Technical Report CS78005-E

STYLITISM, SYNERGISM AND SYNCRETISM:

THE INTERFACE OF COMPUTER SCIENCE AND OPERATIONS RESEARCH*

Richard E. Nance

Department of Computer Science
Virginia Polytechnic Institute and State University
Blacksburg, Virginia 24061

July 1978

*To be published in the Proceedings of the 1978 ACM Annual Conference, Washington, D. C., December 4-5.

ABSTRACT

The interface of computer science and operations research is described from a stylitic perspective. This position enables an honest appraisal of the disciplinary synergism often claimed for the two. Operations research emphasizes the development of algorithms and the implications therein; while computer science gives primary emphasis to the representation of algorithms and the implications of execution on a digital computer. The realization of a syncretic state can be claimed only for discrete event simulation and, to a lesser extent, for scheduling theory. To the extent that the "algorithm" can serve as common ground for both disciplines, a broader, more fundamental form of syncretism might be achieved.

Keywords: practicality gap, joining disciplines, algorithm, interactive problem solving, modeling

CR Category: 1.59

Stylitism, Synergism, and Syncretism:

The Interface of Computer Science and Operations Research

In 1970 I authored a paper dealing with the interface of Operations Research and Computer Science, which was presented at the 37th Meeting of ORSA on April 20-22, 1970. [NANCR70]. That paper, which included the thoughts and opinions of several eminent computer scientists, has proved to be interesting reading some eight years later. My view of the interface between the two disciplines has changed but slightly in that time; nevertheless, certain intersections of the two disciplines are now more clearly defined. Nevertheless, the belief that I have a better picture of the computer science and operations research interface is probably as illusory now as it was in 1970.

A CHRONOLOGICAL CONTEXT

In describing the interface of the two disciplines, I am impressed by the striking similarities. Both disciplines have early historical roots. In fact, they share an important historical personage in Charles Babbage, whose contributions to the concept of a stored-program digital computer are well known but whose early work entitled Economy of Manufactures and Machinery (1832) is not. Both disciplines must acknowledge their origins with the military and sometime during or just after the Second World War. Operations research took its first steps as part of the British air defense efforts [ACKOR68], and the first electronic digital computer, the ENIAC, was a project supported by the Ordnance Corp of the U.S. Army [ROSES69].

The similarities in development extend beyond the early days, in which both disciplines found academic foster homes in mathematics or engineering departments. As each developed its identity, a growing concern was expressed by academics in both disciplines for the development of underlying theoretical foundations [PERLA67], [DENNP71]. For some time both computer science and operations research have experienced the claims of a chasm between theory and applications [HEAND65], [KINGW67], [STAHP67], also the ACM Forum in the following issues of C.ACM: 17(6), 18(1), 18(11), and 19(5)). The response has ranged from consideration of a proper relationship between theory and practice [BHATU69], [FORSG69], [HAMMR69], to the statements of caution regarding the academician's undue emphasis on theoretical aspects [FORSG67], to truculent criticism of the academic preoccupation with theory [WOOLG78].

The relationship to mathematics is recognized as a similarity between the two disciplines. However, that similarity sparks its own diversity; in that computer science relates more closely to discrete than to continuous mathematics, while the reverse is true for operations research. This contrast in the two disciplines sparks another: the identification of a nucleus for each discipline. The nucleus of computer science is undoubtedly the machine (in existence or in concept). However, the nucleus of operations research is not defined, at least not to this observer. Within this brief chronological description, which emphasizes the similarities in development but also notes some important contrasts, I believe that a more accurate impression of the current interface of computer science and operations research can be obtained.

THE PRIOR PERSPECTIVES

Prior commentaries have dwelled on explicating the relationship between operations research and computers rather than between operations research and computer science. This perspective was defined in third volume of Progress in Operations Research, which described the interface with computers [ARONJ69] and is sustained in later papers. The series of papers by Kleijnen [KLEIJ73], [KLEIJ75], [KLEIJ76], [KLEIJ77] provide an evolving tutorial on computing technology coupled with an emerging picture of the "computer influences" on OR techniques. Kleijnen also cites the contributions of discrete event simulations, queueing theory and scheduling theory to the management of computer systems and management information systems. In the final paper of the series [KLEIJ77, pp. 25-26], he speculates that computing technology may "reduce the practicality gap between management science and management practice."

The perspective of the mutual influence of OR and CS as filtered by "computers" is shown in Figure 1. While perhaps simplistic, Figure 1 assists in formulating a crucial question: Do the two disciplines influence each other only indirectly? I reserve my answer to that question until later.

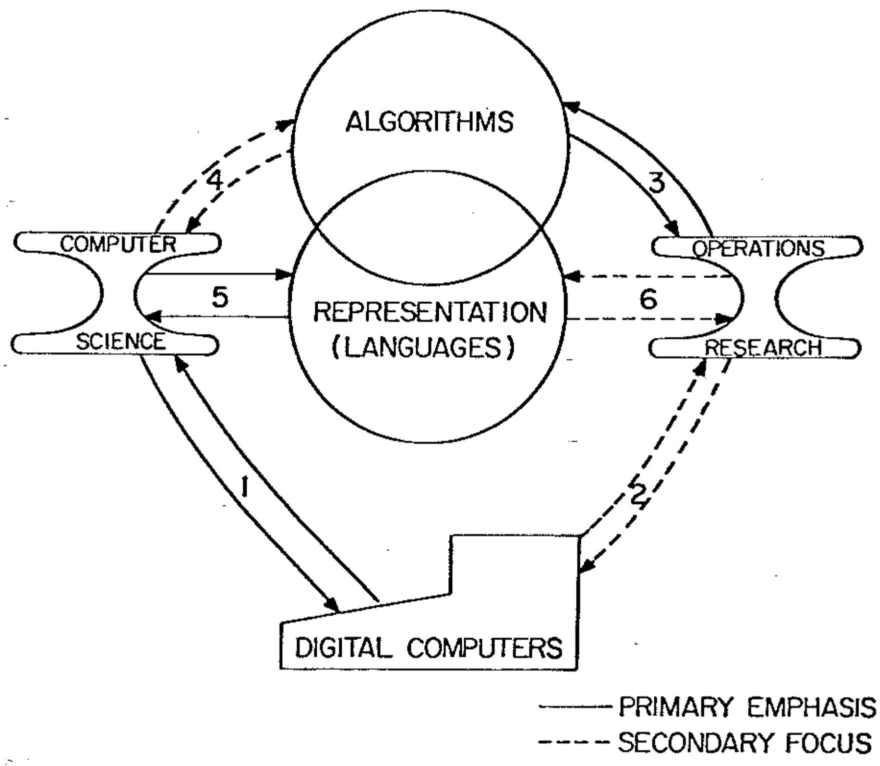

Figure 1. Illustration of the Relationships Between Computer Science and Operations Research

Mjosund [MJOSA72] uses the term "synergistic" in describing the interrelationship between OR and computers. He expresses regret that "the problems of computer use do seem to remain on the outskirts of operations research" [MJOSA, p. 1059]. Mjosund concludes that problems "surrounding the use of computers" are an important area of concern for future OR applications.

Obviously, research and the advancement of a discipline are not motivated by the desire to effect multidisciplinary ties. Loops one and two in Figure 1 are drawn differently because of the realization that the computer is the *raison d'etre* for much computer science research, but it serves only as a problem-solving tool for operations researchers. Engel [ENGEJ74, p. 4] describes OR as a joining (his emphasis) discipline, bringing together the insights, techniques, and tools of other disciplines in order to solve problems.

A STYLITIC PERSPECTIVE*

Engel's characterization of operations research as a joining discipline stimulates me to consider whether computer science is a joining discipline. In essence, does the computer always serve as a filter between computer science and some other discipline? If the joining characterization is accurate, then a explanation of the "indirect" mutual effects and the paucity of published research on interface problems (see Mjosund [MJOSA72, p. 1059]) is possible. The representation in Figure 1 takes on even more significance in assisting this explanation, which might be argued as follows:

> Operations researchers who utilize the computer as a tool consider that facet of their research as secondary. Their concentration is focused narrowly within the third loop (e.g., the development of a more efficient network flow algorithm) and the "best" computer representation of this algorithm (loop 6) is an implementation detail. In contrast, the computer scientist dwells on loops 1 and 5, preoccupied with the computational difficulty and the representation issues, and tending to minimize the broader aspects of algorithm development for problem solving (loop 4).

While perhaps overstated, this explanation also provides clues as to the sources of the "practicality gap" referred to by both Kleijnen [KLEIJ77] and Mjosund [MJOSA72], and described in numerous letters to Communications of the ACM and in the deliberations regarding ACM publications.

In taking the stylitic perspective, I am seeking to rise above the need to document the lack of, or the increasing amount of, interaction between computer science and operations research. In my opinion, no conclusions can be drawn from the fact that seven of 49 papers in the 1977 volume of J. ACM deal with operations research. Nor is it significant that of 94 papers in the 1977 volume of Operations Research I counted only six as dealing essentially with computers. The crucial question is: In what ways have the disciplines of computer science and operations research mutually affected each other, either to the benefit or detriment of one or both, and how is this interdependence likely to manifest itself in the future?

*Stylite - any of a class of Christian ascetics of the early Middle Ages who lived on the top of pillars (Webster's New World Dictionary, 1970).

THE SYNERGISTIC ARGUMENT*

In my opinion the most important contribution of computer science to operations research is the modeling laboratory. This contribution is also probably the least used within operations research. However, the ability to develop solutions, which can be tried and tested or used in a learning situation, contributes to an understanding and appreciation of relationships, assumptions, and consequences that otherwise would be impossible. Currently planned expansion to include the interactive formulation and solution of problems will enable modelling laboratories unlike anything in use today. When the computer provides assistance during problem definition, solution strategy recognition, algorithm development, and algorithm construction and testing, we will approach the potential necessary for problem-solving. This is the promise of the interactive approach to problem solving.

It is true that the computer represents an important applications area for operations research techniques. Operations researchers have found similar problems and quite a few more fascinating extensions of these problems in the areas of scheduling, queueing and inventory theory. I find it a little surprising, and even a little amusing, that current warehouse and materials handling problems often use the term "storage and retrieval," which I associate with information storage and retrieval. That the most advanced materials handling techniques today attempt to maximize random access must not be accidental. Additionally, the ability to employ approximate solutions in more complex models is a clear contribution of computer science. Some might even claim it to be the most significant effect.

At least one detrimental effect can be attributed to the effect of computer science on operations research. This is the acceleration of the divergence between implementation and theory. The knowledge that is required to develop efficient algorithm representations for implementation serves as a barrier to those who might be otherwise encouraged to carry research results further. A few individuals however, like James Kalan, Darwin Klingman, Donald Gaver, and Harvey Greenberg, have ventured into the area of algorithm implementation and programming system development. No doubt, they were stimulated by the work of William Orchard-Hays and Eli

*Synergism - simultaneous action of separate agencies which together have greater total effect than the sum of their individual effects (Webster's New World Dictionary, 1970).

Hellerman, who very early sought to develop large scale mathematical programming systems on second generation machines. However, these people often found themselves talking to almost no one outside a closely knit little group, and their work had a difficult time being recognized. To the extent that computer science assisted in crystalizing the theoretical and implementation camps, I see it as exerting a negative effect.

In its effect on computer science, I believe that the most important operations research contribution is the modeling perspective. The concern for evaluation of systems, the tools for analysis and performance assessments, and the economic questions themselves followed from the early work of operations researchers on computer systems. I believe that the theoretical framework for operating systems, described so ably in the text by Coffman and Denning [COFFE73], is owed to the much earlier work in scheduling theory and queueing applications by such people as Johnson, Conway, Maxwell, and Miller and numerous researchers in queueing theory. The analysis of information structures, the modeling of register assignment, and the analysis of storage requirements and access mechanisms, all engendered by the modeling perspective, are the legacies of operations research.

I also believe that operations research has contributed significantly to language concepts in computer science. For example, SIMULA 67 includes the class concept as a means of dealing with descriptions of members of a set. Developed within the context of solving simulation problems, the class and process concepts of SIMULA 67 are now acknowledged as major ideas by those dealing with abstract data types some eight years later.

THE SYNCRETIC REALIZATION*

The prevailing question in my mind is will the two disciplines realize the full potential of mutual benefits? On the positive side, I think this potential has been realized in one area: discrete event digital simulation. The joining of operations research and computer science in the language developments embodied in SIMULA 67 and SIMSCRIPT II.5 represent the best of both disciplines. The synchronization of processes, event scheduling algorithms, probabilistic modeling, model representation techniques, and emphasis on user/system interaction through report generation are examples of areas advanced by joint efforts and mutual contributions. Even allowing for my own biases, I believe that neither discipline could have realized such progress on its own. The amusing recollection of Kristen Nygaard and Ole-Johan Dahl [NYGAK78, p. 268] reminds us that all too often in the heat of intellectual conflict can the most lasting ideas be forged:

> In the spring of 1967 a new employee at the NCC in a very shocked voice told the switchboard operator: "Two men are fighting violently in front of the blackboard in the upstairs corridor. What shall we do?" The operator came out of her office, listened for a few seconds and then said: "Relax, it's only Dahl and Nygaard discussing SIMULA".

Scheduling theory represents another area of merging interests and mutual results. The recent special issue of Operations Research (January-February 1978) includes papers dealing with the complexity of scheduling problems [LENSJ78] and the scheduling of multiprocessor computing systems in real time [DHALS78]. The collection of papers edited by Coffman [COFFE76] represents a blending of the computer and job shop environments in the study of scheduling problems.

The approach to a syncretic state may be coming through the broader, important domain of algorithms. If so, I am even more encouraged at the prospect of realizing mutual benefits. To achieve this state, computer scientists must appreciate the need to develop algorithms, which embody efficiency and simplicity, and operations researchers must develop an appreciation of the technical problems of implementing such algorithms within the constraints of existing or envisioned machines. In my opinion, the "algorithm" forms a nucleus for both disciplines, and each has perceived this nucleus differently in the past. The computer scientist has been content to dwell on the creation of program representations of algorithms and the subsequent executions of these programs with the implications thereof. The operations researcher has limited his inquiry to the development of algorithms and the extraction of principles and concepts through that process, concerning himself only marginally with implementation questions. Each discipline must broaden its horizon to effect the desired level of syncretism.

A CONCLUDING DISCLAIMER

Currently, it seems quite popular, perhaps even proper, to begin every presentation with a disclaimer that what follows is the individual's opinions and does not reflect the opinion of the corporate, agency, or governmental employer. Being reluctant to do this, I have waited until the end to state my disclaimer:

> All the accurate assessments and good opinions included in this paper are those of the author and the author alone. He has pirated them from the best possible sources. Anything else is simply an accident of the man/machine interaction involved in creating this paper.

*Syncretism - The combination or reconciliation of differing beliefs or practices in religion, philosophy, etc. or an attempt to effect such compromise (Webster's New World Dictionary, 1970).

REFERENCES

ACKOR68 Ackoff, R.L. and M.W. Sasieni. Fundamentals of Operations Research, John Wiley: New York, 1969.

ARONJ69 Aronofsky, J. S. "Introduction," Progress in Operations Research, Volume III, John Wiley: New York, 1969, 3-15.

BHATU69 Bhat, U.N. "Sixty Years of Queueing Theory," Management Science, 15 (6): February 1969, B-280-B-294.

COFFE73 Coffman, Edward G. and Peter J. Denning. Operating Systems Theory, Prentice-Hall: Englewood Cliffs, 1973.

DENNP71 Denning, Peter J. "Operating Systems Principles and Undergraduate Computer Science Curricula," Technical Report TR-99, Department of Electrical Engineering, Princeton University, September 1971.

ENGEJ74 Engel, Joseph H. "The Philosopical Underpinnings of Computers and Operations Research," Computers and Operations Research, 1(1): March 1974, 3-7.

FORSG67 Forsythe, G.E. "A University's Educational Program in Computer Science," Communications of ACM, 10(1): January 1967, 3-11.

FORSG69 Forsythe, G.E. "Let's Not Discriminate Against Good Work in Design or Experimentation," Proceedings of AFIPS 1969 SJCC. 538-539.

HAMMR69 Hamming, R.W. "One Man's View of Computer Science," Journal of ACM, 16(1): January 1969, 3-12.

HEAND65 Heany, D.F. "Is TIMS Talking to Itself?" Management Science, 12(4): December 1965, B-146 - B-155.

KINGW67 King, W.R. "On the Nature and Form of Operations Research," Operations Research, 15(6): November-December 1967, 1177-1180.

KLEIJ73 Kleijnen, Jack P.C. "Computers and Management Science: Their Interface Today and Tomorrow, "Tilburg School of Business Administration, November 1973.

KLEIJ75 Kleijnen, Jack P.C. "The Interdependence of Operations Research and Computing: A Survey," Tilburg School of Business Administration, March 1975.

KLEIJ76 Kleijnen, Jack P.C. "Computers and Operations Research: A Survey," Katholieke Hogeschool Tilburg Reeks "Ten Discussie," February 1976.

KLEIJ77 Kleijnen, Jack P.C. "Operations Research and Computers," to appear in Encyclopedia of Computer Science and Technology, Jack Belzer, et.al. (ed.), Marcel Dekker, Inc.: New York, September 1977.

MJOSA72 Mjosund, A. "The Synergy of Operations Research and Computers," Operations Research, 20(5): September-October 1972, 1057-1064.

NANCR70 Nance, Richard E. "Operations Research and Computer Science - The Interface," Technical Report CP 70004, Computer Science/Operations Research Center, Southern Methodist University, March 1970.

NYGAK78 Nygaard, Kristen and Ole-Johan Dahl. "The Development of the SIMULA Languages," SIGPLAN Notices, 13(8): August 1978, 245-272.

PERLA67 Perlis, A.L. "University Education in Computing Science," Proceedings of the 1967 Stony Brook Conference, Academic Press: New York, 1968, 70.

ROSES69 Rosen, Saul. "Electronic Computers: A Historical Survey," Computing Surveys, 1(1): March 1969, 7-36.

STAHP67 Stahlknecht, P. "Quo Vadis, Operations Research?" Elektronische Datenverarbeitung (Electric Data Processing), 9: January 1967, 16-19.

WOOLG78 Woolsey, G. "Prolegomena on Promoting Practice or Pecksniff, Götz von Berlichingen, Symptoms & Reality," Interfaces, 8(3): May 1978, 7-12.