Comput. & Ops Res., Vol. 3, pp. 327–335. Pergamon Press, 1976. Printed in Great Britain

COMPUTERS AND OPERATIONS RESEARCH: A SURVEY

J. P. C. KLEIJNEN*

Department of Economics and Business, Tilburg University, Tilburg, Netherlands

Abstract—The rôle of the computer as a number-crunching device in operations research (OR) is first investigated including techniques like simulation, etc. The developments in speed, size and costs of the central processor and memory are studied, and also means of reducing software cost; see section 2. Heuristics and interactive man-computer systems are examined, especially their rôles in model formulation, scope, and solution. The discussion includes the various degrees of possible computerization, and the practicality gap between management science and management practice; section 3. The data input into management models can be provided by on-line data-capture (point-of-sale terminals), data bases and Management Information Systems; section 4. The quantification of financial benefits of computerized information systems is examined in section 5. OR may be used to improve the design and use of computer systems; section 6. Conclusions are summarized in section 7. 45 selected references are given for further study.

Scope and purpose—The purpose of this article is to survey the many relationships between operations research (OR) and computers. Computers can help OR by both their computation capability (compare linear programming calculations and simulation) and their data storage capability (they may provide the data needed as input to the OR models). We discuss trends in the speed, size and costs of the various computer components: CPU, main memory, software, etc.

Special attention is paid to a development in OR that is becoming of increasing importance in solving real-life problems, namely iterative, conversational, man-machine systems. These systems do not rely exclusively on algorithmic optimization but instead emphasize heuristic trial-and-error methods. This 'what-if' approach drastically reduces the practicality gap between management science and practice.

Conversely, OR can serve information and computer systems. The design and the running of a management information system should be based on OR techniques to determine which data to collect and how to use them. Also the structure and quality aspects (like accuracy and timeliness) of information systems might be based on modelling.

OR has already been extensively used for the technical performance evaluation of computer systems. The management of computer centers shows fewer OR applications.

45 selected references are included for further study of the above topics.

1. INTRODUCTION

A modern manager is confronted with computers in different ways:

(1) He may be a user of computers as a *tool* for preparing decisions. The computer may be used for (i) number-crunching (linear programming, simulation, etc.), or (ii) data retrieval (data banks and management information systems).

(2) As a *top-level* executive he is responsible for the *cost-benefit* analysis of computers and the information system supported by the computer.

(3) Some managers are solely responsible for the management of the *computer center* itself. In section 6 we present several techniques for tuning computer systems, running jobs efficiently, etc.

First we briefly survey from an OR view-point the history of number-crunching emphasizing trends in speed, size and costs of hardware and software (section 2). Then we discuss the opportunities modern computer systems offer for interactive, man-machine problem-solving (section 3). Next computers as sources of data are presented, emphasizing on-line data capture and retrieval, data banks, and Management Information Systems (MIS) (section 4). The quantification of financial benefits of computerized information systems is discussed next (section 5). OR for improving the design and use of computer systems is also surveyed (section 6). Conclusions are summarized in section 7. We assume that the reader has a basic knowledge of computers; for unfamiliar terminology we refer to[13]. Some recent articles on the impact of computers on OR are[3, 33, 44] which, however, have a more limited scope.

*Dr. Jack P. C. Kleijnen is a Research Associate at the Katholieke Hogeschool Tilburg, Netherlands, where he obtained his Masters and Doctoral degrees. He has worked at the University of California at Los Angeles, Duke University, and IBM Research in San Jose, California. He has published in *Management Science*, *Management Informatics*, *Simulation*, *Technometrics*, and *Statistica Neerlandica*. He is also the author of a book on statistical techniques in simulation. His research interests are in management information systems, quantification of value for information, interactive man-computer systems, statistical design and analysis of simulation, and inventory control.

2. NUMBER-CRUNCHING IN OPERATIONS RESEARCH

Originally computers were used only for doing calculations, so-called number-crunching. In such scientific applications the computer's *speed* is primarily limited by the central processing unit (CPU) and main-memory cycle times since not many input-output operations are needed. CPU and main-memory have shown dramatic increases in speed. One typical source, [17], mentions that in the mid-1950's the average speed of main memory was about 10 μs, in the mid-1960's 1 μs and in the mid 1970's a tenth to a hundredth of a microsecond.* This spectacular growth will not continue since speed approaches its physical limits, the speed of light. What will be the *impact* of increased computer speed on OR? Many OR techniques have become realizable through the speed of the computer. Presently, Monte Carlo simulation is the technique most used in practice (queuing, inventory, transportation, investment analysis, economics, etc.); see the sample surveys in[37] and [43]. Mathematical optimization techniques based on iteration, like linear programming and hill climbing methods (e.g. steepest ascent), have become feasible through the computer's speed; see the papers in[3] for more details. Complicated and frequent forecasts can be performed through regression analysis and exponential smoothing. It has become attractive to use heuristic methods, i.e. methods based on common sense and not necessarily yielding optimum solutions. The latter methods are also affected by other computer developments and will be discussed separately in section 3. The possibility of extensive number-crunching does not mean that all scientific problems are eliminated. For instance, in simulation many problems of statistical design and analysis remain; see[21].

The computers' rôle as a number-cruncher is influenced not only by its speed but also by its *memory-size.* A large memory makes it easier to work with large programs, including data (compare linear programming or regression analysis requiring large matrices). The increase in main memory capacity has been spectacular too: mid-1950's 100 thousand bits, mid-1960's 1 to 10 million, mid-1970's nearly 1 billion bits. Secondary storage has been greatly expanded by the use of discs. Primary and secondary storage have been integrated by the virtual memory technique.

Though the computer's CPU and memory showed tremendous increases in speed and size, this gain was not accompanied by a corresponding increase in *costs.* The cost of memory was 100 cents per bit in the mid-1950's, 10 cents in the mid-1960's and 0.1 cent in the mid-1970's. The application of large scale integration (LSI) will keep hardware expenditure low. Sharpe[38] gives detailed data analyses showing the effects of both technical progress like LSI and "economies of scale" (larger machines require lower average cost). The decreasing costs per unit make large-scale OR applications economically feasible. Smaller-scale computations can be done on cheap minicomputers, even by small companies. Note, however, that communication costs may cancel some of the advantages of LSI. *Software costs,* have also been rising, both relatively and absolutely. The cost of software was 5% of the total (hardware plus software) costs in 1950, 50% in 1965, ±80% in 1970. We expect that software will remain expensive since labor costs continue to rise and this is not offset by increased labor productivity. Moreover, computer systems (including computer networks) and some computer applications are becoming increasingly complex requiring sophisticated software. So software is *the* bottleneck in most computer systems today: "first-generation software on third-generation hardware". Let us classify the various solutions for this economic problem from an economic viewpoint.

(i) *Labor-capital substitution in programming*

Several developments have helped to reduce programming effort. Higher-level languages like FORTRAN, ALGOL, PL-1, and COBOL have replaced assembler languages to a great extent. There is a trend towards languages with a free format and more error checking. Interactive programming (see section 3) and debugging are possible with languages like JOSS, APL and BASIC, and with interactive simulation languages; see Tocher in[3], and [28, p. 72–76, 281–324]. Problem-oriented languages (to be distinguished from procedure-oriented languages like FORTRAN etc.) permit the user to formulate his program the way he is used to think about his problem; see Sussman *et al.* in[3]. The progress in data management software will be discussed in

*Figures on CPU speed, memory size, costs etc. are scattered over numerous sources. For simplicity we use a single reference [17].

section 4. So programming languages can be learned faster, while programming itself takes less time since fewer errors are made and residual errors are detected and corrected more rapidly. These higher-level languages, however, require more compilation and running time, and more memory space. In other words, labor has been replaced by machinery.

(ii) *Mass production of software*

A growing number of *software packages* is offered by hardware manufacturers and software houses. At present there are packages for inventory control, mathematical programming, critical path analysis, financial modeling and risk analyses, statistical analyses, simulation of particular system types (hospitals, computer systems), etc. These packages have been made flexible by the use of *macro generators*, i.e. the software package contains various broad methods of performing a task and the user may specify a particular method by specifying the parameters, whereupon the generator automatically provides a completely specified program or subprogram.

(iii) *Automated production of information systems*

The computerized generation not only of some subprograms but of a whole information system, forms the object of the experimental project ISDOS (Information System Design and Optimization System). Procedures less ambitious than ISDOS are SOP (Study Organization Plan), ARDI (Analysis, Requirements Determination, Design and Development, and Implementation and Evaluation) and TAG (Time Automated Grid System). See [10] for more details.

(iv) *Division of labor*

In *modular programming* a program is split into highly independent parts or "modules" so that a programmer can work on the part to which he is best suited; see[40].

(v) *Life expectancy*

Modular programming means that changes and additions can be more easily realized. High level languages also make programs more flexible since it becomes easier to change the program or to run it on a different computer. The conversion problem is further mitigated by "firmware" (microprogram): basic instruction sets are no longer software but are built into the computer hardware by means of read-only memory modules. This firmware may be used to execute programs originally written for a different computer (emulation).

The above software developments simplify programming, testing and implementing OR techniques. Unfortunately, so far they have not completely compensated the increase in labor costs. For the present, software is likely to be *the* continuing bottleneck.

3. HEURISTICS AND MAN-MACHINE SYSTEMS

Heuristics were described as methods based on common sense, not necessarily yielding optimal solutions. For example, practical inventory-management packages use heuristics instead of, say dynamic programming. A fruitful application area is that of *combinatorial* problems, where the number of combinations is so big that no computer can ever check all possibilities. Examples are job shop, plant layout, assembly line balancing, travelling salesman, depot allocation, critical path calculations, etc. In these examples the problem may be well defined but is hard to solve mathematically. A different class is formed by *ill-structured* problems for which interactive man-computer systems are ideal as we shall explain. See also[31], Little and Thompson in[34], and Newell in[3].

With modern (on-line, real-time) computer systems it has become feasible to interrupt the computer and to give it new instructions depending on the intermediate output, i.e. fast feedback and man–machine *interaction* or *conversation* are created. In the interactive mode the human operator and the computer do that part of the job to which they are best suited: The computer's speed, accuracy and large memory capacity for non-structured quantitative data are combined with man's capacity for creative thinking, learning, pattern recognition and association; see [29, p. 27].* Various aspects of the modeling process are affected by man-machine systems.

*We discuss only the impact a man-machine system has on OR. Its rôle in process control in a manufacturing environment is discussed by Rosenberg in[14]. Meadow[28] describes information retrieval (in libraries etc.), computer assisted instruction, text editing, engineering design (steel structures, automobiles). See also Sussman *et al.* in [3].

(i) *Model formulation*: The user may formulate a model for the system or for a system component, test the model and depending on the computer results either reformulate the model or proceed further with model building or model solution. Interactive simulation is discussed in [25] and [39]. References [29] and [30] give applications with parameters depending on intermediate output and human evaluation, occurring in personnel assignment, media selection in marketing, long term planning, production planning, etc.

(ii) *Model scope*: We need not build a model representing the whole system under study. Instead we may model only those components that behave "well programmable" and let the *non-programmable* components be filled in by the human decision maker. Examples of such unstructured decision problems exist especially at the top management level, e.g., in warehouse location, sales and capacity planning. (Management games are also examples where decisions are not computerized.) Programmable problems that arise only exceptionally, can be left out of the model. If such an exception (say, a machine break-down in a production planning model) does materialize then the human partner takes over. If the programmable problem changes frequently, again *ad-hoc* solutions are more efficient. See also the exposé in [29] and [30].

(iii) *Model solution*: The human participant (or team of participants) may play different rôles in the man-machine dialogue.

(1) He may propose a good *starting point* for an *algorithm*, say a linear programming (LP) or a hill-climbing algorithm. From there on the computer takes over completely. Examples can be found for LP in the oil industry.

(2) The computer contains *data* but no decision models. An example is a geographical data base with data on income per geographic area, location of supermarkets, crimes, population density, etc. The user can change the area boundaries and the computer then calculates the resting attribute values like crime rates. Such a system can be used to find a desirable area configuration. Note that the computer does not compute any *performance function*; the evaluation of an area configuration is completely up to the human participant. Applications can be found in urban planning, market analyses, transportation planning, police zone selection and military studies; see [7].

(3) The user specifies a *solution*, the computer calculates the *consequences* using a *model* (possibly consisting of non-linear and discontinuous functions), the user evaluates the (possibly multicriteria) performance, revises his solution proposal, etc. Finally, the user accepts a "good" solution, not necessarily an optimum but possibly a satisfactory solution taking qualitative and side-effects into account. Examples are provided by numerous simulation models, many in strategic decision-making. The rôle of the computer in this trial-and-error process can be further expanded. The computer may apply an algorithm to try solutions "around" the solution proposed by the user (keeping track of its previous trials); see [35]. The computer may try to learn from its past trials (artificial intelligence). It may also compute a solution for a simplified system (smaller in scope or in detail); this solution may guide the user in his solution proposal.

The above developments in the model formulation, scope and solution may help to fill the *practicality gap* between management science and management practice. The decision maker may get more involved in both model building and model solution. Rather than unrealistic optimization exercises, "*what if*" questions become the purpose of OR modeling. Besides, modeling may now be applied not only at the operational management level but at the *strategic* level too. Examples of such top-management models are the increasing number of corporate financial models based on the company's balance sheet, formulated in interactive simulation by managers themselves; see [29, 30, 36]. Other strategic models are "risk analysis" models, i.e. capital budgeting with subjective probabilities as inputs. Little describes interactive marketing models in [34]. At present a variety of interactive models are in actual operation in, e.g., the electronic and steel industry at the operational and strategic level. We believe that man-computer cooperation will indeed profoundly affect the OR practice and theory.*

4. DATA BANKS AND MANAGEMENT INFORMATION SYSTEMS

In this section we concentrate on the service to management by the computer's *electronic data processing* (*EDP*) capability. Most OR textbooks emphasize techniques and assume that

*In [3] Sussman *et al.* discuss computer prerequisites for interactive problem-solving, *viz.* graphic in/output via cathode ray tubes, time sharing, problem-oriented languages, and a flexible data base permitting data structures like trees and networks.

the values of parameters and exogeneous variables are known. In practice, gathering these values may be very time consuming or even impossible. EDP may help in a number of cases.

As an example consider a retail inventory system. To run such a system data are needed on sales and prices of as many as a million articles ("stock keeping units"). To obtain these data *point-of-sale* equipment can be used, i.e. at the supermarket's check-out sales transactions per item are recorded by electronic equipment that is connected with a computer. On-line data capture gives more timely, accurate and detailed information. Moreover, more frequent reviewing of the inventory becomes possible. See[19]. Other examples of on-line data capture, and data retrieval are found in banking, travel agencies, manufacturing, etc.; see[29].

Computers can mitigate the problem of lack of data though problems do remain; compare the data on loss of goodwill in an inventory system. It is interesting that in one experiment it was found that accurate data with rough modeling gave better results than inaccurate data with sophisticated OR techniques; see Ackoff and Beer in [3]. In [29, pp. 171–179, 222–226] examples are given of simulation models for production planning and maintenance, and LP models for integrated sales, production and capacity planning, where input data are readily available since the OR models are embedded in a MIS. We now consider data banks and MIS.

Traditionally an EDP system contains a number of *files* (ordered collections of records), e.g. files on inventory, and accounts receivable. Certain data occur in more than one file. Consequently files may contain duplicated information. Updating this information cannot be achieved simultaneously so that discrepancies among files may be created. In a *data bank* data occur only once. Relations among the records formerly forming a physical file, are created by "links" (the address of the next record is stored in the current record). Physically the records are stored on random-access devices like discs whence they can be retrieved using their addresses. Since the records are linked forming data structures like networks, the user can question the data bank to answer *ad hoc* questions like "how many employees work in a warehouse receiving more than x articles per month". The software needed to maintain such a data base is very sophisticated while also processing and data storage overhead is incurred. For example, if a new record is created this "data management" software must update all links with other records. In addition, the user should be able to operate on a single data base using various programming languages: "program independence" of data. Progress to such bases is being made by systems like IBM's IMS (Information Management System) and CODASYL's DBTG (Date Base Task Group) proposal; see [5] and [8].

In a data base with on-line data capture each activity of the company leads to the creation or change of one or more records. The data in the data bank can be aggregated (after retrieval, on-line or in batch mode) to form information on the strategic level, so called "*bottom-up*" approach. Alternatively a strategic, corporate model may be based on highly-aggregated data collected separately, the so-called "*top-down*" approach. The *scope* of the data base can be broadened by connecting data banks in computer networks and by inserting national economic data into the companys' database.*

In a *Management Information System* (MIS) we may distinguish the following characteristics.

(1) The MIS contains an on-line *data bank.* The user can ask planned and *unplanned questions* since the data are linked and may form network structures.

(2) The system guarantees *security and reliability.* Security means that data are not available to or cannot be changed by unauthorized users. Security can be assured by proper hardware and software, as well as by non-computer provisions like computer center lay-out, organization, and legal measures; see[8] and [26]. Reliability means that the system has a high probability of performing correctly. Reliability has been improved drastically by more robust hardware and software able to detect and correct parity errors, and multiple hardware components. If nevertheless the computer breaks down back-up and recovery procedures make it possible to switch to another computer temporarily.

(3) *Operations research* is needed to determine *which data* should be collected and stored in the data bank, and how to *use* the data properly. In [2] Ackoff emphasized that the data bank and

*A particular type of data bank is one containing OR literature; its storage and retrieval are discussed in [28, pp. 145–181]. Computers can also be used to teach OR techniques (computer assisted instruction or CAI); see [28, pp. 218–243]. A standard application is the teaching of OR via business games; see Ackoff and Beer in [3, pp. 542–547], [11] and [23].

the manager, should not be flooded by irrelevant data (information pollution). Moreover, he claims that the manager does not know which data he really needs. The relevance of data is guaranteed if the data are needed by a model. See also Kriebel in[34]. Such a model may concern a subsystem within the company (e.g. the inventory system), or the whole corporation (a corporate simulation). To *use* the data properly the manager can be provided with OR techniques like LP, regression analysis, etc. As we noticed in section 3, interactive systems permit the extension of scientific modeling from the operational level to the strategic level. Ideally, the MIS provides an exception report when later on the actual output significantly deviates from the expected model output, so that the model should be adjusted and action should be taken; see [2] and [29, pp. 190–192].

A MIS satisfying the above three requirements is not in existence yet. There do exist MIS permitting only planned questions within a specific area of the company, e.g. sales, and at a specific management level, the operational level. Some authors doubt if a full-fledged MIS will ever be realized. Other authors feel that the advances in computer hardware and software (especially data management) and scientific management style will lead to MIS. For additional reading see[6] and [42].

5. COST-BENEFIT ANALYSIS OF INFORMATION SYSTEMS

The information system (IS) is the nervous system connecting all parts of the organization. We limit our attention here to the computerized part of the IS. Requests for (major) changes by users of the IS or by the computer center itself must be approved of by top-management. To evaluate the financial benefits of computers we distinguish two basic application areas.

(i) *The replacement of clerks*

Examples are the computerization of payroll preparation, accounts receivable and payable, etc. Such computerization means a more capital-intensive method of production. A comparison between the costs of producing basically the same (physical) output (invoices, etc.) should reveal whether a cost reduction can be achieved. Sometimes a revenue increase occurs, e.g. when computerization results in faster invoicing interest is earned. Anyhow, the economic analysis is straightforward.

(ii) *Improved decision-making through better information*

Computers affect the quality of the immaterial information so that better decisions can be made, resulting in either decreased costs (e.g. inventory costs) or increased revenues (sales). It is useful to distinguish between operational and tactical-strategic decisions. Better *operational* decisions, say inventory management, are possible through the use of scientific models (sections 2 and 3) or better data (section 4). The data is better because it is more timely, accurate, detailed, etc. It is rather straightforward to develop a model of the operational process to show how improved information affects costs and revenues. For instance, in inventory management increased bookkeeping accuracy means smaller statistical variance so that safety stocks can be reduced which results in lower inventory costs; more frequent reviews (timely information) reduces both safety and working stock; see[19]. Other examples are dampened capacity requirements by airline and railway reservation systems; decreased interest loss and risk through cash and debt management in banking systems; see[45]. Measuring revenue increases is easy in the above examples; difficult in other cases (goodwill effect of improved inventory management). It is always hard to model the effect decisions have on the relations between the company and its environment, i.e. its customers and competitors. For that very reason most *tactical-strategic* decisions are much more difficult to evaluate; for an example we refer to[23].

In practice most companies do not quantify financial benefits of their IS (*ex ante* or *ex post*, i.e. after installation). Most times *technical* evaluations of the computer part are performed, yielding criteria like throughput and response time; see next section. Sometimes it is tried to measure these technical variables together with other variables like flexibility and costs, assigning subjective weights to each variable so that a final score of particular IS can be computed. As disadvantages of this *scoring approach* we see: (1) Subjective weights. (2) Oversimplified, linear selection process. (3) Applications are considered given and must be realized in the most efficient way considering restrictions like response time (cost minimization, no revenues considered). (4) No causal model: it is not shown how inputs affect output.

Another approach that explicitly studies the effect of the information quality on the results of a company is *Bayesian decision analysis*, or *information economics*. Its elements are: (1) Surprise content of information, or difference between prior and posterior distribution of outcomes, the latter distribution incorporating new information. (2) Effect of the information on the decision, and of the decision on the performance: sensitivity analysis. For instance, in inventory management the square root formula implies that inaccurate parameters have a dampened effect. Unfortunately this approach has several drawbacks: (1) Only one information characteristic is considered, viz. accuracy. (2) No dynamic decision-making is considered while actually information influences decisions which in turn affect the probability of future events, etc. The method has its merits when determining whether additional information is needed for a one-shot decision, e.g. an investment decision. For the study of computerized IS the following approach seems more promising.

Simulation may be used to study the (multiple) effects exercised by various structures of the IS and quality characteristics of information: age, accuracy, detail, presentation mode, etc. We can model the structure and the characteristics, and the relationships among variables. Indeed such studies have been done using several forms of simulation, viz. Industrial Dynamics, discrete-event simulation, etc.; see [4, 23, 41, 45]. Also man-machine simulation or gaming has been used; [11]. We refer to [22] for more details on the various approaches to benefit quantification for an IS.

6. OPERATIONS RESEARCH FOR COMPUTER SYSTEMS

A computer system comprises many subsystems: hardware subsystems (CPU, primary and secondary memory, peripherals, channels), and software subsystems (operating, programming, data support and application software). At another level of detail these subsystems consist of components and subcomponents, e.g. secondary memory comprises several disc packs, each pack containing a number of discs. Interactions occur among the components of the computer system because there is traffic among them. Moreover, human operators are required to make the system work. Depending on the problem we want to solve, we can model the whole computer system, a particular subsystem or a specific component.

If the model represents the whole computer system and has to represent the effects of specific subsystems then usually *simulation* is applied. This technique is well-suited to study a time-sharing system with its queueing, job shop and allocation problems (when to execute which program; where to store programs and data); see [28, pp. 99–110]. Tutorials on simulation applied to computer systems evaluation are given in [24] and [12, pp. 147–172]. In addition to *discrete-event* stochastic simulation, *trace driven* simulation has been developed. It uses as input the deterministic empirical data obtained by executive specific programs on a specific computer, while tracing these programs. The input is then processed by a model simulating an alternative computer. This model is usually less detailed than a discrete-event simulation model; see[12, pp. 172–197]. Just as packages have been developed for systems such as inventory control, generalized simulators of computers have become available. For discrete-event simulation we refer to[25]. SCERT (System Computer Evaluation and Review Technique) is a well-known example of a simulation package not using the clock mechanism of discrete-event simulation. In addition SCERT contains a file with data on technical and economic characteristics of hardware and software components available on the EDP market. Note that the input data can be provided by the computer itself in so far as data on an existing system are concerned. For the computer can measure its own activity through hardware and software monitors; see[12, pp. 219–280], [15] and [25]. For a very extensive bibliography on simulation computers we refer to[32]. When studying a small subsystem or a whole system with little detail analytical approaches can be tried.

Queuing theory has been applied to *time-sharing* computer systems. The queuing discipline is a very important factor in such systems. The basic discipline is the "round-robin", where on completion of his time allocation the customer is put at the end of the queue, the queue discipline being first-come-first-served. Alternative disciplines, however, are possible. Another factor of interest is the length of the time slice allocated to a job. For instance, a longer time slice decreases the probability of having to remove the current customer and having to fetch a new customer whose program may reside on disc so that "swapping" is required between primary and secondary storage. An excellent introduction and survey is given in[15] and [27]. Analytical models for multiprogramming with virtual memory techniques (or "paging") are investigated in[15]. Networks of computers are analyzed by Kleinrock in[1].

There are a number of miscellaneous analytical models. The above models concentrated on *CPU scheduling* algorithms. Another group concerns *data management* models; see[18], [32], [38, pp. 363–441]. Analytical models for the *whole* computer system must be rather crude; see[18] and Kriebel in[34]. Note that the above OR models are alternatives to simple computer science techniques of computer evaluation, like instruction mixes and benchmark programs; see [12], [15], [38, pp. 295–314].

OR can play a role not only in the design of a computer system but also in its *running*. The computer center can very well be compared to a *job shop*: There are a number of jobs each having its own characteristics, viz. its particular requirements for CPU, printer, etc. These jobs compete for the limited capacities of the available "machines". In running a computer center it must be decided when to execute which job. Usually this is done manually but heuristic job scheduling rules can be proposed by an OR model or the human participant; a model of the computer system and the job stream evaluates the proposed schedules; see[9]. Other problems arising when running a computer system are: Where to store information such that seek time and storage space are balanced, how to assign priorities to jobs being simultaneously in the multi-programming system; see[9] and [18, p. 588]. The maintenance of computers is discussed in[16] and [12, pp. 9–15].

7. CONCLUSION

Computers have become an indispensable tool for operations researchers. Time-consuming applications have become feasible through the increased speed and the memory size of computers: simulation, linear programming, inventory control, etc. Economically these applications have become more attractive through the decreased unit costs and, for smaller users, through time-sharing and minicomputers. Software remains a major problem, both technically (complicated software systems, such as sophisticated operating systems) and economically: high labor costs not offset by productivity increases though some encouraging developments are on the horizon.

A well-known problem is the gap between management science and management practice. Interactive *man-machine* systems are becoming available that may exercise a dramatic effect on this gap. These interactive systems also mean that OR may experience a structural change, from optimization algorithms to "what if" heuristics, and from the operational to the strategic decision-making level.

Besides number-crunching the computer may supply OR models with the necessary *input data*, an aspect often neglected in traditional OR literature. A new development here is the creation of *data banks* with network relations among data. Actually such a data bank forms only part of a larger information system, the MIS.

The functioning and *cost-benefit* analysis of a *MIS* requires pioneering research in OR. Several approaches were discussed in this paper.

The use of OR in the *technical design* and performancy evaluation of computers is well accepted. OR for computer center management is still a rather underdeveloped area.

Acknowledgements—This paper is based on some ideas in[20] and [22] to which we refer for 50 more references. We gratefully acknowledge the comments on an earlier version received from E. Carlson, IBM Research San Jose, R. Gilman, California State University Fullerton, and an anonymous referee. Any deficiencies remain the author's sole responsibility.

REFERENCES

1. N. Abramson and F. Kuo, eds., *Computer-Communications Networks*. Prentice Hall, Englewood Cliffs (1973).
2. R. L. Ackhoff, Management misinformation systems, *Mgmt Sci.* **14**, B147–B156 (1967).
3. J. S. Aronofsky, ed., *Progress in Operations Research, Relationship Between Operations Research and the Computer*, Vol. III. John Wiley, New York (1969).
4. D. F. Boyd and H. S. Krasnow, Economic evaluation of management information systems, *IBM Systems Jl.* **2**, 2–23 (1963).
5. R. G. Canning, The current status of data management, *E.D.P. Analyzer* **12**, 1–14 (1974).
6. M. R. Carr, A survey of management information systems literature, *Computer Bull.* **15**, 218–223 (1971).
7. E. J. Christiani, R. J. Evey, R. E. Goldman and P. A. Mantey, An interactive system for aiding evaluation of local government policies, *IEEE Trans. Systems, Man and Cybernetics*, SMC-3, 141–146 (1973).
8. *Codasyl Data Base Task Group April* 71 *Report*. Association for Computing Machinery, New York, April (1971).

9. T. M. Cook, Schedule-constrained job scheduling in a multi-programmed computer system, *Winter Simulation Conference*, Vol. 2, pp. 674–685. 14–16 Jan. (1974).
10. J. D. Couger and R. W. Knapp, *System Analysis Techniques*. John Wiley, New York (1974).
11. G. W. Dickson, J. A. Senn and N. L. Chervany, *Research in Management Information-Decision Systems: The Minnesota Experiments*. MISRC-WP-75-08, Management Information Systems Research Center, Graduate School of Business Administration, University of Minnesota, Minneapolis, Minnesota 55455, May (1975).
12. M. E. Drummond, *Evaluation and Measurement Techniques for Digital Computer Systems*. Prentice-Hall, Englewood Cliffs (1973).
13. I. H. Gould, ed., *IFIP Guide to Concepts and Terms in Data Processing*. North-Holland, Amsterdam (1971).
14. F. Gruenberger, ed., *Fourth Generation Computers: User Requirements and Transition*. Prentice-Hall, Englewood Cliffs (1970).
15. H. Hellerman and T. F. Conroy, *Computer System Performance*. McGraw-Hill, New York (1975).
16. J. I. S. Hsu, An empirical study of computer maintenance policies, *Mgmt Sci.* **15**, B180–B195 (1968).
17. E. C. Joseph, Management and computers in the 70's, *Informatie* **14**, 586–590 (1972).
18. S. R. Kimbleton, Performance evaluation; a structured approach. In: *Proceedings AFIPS* 1972 *Spring Joint Computer Conference*, AFIPS Press, Montvale (1972).
19. J. P. C. Kleijnen, *Inventory Systems for the Grocery Retail Trade*. Katholieke Hogsechool, Tilburg, Sept. (1973).
20. J. P. C. Kleijnen, *Computers and Management Science: their Interface To-day and To-morrow*. Instituto Nacional de Racionalizacion y Normalizacion, Madrid, 1974.
21. J. P. C. Kleijnen, *Statistical Techniques in Simulation*. (In two parts). Marcel Dekker, New York (1974/1975).
22. J. P. C. Kleijnen, Economic evaluation of information and computer systems: a survey, *Euro I First Congress on Operations Research*, 27–29 Jan. (1975).
23. J. P. C. Kleijnen, *Het Economisch nut van Nauwkeurige Informatie: Simulatie van Ondernemingsbeslissingen en Informatie*. (The economic benefits of accurate information: simulation of corporate decisions and information). Katholieke Hogeschool, Tilburg, Nov. (1975).
24. H. M. MacDougall, Computer systems simulation: an introduction, *Computing Surveys* **2**, 191–209 (1970).
25. J. N. Maguire, Discrete computer simulation—technology and applications—the next ten years. In: *Proceedings AFIPS* 1972 *Spring Joint Computer Conference*, AFIPS, Montvale (1972).
26. J. Martin, *Security, Accuracy and Privacy in Computer Systems*. Prentice-Hall, Englewood Cliffs (1973).
27. J. M. McKinney, A survey of analytical time-sharing models, *Computing Surveys* **1**, 105–116 (1969).
28. C. T. Meadow, *Man-Machine Communication*. Wiley-Interscience, New York (1970).
29. P. Mertens and J. Griese, *Industrielle Datenverarbeitung, Band* 2: *Informations- und Planungssysteme*. (Industrial data processing, volume 2: information and planning systems). Verlag Dr. Gabler, Wiesbaden (1972).
30. P. Mertens and H. Kress, Mensch-Maschinen-Kommunikation als Hilfe bei Entscheidungsvorbereitung und Planung. (Man-machine communication as an aid in decision-making and planning.) *Zeitschrift für Betriebswirtschaftliche Forschung*, **22**, 1-21 (1970).
31. G. C. Michael, A review of heuristic programming, *Decision Sci.* **3**, 74–100 (1972).
32. E. F. Miller, Bibliography and KWIC index on computer performance measurement, *Report RM*-1809, General Research Corporation, Santa Barbara, California, June 1973.
33. A. Mjosund, The synenergy of operations research and computers. *Ops. Res.* **20**, 1057–1064. (1972).
34. J. F. Pierce, ed., *Operations Research and the Design of Management Information Systems*. Technical Association of the Pulp and Paper Industry, New York (1967).
35. J. B. Schneider, J. G. Symons and M. Goldman, Planning transportation terminal systems in urban regions: a man-machine interactive problem-solving approach. *Transp. Res.* **6**, 257–274 (1972).
36. A. N. Schrieber, ed., *Corporate Simulation Models*. Office of Publications, Graduate School of Business Administration, University of Washington, Seattle (1970).
37. R. E. Shannon and W. E. Biles, The utility of certain curriculum topics to operations research practitioners, *Ops. Res.* **18**, 741-745 (1970).
38. W. F. Sharpe, *The Economics of Computers*. Columbia University Press, New York (1969).
39. R. C. Sohnle, J. Tartar and J. R. Sampson, Requirements for interactive simulation systems, *Simulation* **20**, 145-152 (1973).
40. W. G. R. Stevens, *Modular Programming and Management*. Pall Mall Press, London (1971).
41. C. V. Swanson, Evaluating the quality of management information. Working paper 538–71, Alfred P. Sloan School of Management, M. I. T., Cambridge, Massachusetts, June (1971).
42. R. I. Tricker, *Management Information Systems, an Annotated Bibliography*. The General Educational Trust of the Institute of Chartered Accountants in England and Wales, London (1969).
43. E. Turban, A sample survey of operations-research activities at the corporate level, *Ops. Res.* **20**, 708–721 (1972).
44. A. Vazsonyi, Why should the management scientist grapple with information systems. *Interfaces* **3**, 1-18 (1973).
45. M. Verhelst, *Contribution to the Analysis of Organizational Information Systems and Their Financial Benefits*. Faculteit der Economische Wetenschappen, University of Louvain, Louvain (1974).

(*Received May* 1976)