of computer power and research and teaching outputs. The contents page for this report is shown in **Fig. 1.**

The Future?

Proposals to merge the Board with the UGC and SRC were made at a symposium of members from the DES, UGC, SRC, CVCP, etc. in November 1975 (See *Computer Weekly*, 30 Oct. 75, and *THES* 28 Nov. 75). Further, it was also suggested that polytechnics should be invited to collaborate with universities in sharing computer facilities (Computer Consultative Council (1976)).

The Computer Board has served a valuable purpose in pump priming for a decade the crucial need for computing hardware. There is a continuing need to protect:

1. Specifically earmarked funds in the foreseeable future, because the promise of reasonable access of scientists and engineers to online facilities is only just beginning to be realised. In another decade, this should have been consolidated and it is possible that pump priming can then give way to a gradual replacement programme within a global university budget.
2. The maintenance of another channel of communication between academics and Government policy making. Indeed, computing developments might suffer if compelled to compete too directly for resources *vis-à-vis* other sectors of the university.

The growth of computer networks linking universities, polytechnics and Government laboratories, etc. plus the growing pressure to merge the central Government funding mechanism for institutions of higher education, must lead to greater control by the Department of Education and Science. This could lead to the DES being reorganised, responsibility for science and technology being transferred to a new department of state, which would also take over some of the Industry Department's responsibility in this field including the National Research and Development Corporation. Perhaps the Ministry of Science of 1959 will be resurrected and undoubtedly with more than 'a bus load of civil servants'.

Allocation according to goal performance

Particularly at a time of financial stringency, there is a need to allocate computing facilities so that it produces the maximum benefit. The academic merit and contribution to national priorities, of any project requesting substantial amounts of computing, must be judged. This requires tolerably explicit objectives and priorities from each university, plus quantitative measures to monitor a project's performance in trying to attain them.

Computing is generally treated as a free good in higher education. When capacity approaches saturation considerable skill and ingenuity is used to improve the *efficient operation* of the equipment. Project selection and resource allocation criteria, to improve the *effective usage* of the computing facilities to the institution, depend on intuitive appreciation of competing projects' relative priorities. Indeed, until recently the Computer Board has claimed that it is only really concerned with the efficient operation of computers (Norton, 1975).

The effective work achieved by a computer is not measured in CPU cycles, job turnaround, etc. but in the value of the teaching and research it makes possible. Quantitative evidence on the resulting output, like the number of degree courses influenced, publications, patents, etc. with indicators of satisfaction with computing facilities by the priority users and sophistication of usage, do indicate the progression to achieving institution wide objectives, to form a high performance portfolio of computing resource allocations.

References

THE COMPUTER BOARD (1977). *Report of the Computer Board* HMSO Cmnd 6696.

NORRIS, G. (1977). The effective allocation of computing to and within Leeds University, Ph.D. thesis, Leeds University.

COMPUTER CONSULTATIVE COUNCIL (1976). Survey of University and Polytechnic Co-operation, CCC 12/6, The Computer Board.

NORTON, H. (1975). A strategy for computing provision in universities *IUCC Newsletter*, December 1975, page 6.

An interactive graphical minicomputer system for the management and exploration of clinical renal data

M. Gordon

Department of Medicine, Charing Cross Hospital Medical School, Fulham Palace Road, London W6 8RF

A minicomputer system has been developed to help the management and assimilation of multi-variable clinical data for renal dialysis and transplantation. The system employs a variety of peripherals including graphical devices to clarify the trends and correlations in data. To implement a practical system capable of real benefits, it proved necessary to write an extensive system of interlocking applications tasks. The structure and organisation of these tasks is described.

1. Introduction

The computer system described is used in the clinical management of patients who are on maintenance haemodialysis or have renal transplants. These treatments are characterised by the continuous accumulation of data on 40 or more variables (Gordon, 1970) over many years in a gradually increasing number of patients. The closeness of scrutiny possible with conventional records deteriorates as the number of patients increases. The numbers of staff do not increase in proportion and changes of staff and duty rotas mean that a patient cannot see only those doctors and nurses who may know his case in detail. Although much early data remains relevant it is lost from view as new data accumulates. These factors inevitably lead to a deterioration in the quality of care. To take a single example, the one-year kidney survival rate at Johannesburg General Hospital was about 85% for the first 30 transplants, about 75% for the first 100 and was about 65% after 140 transplants (Milne, 1975). These results parallel the experience of dialysis and transplantation at Charing Cross Hospital and, it is thought, that of other competent units. The diminished standard of care undoubtedly follows from the reduced time which staff can devote to each patient, and in particular from the difficulty of following clinical and pathological data closely, and distinguishing the important trends and correlations.

The writer made experiments some years ago which showed that interactive computer graphics could help the assimilation of multivariable data, and rather simple transformations of data could reveal quite unsuspected relationships and complexities (Gordon, 1970; 1973). The falling costs of minicomputer equipment promised to make such methods economically viable, while a dedicated minicomputer concentrated on a limited problem area should produce more flexible and versatile techniques for handling data.

Such a minicomputer system has now been implemented at the Department of Medicine of Charing Cross Hospital Medical School. Two principal problems have had to be overcome. First, while it is clear that graphic terminal devices are appropriate, no device yet marketed appears to be ideal, despite the continuing proliferation of new and cheaper devices. We moreover had to purchase such equipment as funds allowed at different times. One valuable outcome was the development of software techniques which could cope with different hardware environments including an intelligent graphics satellite.

Second, it emerged that the applications software had to comprise many independent but interlocking programs or tasks. A considerable number of tasks had to be operational before any one of them would produce useful results. The operation of graphical or other peripherals was only the final element to be keyed into place in the structure.

The software has been described in detail in another paper (Gordon and Good, 1977). The present paper outlines the hardware, software and data and then describes the set of applications tasks found necessary for a working information system.

2. The computer system

2.1 *Hardware*

The hub of the system is a DEC PDP 11/40 minicomputer with 28K 16-bit words of 0·8 μsec core store, and two RKO5 cartridge disc drives of capacity 2·4 Mbyte each. The peripheral devices include a Hazeltine model 1200 alphanumeric VDU and DEC VT55 graphic VDU; a Versatec D1100A electrostatic printer/plotter; and an IBM golfball printer.

The principal graphic devices are an Intecolor 8001 raster display, incorporating a 23 inch colour video monitor, controlled by an Intel 8080 microprocessor; and a DEC GT40 intelligent satellite display incorporating a PDP 11/05 processor with 8K memory, a display processor with hardware vector and character generators, and an 11 inch refreshed CRT display screen.

All the displays are connected to the PDP 11/40 by serial communication lines at 9600 baud. The devices can operate at a few hundred metres from the PDP 11/40 computer and the Intecolor display is presently in use in the newly opened dialysis ward. We intend later to place terminals in the Transplant Unit and Outpatient clinic. The PDP 11/40 also has a serial communication link at 1800 baud to the Hospital Sigma-6 mainframe computer.

2.2 *Software*

Program development has been carried out under DEC's RT-11 operating system. At first a software package from DEC was used for the GT40, and we deliberately tried to avoid the costly development of systems software. We programmed the PDP 11/40 in FORTRAN. However, while the development facilities of RT-11 were comparatively excellent it did not meet the requirements for run time facilities. The use of FORTRAN proved cumbersome and restrictive, and there was no means to provide for multiple simultaneous users.

We therefore developed a compact executive (NET-11) which can run independently in the GT40 or as either the foreground or background 'job' under RT-11. NET-11 provides for multiple simultaneous users and the execution of multiple simultaneous tasks. It permits close control of the hardware at the cost of programming in MACRO-11 assembly language; this has proved acceptable, particularly since many application tasks can utilise common subroutines and MACRO definitions.

NET-11 includes an implementation of DEC's Digital Data Communications Message Protocol, which permits tasks executing in the GT40 and the PDP 11/40 to communicate. A GT40 task can thus access data from the main data base held on the PDP 11/40, and process it locally. A 'networking' facility allows communication between multiple intelligent satellites.

2.3 *Data*

The Renal Unit at Charing Cross Hospital maintains some 130 patients on haemodialysis, of whom about 100 are on home dialysis, and supports a population of about 30 transplant patients. The clinical data can be broken down as:

Identity data (7 items)
Screen Test data (up to 140 results)
Synoptic histories (up to 512 characters)
Shunt and surgical history (3 variables)
Dialysis data (7 variables)
Blood pressure status (5 variables)
Calcium metabolism (4 variables)
Protein metabolism (3 variables)
Haemological status (5 variables)
Liver function (8 variables)
Diet (7 variables)
Current treatment (up to 40 entries)

Most variables are sampled monthly on average, but sometimes as often as daily. The computer system acquires this data from secretarial, medical and computer staff who key in data to online terminals and from the Hospital computer system which transfers pathology results to the PDP 11/40's data base via the communications line.

3. The operation of the system

The user logs in at a terminal with his password, which will be the same as any he may hold for the Hospital computer system. The system then displays a 'menu' of tasks permitted to that password, and assigns appropriate permissions to access the data base. A task is selected by typing a two character numeric. (Task names are written here as in DEC assembler syntax preceded by quotation marks). The following are tasks available in the system.

The Hospital VDU task, "SG, emulates a DELTA 5500 VDU, and so routes the Hospital VDU reporting service through to the calling terminal.

The data base task, "DB, is a re-entrant task which manages data and indexes and resolves conflicts between multiple users seeking to write to the same record.

The identification update task, "ID, allows updates to identity data which it displays in formatted fields.

The index builder task, "XB, creates master indexes by Renal Unit number, Hospital number and surname. The index sorter task, "XS, sorts the master indexes.

The screen test results (or 'tick list') task, "TK, displays and allows updates to the extensive screen test results given to patients before they start dialysis.

The report generator task, "RP, generates simple alphanumeric reports, particularly lists of patients by name or number.

The Sigma transfer task, "XF, transfers data (normally new chemical pathology results) from the Hospital mainframe computer to the PDP 11/40. It does so by interacting with a FORTRAN program running in the Sigma mainframe. It includes limited error detection and retransmission facilities.

Haematology and microbiology reports are not presently routinely transmitted by communications line, but can be inspected via the task "SG.

The biochemistry update task, "UB, inserts the received data into the data base in a compacted form. The data is stored to an accuracy of 12 bits. A local 3-bit exponent is stored with each fraction, and can be added to a header exponent for the whole record. This technique accommodates the range of variation found in most biochemical variables. Qualifiers such as '<', '>', and alphanumeric codes can also be accommodated. Record overflow is handled automatically.

The new biochemistry results task, "NB, displays newly received data as text files.

The notes task, "NO, allows ward staff to type in free-form notes for the attention of computer staff. These may include updates to stored data.

The news task, "NE, displays news items set up by computer staff.

The graphic retrieval task, "GT, allows the display of several groups of four variables each. Currently the groups may be sodium data, calcium metabolism data, or protein metabolism data, for any patient. The time scale may be selected as 4 months or two years. On the Intecolor terminal a given trace may be displayed in colour, to resolve traces which cross. In the GT40 the graphs are generated locally, and a 'mini' version of the data base task relays requests and data to or from the main "DB task in the PDP 11/40.

The task, "GT, is the current focus of development in the system. The treatment records task "TR, chart generation task "CG and archive task "AR are also under development.

4. Results and discussion

As this paper is written the system has operated for only a few weeks. Therefore, no adequate evaluation of the system's success has been possible. It is clear, however, that users like the system and find it a valuable source of information. Some undoubted benefits are established already, such as the generation of current patient directories, saving the two or three days work previously required of the secretaries; and the use of a preliminary version of the task "CG to generate biochemical charts for all patients, as in a recent review of calcium metabolism. This latter facility alone has saved some scores of man hours of doctors' time.

The fast online access to detailed local records is also appreciated by users, and it is clear that the concept of graphic terminals as information transducers will prove successful. Experienced clinicians have helped evaluate calibrations and layouts for graphs displayed on the Intecolor terminal. It has become clear they can elicit detailed clinical themes and suggestions for improved treatments from the data, which is made more comprehensible by the computer. Their comments and suggestions are often prefaced by the remark 'Once you see the data laid out like this . . .', implying a real improvement on tabular data. Engineering the design and control of graphical formats is a major part of present work.

By comparison with the GT40, the Intecolor is quieter, and expected to be more reliable, since it is less complex. However, its resolution is too limited to be ideal; and it is restricted in the speed with which it can present changing displays. Some intermediate device would be optimum, and we are investigating other combinations of video and microprocessor techniques.

The difficulties of development have related largely to the interlocking nature of the applications tasks. Consider the task "NB, which generates a list of pathology results for the ward sister. "NB must be able to access a complete set of identity records for patients which means that the tasks "ID, "DB and "XB must be fully operational. Delivery of the pathology results requires the tasks "SG and "XF, together with the co-operating FORTRAN program in the Sigma computer. The Sigma program to select data must also be available, as well as the task "UB which loads the data into data base records. The request for the task "NB was made at an early stage of the project, and has acted as a 'seed' round which many other tasks crystallised.

The software developed so far represents four to five man years of development effort, while the hardware capital cost has probably been too high for the majority of potential user sites. However, a fall in hardware prices has occurred since the project started, and the present system has been configured for research as much as for service use. An equivalent system could now be assembled for a considerably lower cost. The successor terminals to the GT40 have however increased in real cost since it was first introduced. The reason for these adverse changes may be a low volume of sales, and one hopes that wider adoption of the techniques described may eventually contribute to lower terminal costs.

5. Conclusion

A minicomputer system has been implemented which incorporates a system of interlocking applications tasks appropriate to the management of multivariable clinical renal data and the use of graphical peripherals for the assimilation and exploration of the data. The initial reaction of users is encouraging, and the way appears clear for further developments in the interactive graphical techniques.

6. Acknowledgement

The writer thanks Professor H. E. de Wardener for his support and encouragement and Mrs. J. Lyall for preparing the manuscript.

References

GORDON, M. (1970). Graphic display for clinician-computer interaction in clinical management, *Bio-Medical Engineering*, vol. 5 no. 11. November 1970, 539-548.

GORDON, M. (1973). Information processing and clinical management, *International Journal of Biomedical Computing*, vol. 4 no. 4.

GORDON, M., and GOOD, C. J. (1977). Software for an interactive graphical minicomputer information system, *MEDINFO* '77 conference

MILNE, J. (1974). Personal Communication.

To the Editor
The Computer Journal

Sir

Joint replenishment quantities

The paper by Adamczewski in the August 1976 issue completely ignores the problem of safety stock which is a major complication of joint replenishment as compared with single item ordering.

Passing over the statement that 'an order is placed for all appropriate items in the group whenever any one item reaches its "ROL",' since this would general large quantities of remnant stock and also ensure that the item order quantities exceeded the desired total ($\sum w_m > \sum f_m$), we may assume that the decision to order is taken accordingly to the joint service objective of the group.

There are two ways in which the safety stock is increased by the use of joint replenishment. Firstly, the safety factor is a decreasing function of order quantity which will be smaller for a normal jointly-ordered item than the corresponding EOQ for the item. This relationship is given by Brown (1959) and Barrett (1969).

Secondly, if the contribution made by each item to expected disservice is sufficiently small, the effect of joint ordering is to introduce a 'review time' equal to the order cycle time, so that the safety stock to be used in allocation is a function of the dispersion of forecast errors over the period

Lead time + expected time to next order.

On this basis, items for which

$$\frac{w_i}{f_i} \geqslant \frac{Q}{\sum f_i} + 1$$

when w_i = stock on hand—safety stock for one lead time may require a small allocation to prevent a premature order being raised towards the end of the current order cycle. Now, as the total of the individual order quantities is a sum of non-linear functions of 'expected time to next order' it is necessary to carry out the allocation iteratively using successive approximations to this period. It is easy to include in such a procedure the requirement that quantities may not be negative.

If the need for ease of implementation precludes the use of an iterative procedure, then a more primitive formula can be used, but it must be recognised that this is at the expense of the planned service and stock total objectives.

I would like to thank Mr. Adamczewski for bringing to my attention the lack of an adequate treatment of this topic in the computer applications literature, although it was discussed in connection with IMPACT (*c.* 1962) and Brown (1967) gives a number of operations research references.

I hope to discuss in a future communication the case where the contribution of one item to expected disservice is not small.

Yours faithfully,

P. J. Coats

Computer Centre
Sunderland Polytechnic
Chester Road
Sunderland SR1 3SD
2 September 1976

References

Adamczewski, B. F. M. (1976). A formula for establishing reorder quantities for individual items within a joint stock replenishment scheme, *The Computer Journal*, Vol. 19, No. 3, pp. 194-197.

Barrett, D. A. (1969). *Automatic Inventory Control Techniques*, London: Business Books Ltd.

Brown, R. G. (1959). *Statistical Forecasting for Inventory Control*, New York: McGraw-Hill.

Brown, R. G. (1967). *Decision Rules for Inventory Management*, New York: Holt, Rinehart and Winston.

Impact (*c.* 1962). *Advanced Principles of Wholesale IMPACT*, International Business Machines Corp.

To the Editor
The Computer Journal

Sir

My heart warms to anyone who presents basic 'computer science' fare, seasoned with practical considerations, in a form which suits the digestion of the commercial applications programmer. By and large, I think that Wilkie (1977) has done this successfully in relation to table look-up. The author has wisely restricted his use of COBOL to a few primitive statements. This has led to a programming style which may offend some readers, but has the advantages of characterising more clearly the relative efficiencies of the various algorithms and (because the statements used *are* so primitive) of making the paper intelligible to non-COBOL programmers.

I should like to draw attention to a danger of this last facet. I am sure that I am not by any means the only Journal reader who gets the flavour of a language he doesn't know by reading papers which use that language in program examples. Readers like myself are prone to assume, unless told otherwise, that the restrictions which the author has in fact imposed on himself are imposed by the programming language. 'This', we assume, 'is how you program in language A—language B, which I know, allows me to do the same thing in a better way'. For the benefit of such readers, it would be well to point out that procedure calls actually exist in COBOL, as do constructs of the **while** and **for** types, despite Wilkie's misleading statement that the latter are available 'in other languages'. Indeed, in relation to table look-up, COBOL provides more facilities than do many other languages. For example, if we assume that the variable *i* has been declared as the (first) 'index' for the table, then Wilkie's sequential search program can be written as the two statements:

```
set i to 1;
search entry
  at end move 'x' to error-mark
    go to A4
  when key-x(i) = input-x
    move result-y(i) to output-y
    go to A3.
```

and, assuming that *key-x* has been declared as a key on which the table is ordered, his binary search program can be written as the single statement:

```
search all entry
  at end go to A4
  when key-x(i) = input-x
       go to A3.
```

(Strictly speaking, the **search all** statement allows the use of a 'non-serial' searching technique which gives predictable results only when the table is key-ordered; binary search is the obvious implementation, but an implementor might determine the searching technique on the basis of table size.)

Finally, may I give one of my hobby-horses a brief trot? It has long seemed to me that physical readability should not be confined to ALGOL-based languages and that the use of lower case and bold type, as above, makes for easier human communication of COBOL algorithms too.

Yours faithfully,

J. Inglis

Department of Computer Science
Birkbeck College
Malet Street
London WC1E 7HX
12 September 1977

Reference

Wilkie, A. D. (1977). Table look-up (with examples in COBOL), *The Computer Journal*, Vol. 20 No. 3, pp. 202-206.

Advances in computer chess

A two-day conference on new developments in computer chess will be held in Edinburgh, Kings Buildings, on 10-11 April 1978. Those with papers or exhibits to contribute should write to:

Mr M. R. B. Clarke
Department of Computer Science & Statistics
University of London
Queen Mary College
Mile End Road
LONDON E1 4NS

who will edit the published record of the proceedings.

The conference will include an exhibition and demonstrations. Conference fee £10 (£5 for students). Accommodation in Halls of Residence, including two meals, approximately £6.50 per person per day. Those wishing to attend should write to:

Mrs P. Montgomery
Machine Intelligence Research Unit
University of Edinburgh
1 Hope Park Square
EDINBURGH EH8 9NW