caused some repetition even to the extent of repeating one figure; little detailed cross referencing occurs (only one author numbers expressions) and the symbolism is not consistent nor always clear. These defects make reading rather difficult; it is, however, rewarding.

D. WILLIAMS