1981 Winter Simulation Conference Proceedings
T.I. Ören, C.M. Delfosse, C.M. Shub (Eds.)

APPLICATIONS OF MATHEMATICAL SYSTEM THEORY TO SYSTEM DESIGN, MODELLING AND SIMULATION

A. Wayne Wymore, Ph.D.
Professor of Systems &
Industrial Engineering
The University of Arizona
Tucson, Arizona, 85721

ABSTRACT

A mathematical system theoretic framework is described within which the necessary elaboration of the classical System Design, Modelling, and Simulation (SDMS) might be carried out. The applications of the new, elaborated problem-solving methodology to the design and analysis of communications systems is discussed as well as the concept of modelling as the design of an information system.

1. INTRODUCTION

The successes of system design, modelling and simulation (SDMS) are many and are not to be denied. Many of us, however, even in the face of these successes, feel uneasy. So often, it seems, successful projects are accomplished through the application of more or less standard, well-studied models: networks, linear programming, queueing, inventory, etc. The uneasiness about these successes comes from the almost sure knowledge that these standard, well-studied models are almost always very rough approximations of the real situation, and that many critical aspects of the real situation have been ignored. On the other hand, other successful projects seem to have been accomplished by unfettered model building: the development of models tailor-made to particular situations accompanied by computer programming projects and extensive and expensive simulation runs. The uneasiness about these successes comes from the almost sure knowledge that these tailor-made simulations almost always cost a great deal more than the information derived from them is worth; or, to put it more kindly, almost always it seems that what useful information was derived could have been derived at somewhat less cost.

Thus, SDMS appears to many outsiders as a bag of tools and the SDMS practitioner appears as a person carrying this bag of tools about the world looking for nuts and bolts that his particular set of wrenches will fit. He thus tends to state problems in terms of the tools in his kit bag.

On the other hand, SDMS seems many times, particularly to clients, as unfettered, and sometimes fiscally and perhaps even scientifically irresponsible, computer excercises. The practitioner frequently appears as a person with a compulsive and insatiable appetite for computer printouts and plots, determined to assuage his hunger at any price.

Such caricatures are, of course, unfair and somewhat misleading. But if we, as a profession, are to learn from our experiences, we must be very critical of our own performance and try to look at ourselves as others might see us.

To the extent that these exaggerations contain a grain of truth, it is not surprising that SDMS projects are subject to confusion from many sources: disastrous oversimplification of the statement of the problem, equally disastrous ambiguities in the statement of the problem, confusion of ends and means, unenlightened compromise of the statement of the ideal by practical considerations, confusion of the abstract and the concrete, and the worst source of confusion of all: stating the problem in terms of a preconceived solution.

Such lessons are difficult to assimilate because

81CH1709-5/81/0000-0209$00.75

there is no obvious alternative:

Either a standard model fits a situation reasonably well or else a tailor-made model must be developed to a level of detail at which the model fits a situation reasonably well, in the judgment of the people involved. What else can be done?

This challenge is met by a re-examination herein of the basic and classical (Note 1) SDMS problem-solving methodology:

1. State the problem.
2. Identify goals and objectives
3. Generate alternative solutions.
4. Develop a model.
5. Evaluate the alternatives.
6. Implement the results.

Usually, it is found that Steps 1 and 2 are done very sketchily and in narrative form, almost never comprehensively and precisely enough to guide effectively the activities of Steps 3 through 6. Because Steps 1 and 2 are accomplished neither comprehensively nor precisely enough, the gates are left open to every source of confusion enumerated above. The appropriate response to the new challenge appears to be, then, that each step in basic problem-solving methodology be accomplished within a single theoretical framework, comprehensively yet with mathematical precision.

Such a theoretical framework exists, called the tricotyledon theory of system design (*T3SD*, Note 2). This theory is applicable to the extent that problems in SDMS can all be thought of as system design problems. This includes practically every known problem; within *T3SD*, even systems analysis appears as a special case. Thus, *T3SD* could be used for every problem where it is not perfectly clear that a classical standard model is applicable. These latter cases ought, these days, to be very few and far between, previous successes of SDMS already having harvested the returns from what appears now to be, an extremely simplistic approach to extremely complex problems. There are no simple, small-scale problems left.

The principal conclusions of this paper are these: That a much greater proportion of the resources available to any SDMS project must be expended in stating the problem, that the problem be stated as comprehensively yet as precisely as possible within a system theoretical framework (such as the tricotyledon theory of system design), and that the process of modelling itself be regarded as the design of a large-scale, complex, man/machine, *information* system.

Each of the points touched on above is elaborated in this paper. Examples could be given in any field of application of SDMS. For the sake of a unified point of view, however, examples are offered exclusively from the field of communications systems. *T3SD* is described briefly and its applications to the design of communications systems and to the design of information systems are discussed. In these contexts, it is shown how the theory is a reasonable response to the challenge discussed above and how the various sources of confusion are avoided. In this way, the SDMS community can shuck the image of providing exact solutions to the wrong problems and begin to evolve real solutions to real problems.

2. THE PROBLEMS OF SDMS

In this section are elaborated the themes that SDMS utilizes blindly a bag of standard tools, or else engages in unfettered modelling, and that SDMS is inundated by confusion from many sources. References to the literature to support any but the most intuitively evident assertions are hard to find. For example, rarely is the total cost of an ADMS project reported in the literature and rarer still is the report of an attempt to assess the cost-effectiveness of an SDMS project. Thus, most of such assertions that appear in what follows must be classified as personal opinions based only on impressions or on personal experiences where even the circumstances of the experience must be disguised to avoid embarrassing the protagonists and to protect the author from the ire of the embarrassed protagonists.

Inherent in the technical educational system of the United States is the focus on the *solution* of problems. Almost every technical textbook has the following format: Here is a class of problems simply expressed. Here is the way to solve this class of problems. Here are some exercises to give the student practice in solving this type of problem. The student graduates from such an educational system with a sack full of techniques for solving simple problems of special classes. He learns to think of all problems in terms of possible solutions to the problem. Ask *any* reasonably-well-educated, technically educated person his opinion about *any* current problem--energy, communications, transportation, population, water, juvenile delinquency--and he will offer a *solution* to the problem. Never will he suggest that he or even we, as a society, do not *understand* the problem, do not *know* what the problem is. His education has taught him to think of problems in terms of the solutions to problems, not to find out precisely, comprehensively, what the problem is. The graduate of such an educational system wanders the world with his kit bag of techniques looking for problems that he can solve with his techniques and either not finding any, or finding them where aren't any.

The approach of the SDMS practitioner with his kit bag of classical techniques to the design of communications systems is typified by the attempt to characterize the communications system as a network of interdependent queues (Note 3) in the hope that classical queueing theory can be applied; or by the attempt to analyze a communications system as a phenomenon of flow through a network; or the attempt to apply linear programming to the allocation of channel capacity between preset nodes (Note 4). All of these approaches have yielded some useful results and information but each of them assumes the solution in some essential way or ignores some critical aspects of the problems of the design or analysis of the communications system.

The present state of the art of modelling is that any given phenomenon can be modelled to any level of detail and with any degree of accuracy (no matter how these terms are defined). But the price must be paid in actual dollars and time invested in the development of the model, its implementation, and its exercise.

Given any model or computer simulation, any scientist or engineer worth his salt can think of an aspect of the phenomenon being modeled that hasn't been included or can justify yet another level of detail in the model. There is no criteria given for halting the elaboration of the model and hence, in most cases, the elaboration of the model continues until the project has been exhausted of resources. Yet the classical SDMS methodology says, simply, "Develop a model."

I dare say there have been literally hundreds of digital simulations of communications systems developed over the last 20 years. Hard data to support this and the following assertions are difficult to come by but the impression remaining to me from 20 year's experience in, and observing of, scientific and engineering activity in the field of communications systems is that these simulations typically cost on the order of, at least, hundreds of thousands of dollars and have required on the order of years to develop and none can demonstrate any but little actual earnings (measured in any terms whatever) in real applications. See, for example, Note 5.

Among the sources from which confusion arises in almost every SDMS project are those that can be characterized by the names: over-simplification, ambiguities, ends versus means, ideal versus practical, abstract versus concrete, and stating the problem in terms of the solution. These are not collectively exhaustive nor mutually exclusive; they are sources from which confusion seems to have arisen in SDMS projects I have known.

Data recently collected on an extensive basis (note 6) indicate that the problem of the design or analysis of communications systems has been vastly oversimplified. Almost all such design or analysis projects have focused on the network through which a given message is transmitted. In the writer-to-reader data referred to, the total message delivery time is broken down into seven component times:

τ_1 is the time required for administrative handling of a message;

τ_2 is the time elapsed after the message is released to the time the telecommunications center accepts the message.

τ_3 is the telecommunications center in-station processing before transmission of the message;

τ_4 is the transmission time through the network;

τ_5 is the telecommunications center in-station processing after receipt of the message;

τ_6 is the time the message awaits pick-up;

τ_7 is the administrative handling time.

The overall averages of these times are:

$\bar{\tau}_1$ = 15.83 hours;

$\bar{\tau}_2$ = 5.84 hours;

$\bar{\tau}_3$ = 2.83 hours;

$\bar{\tau}_4$ = 12 minutes;

$\bar{\tau}_5$ = 1.64 hours;

$\bar{\tau}_6$ = 16.25 hours;

$\bar{\tau}_7$ = 7.2 hours.

Thus, it would appear that to the extent that message delivery time is a criterion for optimization in a problem of the design or analysis of a communications system, any project that focuses on τ_4, the transmission time through the network, has vastly oversimplified the problem.

To attack the other parts of the problem, however, is obviously very difficult. It is relatively easy to continue to reduce the transmission time τ_4 and even the in-station processing times τ_3 and τ_5, because that can be done by better hardware, with computers and software, but attack τ_1 τ_2, τ_6, and τ_7, involves the organization and its personnel, traditionally messy and difficult work.

But it is clear that these are the real problems in the desing of a communication system. There are not any significant, small-scale, simple problems left (if there ever were any).

Another source of confusion for the SDMS project is the existence of ambiguities in the problem statement. An example from the field of communications systems is the concept of reliability. It is often stated as a requirement for the design of a communications system that the reliability of the system must be 0.999, for example, yet never has a rigorous definition of the reliability of a communications *system* been given. Usually, the reliability of the system is interpreted to refer to the reliability of the hardware components. Yet communications systems are so vast and so complex that none of the series-parallel analysis of classical reliability theory are adequate to relate the performance of hardware components to the performance of the system in its basic function of delivering messages or enabling conversations. Thus, even the definition of a failure *mode* for the behavior of a communications system is lacking, or, at best ambiguous.

In communications systems the ends that are to be served are the deliveries of messages from one human being to another human being. Among the means available to perform that function are, for examples, tropospheric scatter, line of sight microwave, ground based or satellite based HF, hard wire or cable; and always the problem is stated in terms of bit rates, band-width, and channel capacities of the hardware. These considerations are but tenuously related to the message delivery capacity of the system. Clearly, the ends and means are confused.

Reconsider the concept of the reliability of a communications system to generate an example of the ideal versus practical source of confusion. Suppose that grade of service, speed of service, quality of service, and quantity of service have been rigorously defined as performance indices in system theoretic terms for any communications system model and that we have agreed upon a tradeoff composition of these performance indices to arrive at a general performance index reflecting the message delivery behavior of the system. These are idealizations and, ideally, the failure mode of the system can be defined in terms of a threshold value for this general performance index. Yet, practically speaking, it may be almost impossible to measure the value of this general performance index for any real system in operation. Hence, almost always the ideal never gets defined and discussed. But almost always, further downstream in the life cycle of the system or even in the systems desing process, the necessity to measure the message delivery performance of the system becomes evident, so various ad hoc sampling and instrumentation plans are advanced that are terribly expensive, ultimately ineffective, and finally, ignored. If the statement of the ideal had been included in and accepted as part of the problem statement at the outset, a practical plan of instrumentation and sampling could have been negotiated and designed to estimate the ideal values, as an integral part of the desing of the communication system.

It is frequently possible to discover many interesting aspects of the behavior of systems under consideration by assuming the idealization that we, as observers or systems, are omniscient and, at least intellectually, omnipotent with infinite computing capacity, so that, for example, we might consider comparing two proposed systems on the basis that we have available *all* the input/output data that could possibly be generated by each of the two systems over the entire lifetime of each system under every possible input trajectory and that we could perform instantly any computation that we wanted on this data. Many of the aspects of the behavior of systems discovered in this way must be considered to be idealizations but that does not make them less useful. It is only necessary to state, along with the idealization, how such an aspect will be assessed or approximated practically in the real world where we are not omniscient, nor ominpotent and have extremely limited computing power.

Related to, and overlapping somewhat, the ideal versus practical source of confusion is the abstract versus concrete source of confusion. In dealing with any aspect of the design of any system there are always two basic questions: "How is this aspect defined as a mathematical, system theoretic, entity?" and "How is the mathematical entity to be related to the real world?" These two basic questions must also be asked at two distinct stages in the system design process: "How is this aspect of the system to be represented as a mathematical entity and the mathematical entity to be related to the real world *during* the system desing process when a real system does not yet exist, when we are dealing with models, in some form, of *proposed* systems?" and, "How is the mathematical representation of this aspect to be related to the real, final system that is actually developed and deployed?"

Usually, we do not answer any of these questions, except one little part of the last: When the system is already designed and built, *then* we begin to think about testing the hardware components. Yet, I claim that the answers to all these questions must not only be supplied but are an *essential* part of the statement of the problem of the design of a system! And that when any answer is missing, the result is confusion and ineffective design. Both abstract definitions and their concrete realizations must be included in the statement of the problem. Thus, the concept of grade of service must be given as part of the statement of the problem of the design of a communications system, not only as an abstract mathematical definition but also as a concrete method for the assessment of grade of service for any model proposed during the design process and for the assessment of grade of service for the real, final system.

Many aspects of these sources of confusion seem to be special cases of one pervasive methodological error: stating the problem in terms of a preconceived solution. This practice takes two forms: either the assumption is made that the phenomenon under study is, say, a queueing problem and the problem is therefore stated as such; or, the assumption is made that the solution is in the form of some particular type of hardware and therefore the problem is stated in terms of this hardware.

The challenges clearly are: to try to develop solutions to the *right* problem in every project, to try to accomplish the necessary modelling and simulation in a cost/effective manner, and to try to avoid the various sources of confusion: oversimplification, ambiguity, ends versus means, ideal versus practical, abstract versus concrete, and stating the problem in terms of a preconceived solution either in the form of a model or in the form of hardware.

3. THE APPLICATION OF MATHEMATICAL SYSTEM THEORY

In this section, the classical SDMS methodology is discussed with a view to its expansion or elaboration in order to meet these challenges. A mathematical system theoretic framework is then described within which the necessary elaboration of the classical SDMS problem solving methodology might be carried out. The applications of the new, elaborated problem-solving methodology to the design and analysis of communications systems is discussed as well as the concept of modelling as the design of an information system.

The literature contains little discussion of SDMS problem solving methodology and what little discussion there is shows almost no change in more then twenty years from that enumerated by Churchman, Ackoff, and Arnoff in 1957. (Note 1):

1. State the problem.
2. Identify goals and objectives.
3. Generate alternative solutions.

4. Develop a model.
5. Evaluate the alternatives.
6. Implement the results.

It is clear that the difficulties discussed above arise in Steps 1, 2, and 4 in this methodology.

The problem must be stated in such a way to separate clearly the ends that the system is to serve from the means that the system *might* use to accomplish those ends. Included in the problem statement must be the idealizations of all concepts as well as practical means of their assessment in the real world and with respect to models of proposed system solutions. Each concept must be defined abstractly and with mathematical precision as well as concretely in real world terms.

It is difficult to visualize absolute goals and objectives in abstract terms without basing their enunciation on a preconceived solution to the problem. Therefore it would seem reasonable to replace Step 2: Identify the goals and objectives, with three other exhortations: (2.1) Identify all the criteria that could be applied to the comparison of any two proposed solutions with respect to how well comparatively the proposed systems behave to meet the stated ends the system exists to serve; (2.2) Identify all the criteria that could be applied to the comparison of any two proposed solutions with respect to the comparative utilization of resources represented by the means used by each proposed system to serve the stated ends; (2.3) Identify the criteria to tradeoff the criteria identified in Step 2.2 with the criteria identified in Step 2.3.

By the exhortation to identify *all* the criteria in Steps 2.1 and 2.2, we are motivated to avoid oversimplification. We are also faced with the necessity, however, to try to make precise many criteria that are not consciously but operationally ignored by being placed in the category: "too qualitative." As a working hypothesis we must adopt the point of view that there is no concept or criterion for the comparison of systems that is so qualitative, abstract, emotional, complicated or otherwise ephemeral, that it cannot be defined as a composition of other criteria that are themselves, if not numerically quantifiable and objectively measurable, at least mathematically symbolizable and practically assessable or observable or countable as some sort of sensory impression, as an idealization. We have found that even such concepts as "quality of life" can be so defined.

We are also faced with the necessity, however, of attempting to compose or to combine not necessarily commensurate quantities. One of the most promising approaches to the satisfaction of this necessity is to develop for each criterion to be combined a relative weight and a utility function (Note 7). Such trade-off artifacts must be "negotiated" between the SDMS team and its client.

In fact, all parts of the statement of the problem are negotiable between the SDMS team and the client, and the client, in particular, supplies the necessary value judgements under guidance by the SDMS team. Thus, the identification of the client is an extremely important, but unidentified, step in the problem solving methodology.

Steps 1 and 2, state the problem and identify the goals and objectives, must be accomplished so comprehensively and so precisely that reasonable limits are thereby placed on the activities of Step 4: Develop a model. From the results of STeps 1 and 2, it should be apparent what information precisely is required of any model. The function of the model in the classical methodology is to aid in the evaluation and comparison of alternative solutions based on information about the systems defined in Steps 1 and 2, and deduced from the model. The SDMS team and its client will then make a decision selecting the system evaluated as best among all those considered. In order that the model development Step 4 be made as cost/effective as possible, it is clear that Step 4 ought itself to be regarded as the design of an information system for the use of the SDMS team and client with its own ends/means statement and traded-off criteria.

All these concepts must be made mathematically precise. A framework within which such precision of definition might be achieved is discussed next.

The tricotyledon theory of system design (*T3SD*) is explicated in great detail elsewhere (Note 2), so only a brief sketch will be given here to make this paper somewhat self-contained. It will be difficult in a short sketch, however, to make convincing the argument that *T3SD* is an appropriate mathematical context for the elaborated SDMS problem solving methodology. But the attempt will be made.

T3SD is based on mathematical system theory where the system concept is axiomatized in precise mathematical set and function theoretic terms (Note 8). Specifically, a system is a 7-tuple of the form $Z = (S,P,F,T,z,Q,\zeta)$ where S is the set of states of the system; P is the set of individual inputs of the system; F is the set of time trajectories of inputs admissible by the system; T is the time scale of the system; z is the state transition function of the system such that if f is a time trajectory of inputs of the system Z, $f \in F$, x is the state of the system Z at time 0, $x \in S$, and t is a time value in the time scale T of the system Z, $t \in T$, then $z(f,x,t)$ is the symbol for the state of the system Z at time t given that the state of the system Z at time zero is x and that f is the time trajectory supplying inputs to the system Z between times 0 and t; Q is the set of individual outputs and ζ is the readout function such that if x is the state of the system at any time, then $\zeta(x)$ is the output of the system at the time. The mathematical artifacts S, P, F, T. z, Q, and ζ must also satisfy certain consistency requirements not important to us at this level of discussion.

On the basis of this definition of system are defined various related concepts and relationships between system models, and manipulations of the system concept: input port, output port, system experiment, performance index, subsystem, homomorphism, isomorphism, simulation, coupling recipe, and the system resultant of a coupling recipe. The coupling recipe is the description in mathematical and system theoretical terms of the familiar block diagram cartoon of system components related by arrows indicating input/output relationships among the components. The result of a coupling recipe is the *system* whose components are given in the block diagram. The system properties of the resultant are deducible within the theory.

On the basis of such a definition of the system concept a system design or analysis problem is defined: A system desing or analysis problem is a 6-tuple $P = (X, T, \alpha, \beta, \gamma, \nu)$ where:

- X is an input/output specification;
- T is a technology;
- α is a merit ordering over the I/O cotyledon generated by X;
- β is a merit ordering over the technology cotyledon generated by T;
- γ is a merit ordering over the feasibility cotyledon generated by X and T that is a tradeoff between α and β;
- ν is a system test plan.

An input/output specification is a 5-tuple of the form: $X = (V, X, G, Y, H, \eta)$ where X is the set of individual inputs which the system to be designed or analyzed exists to manage; G is the set of all time trajectories of such inputs the system to be designed or analyzed might have to manage; Y is the set of individual outputs which the system to be designed or analyzed exists to produce as a result of its management of the inputs; H is the set of all time trajectories of outputs the system to be designed or analyzed might generate; η is a function that to each given input trajectory mtaches the set of all output trajectories that could be produced as output by any system experiencing the given trajectory as input restricted only by physical, biological, or legal limitations. The I/O specification X is the operational statement of the ends the system exists to serve.

A technology T is any set of systems that represents the components available to be organized by means of a coupling recipe into a resultant system that might solve the problem. The technology T represents the means available to serve the ends.

The I/O cotyledon generated by an I/O specification X is the set of all systems that "satisfy" the I/O specification. A system Z satisfies X if Z can be started in some state y such that if Z experiences any given input trajectory in G then Z will produce as output trajectory one of those in H matched with the given input trajectory by η. The input/output specification itself be defined in such a way that the I/O cotyledon will be as "large" as possible by suspending our judgements of "good" and "bad" I/O performance during the process of defining the I/O specification. But when the I/O specification X has been defined and the I/O cotyledon thereby determined, then the I/O merit ordering α is defined in such a way to compare any two systems in the I/O cotyledon with respect to their I/O performances.

The technology cotyledon is the set of all systems that are buildable in the technology T. A system Z is buildable in the technology T if Z is the resultant of a coupling recipe all of whose components are in T. The merit ordering β is defined over the technology cotyledon in such a way to compare any two systems in the technology cotyledon with respect to the utilization of resources (U/R) represented in their construction and operation.

The feasibility cotyledon is the set of all systems that are in the I/O cotyledon and are implementable in the technology. A system Z is in the feasibility cotyledon if it satisfies the I/O specification X and is implementable in the technology T in the sense that there exists a system Z^r buildable in T that simulates Z, that is, there is a subsystem Z^s of Z^r of which Z is a homomorphic image. The merit ordering γ is defined over the feasibility cotyledon in such a way to compare any two systems in the feasibility cotyledon with respect to a tradeoff (T/O) between I/O performance of the systems and the U/R of the systems represented in their implementations.

The ideal output of the system desing or analysis *process* itself is the selection of a system Z from the feasibility cotyledon that is optimal with respect to the tradeoff merit ordering γ, together with the implementation of Z in the technology T represented by a coupling recipe κ, all the components of which are in the technology T, and its system resultant Z^r. From such a coupling recipe a real system Z^{real} will be brought into being at the implementation Step 6 of the classical SDMS problem-solving methodology. It is here asserted that an essential part of the *statement* of the *problem* of the desing or analysis of any system is the way in which that system Z^{real} will be tested. The system test plan ν must provide the methodology, as part of the problem statement P, for answering the questions: What are the actual levels of I/O performance and U/R of Z^{real}? Is Z^r an adequate model of Z^{real}? Is Z^{real} acceptable? The system test plan ν is thus the practical, concrete definitions in real-world terms of the abstractions and idealizations involved in the other problem statement artifacts X, T, α, β, and γ.

It is herewith recommended that this system theoretic structure be used in the basic SDMS problem-solving methodology in order to achieve a truly comprehensive statement of the problem, precisely stated and to avoid the sources of confusion discussed previously, and, in particular, to avoid stating the problem in terms of a preconceived solution or class of solutions.

4. APPLICATION TO COMMUNICATIONS SYSTEMS

The way in which *T3SD* might the applied to the problem of design of a communications system is discussed in this section. Again, the discussion is necessarily terse and the unpersuaded skeptic will have to await future publications of the details elsewhere (Note 9).

The first step in the statement of the problem of the desing or analysis of a communications system is to define the I/O specification and this means to define the set of inputs and the set of outputs, the time trajectories of inputs and outputs and the matching according to limitations of one kind or another.

Intuitively, an input to a communication system is a message that one user would like to send or tell to another user. Hence at any instant of time, the input that a communications system must be desinged to handle is the set of all messages that any user would like to transmit to any other user. Such an input could be represented ideally by a matrix that has a row *and* a column for each user and whose (i,j) element is the set of messages that user i would like to transmit to user j. Any such matrix will be called a statement of demand for communication. The precise definition of such inputs requires the definition of the set of users and the definition of the set of all possible messages. The set of messages might consist only of the bits, 0 and 1, or the set of all possible finite strings from the English alphabet.

The instantaneous output of any communications system can be characterized intuitively as the set of messages delivered to each user by the communications system. Such an output could be represented ideally by a vector with a component for each user, the ith component being the set of messages delivered to the ith user.

There seems to be no reason for restricting either the set of input trajectories or the set of output trajectories. Nor does there seem to be any basic physical laws (such as conservation of mass or energy) that would prevent the generation of *any* input trajectory. Of course, we would expect that the output trajectory of delivered messages ought to be strongly related to the trajectory of inputted messages; this is not, however, a physical necessity but a criterion for good input/output performance. Such criteria will be considered in due course.

The technology components available to be organized into a system that might be the optimal solution to the design or analysis problem under consideration include all the standard communications hardware.

The next three steps are to develop: the I/O merit ordering α over the I/O cotyledon, the U/R merit ordering β over the technology cotyledon, and the T/O merit ordering γ over the feasibility cotyledon.

The I/O merit ordering is developed in several stages beginning with the definition of performance indices, continuing with the development of a probability distribution over the set of system experiments, defining figures of merit as expected valued, and then "rolling up" or composing the figures of merit into a single, overall, I/O figure of merit in terms of which the ordering α is defined.

A system experiment for a given system Z is determined by an input trajectory f, an initial state x, and a time value t. The system experiment is represented by the triple (f,x,t) and interpreted to mean that the system Z is started in the state x, fed the input trajectory f, and the behavior of Z observed until time t. The state trajectory and output trajectory of Z are determined by the system experiment (f,x,t).

A performance index for a system Z is any function defined over the set of all system experiments of Z. Performance indices are usually defined as averages of output or estimates of probabilities based on output trajectories, maxima or minima of output trajectories, and so forth.

If a system Z satisfies the I/O specification X described above, then Z accepts as input, statements of demand for communication and produces as otuput, deliveries of messages. In terms of such "data" produced furing any system experiment on such a system Z, no less than 23 I/O performance indices are rigorously defined in system theoretic terms representing the following concepts:

1. Grade of service, an estimate of the probability that a message entered as input will be delivered.
2. Average delivery time.
3. Standard deviation of delivery time.
4. Maximum of delivery times.
5. Average rate of information transfer (in terms of the measure of information content defined over the set M of messages).
6. Speed of new user installation.
7. Average transmission quality.
8. Spurious call probability.
9. Interrupt rate.
10. Rate of message delivery.
11. Rate of information transmission.
12. System information capacity (in information theoretic terms).
13. Estimated ility (in terms of a composite performance index of all those above, 1-12, and a threshold value defining the "distinguished condition" or I/O failure mode of the system).
14. Average time before first entry into distinguished condition.
15. Average time between entries into distinguished condition.
16. Empirical distribution of times in distinguished condition.
17. Transient state stability.
18. Steady state stability.
19. Asymptotic stability.
20. I/O stability.

21. Improvability.
22. Interoperability.
23. Transitionability.

The next step is to convert these performance indices into figures of merit as expected values of the performance indices over the set of all system experiments of each system in the I/O cotyledon. This requires the development of a probability distribution over the set of system experiments of any system in the I/O cotyledon. Such a probability distribution can be developed on several grounds. It can be assumed that the probability distribution is purely atomic, that is, that only a finite number of system experiments carry a-l the nonzero probability. This assumption is valid because we can only compute the results of a finite number of system experiments for any system model. If the users of the system are themselves represented by system theoretic models with a class of inputs generated independetly of the communication system, then the desired probability distribution can be deduced in terms of such user models and a probability distribution over the independently generated inputs. There are several ways that historical data or user surveys can be used to generate the required probability distribution.

Assume then, that the 23 performance indices have been concerted into 23 corresponding figures of merit. The next step is to "roll up" or combine these 23 into a single figure of merit. This is accomplished in two stages, a primary roll-up and a secondary roll-up. The reason for this is to try to keep the task of supplying the value judgements necessary to the roll-up process as meaningful as possible to the SDMS team and the client, and this means that the individual tasks must be kept small enough to involve no more than 7 ± 2 (Note 10) concepts simultaneously.

The primary roll-up consists in:

1. bypassing grade of service until the secondary roll-up because of its historical importance in the design of communications systems,
2. combining figures of merit 2-6 into a general speed-of-service figure of merit,
3. combining figures of merit 7-9 into a general quality-of-service figure of merti,
4. combining figures of merit 10-12 into a general quantity-of-service figure of merit,
5. combining figures of merit 13-16 into a general RAM figure of merit,
6. combining figures of merit 17-23 into a general stability figure of merit.

The secondary roll-up consists in combining the six general figures of merit defined by the primary roll-up into a single figure of merit ξ^α for overall I/O performance. Then the merit ordering α is defined in terms of this figure of merit: if Z^1 and Z^2 are two systems in the I/O cotyledon, then $Z^1 \underset{\alpha}{\leq} Z^2$ if and only if $\xi^\alpha (Z^1) \leq \xi^\alpha (Z^2)$.

The roll-up procedures can themselves be defined in terms of value judgment artifacts: baseline values, weights, and utility functions. If $\xi^1, \ldots, \xi^n$ are figures of merit to be rolled up, then we first identify a baseline value b^i for each figure of merit ξ^i that represents the corresponding value for the present system or the "usual" value of such a figure of merit. Then to each figure of merit ξ^i we assign a weight w^i such that $\Sigma w^i = 1$. Then for each figure of merit ξ^i we define a utility function u^i that maps the range of ξ^i into the interval of real numbers from 0 to 1, inclusive such that $u^i(b^i) = 0.5$. If Z is any system for which $\xi^i(Z)$ is defined for each i, then the value of the rolled up figure of merit ξ for Z is defined as follows: $\xi(Z) = \Sigma\{w^i \times u^i(\xi^i (Z)): i \in I [1,n]\}$. Various stories and games and strategies are available to be employed to make meaningful and consistent the negotiation between the SDMS team and the client of the baseline values, weights, and utility functions (Note 7).

Using similar techniques, the technology merit ordering β is defined. First, 13 U/R figures of merit are defined for any system Z buildable in the technology *T*. These are:

1. Capital investment.
2. Operating and maintenance cost.
3. Total cost.
4. Total cost per user.
5. Total cost per message.
6. Total energy consumption.
7. Numbers of personnel required.
8. MTBF.
9. MTTR.
10. Component reliability.
11. Replacement costs.
12. Development time.
13. Probability of success.

Next comes a primary roll-up:

1. Figures of merit 1-5 are combined into a general cost figure of merit.
2. Figures of merit 6, 7 are combined into a general natural resource figure of merit.
3. Figures of merit 8-11 are combined into a general RAM (U/R) figure of merit.
4. Figures of merit 12-14 are combined into a general system development figure of merit.

The secondary roll-up consists in combining the four figures of merit defined by the primary roll-up into a single figure of merit ξ^β representing the level of U/R of any system buildable in the technology.

The roll-ups of the U/R figures of merit can be accomplished in exactly the same way as the roll-ups of the I/O figures of merit, in terms of the value judgment artifacts: baseline values, weights, and utility functions.

The technology merit ordering β is defined in terms of the overall U/R figure of merit ξ^β exactly as was α defined in terms of the overall I/O figure of merit ξ^α. The definitions of α and β can be accomplished in such a way to incorporate threshold values of various figures of merit representing minimally acceptable values or inflexible requirements.

The T/O merit ordering γ can also be defined in terms of a general T/O figure of merit ξ^γ where ξ^γ can be as simple as the sum $\xi^\alpha + \xi^\beta$ of the I/O and U/R figures of merit (representing a generalized profit figure of merit), or the product $\xi^\alpha \times \xi^\beta$ of the I/O and U/R figures of merit (representing a generalized benefit/cost ratio), or as complicated as any "increasing" function of ξ^α and ξ^β can be. Additional T/O criteria might be introduced into the definition of γ only cautiously because the transitivity of γ or the T/O nature of γ might be easily destroyed.

Now it is necessary to reconsider every aspect of the problem statement developed to this point. The objective of the development to this point has been to define each artifact precisely in mathematical, system theoretic terms, in abstract, ideal terms. In order to define a test plan each one of these artifacts must now be defined in operational, real-world terms, in terms of instrumentation, observing methodologies, sampling plans, statistical estimating procedures, and so forth.

When we have accomplished this last step, then and only then, can we consider that our problem of the design or analysis of a communications system is well stated. Now we are ready to consider modlling of communications systems. We might have to develop models of components in the technology *T* because the output of our system design or analysis effort must be a model (a set of blueprints) from which the real communications system can be built or developed. Our problem statement exercise tells us what information we need from our models and hence delimits the level of detail somewhat. But hos can we be assured that our modelling efforts are truly cost/effective?

5. MODELLING AS THE DESIGN OF AN INFORMATION SYSTEM

The only way that we can be assured that our efforts at modelling communications systems are cost/effective is that we approach the problem of modelling as a problem in the design of an information system stated as comprehensively yet as precisely as possible in system theoretic terms. I, of course, recommend *T3SD* for this purpose. It should be pointed out that another formal methodology based on mathematical system theory is extant (See Note 11).

The users of the information system to be designed are the members of the SDMS team and, perhaps, of the client group. These users will generate queries or questions as input and will receive responses to their queries. Some queries might be generated typically as follows:

The SDMS team or the client propose a favorite solution Z to the problem and ask of the information system: Does the system Z satisfy the I/O specification *X*? Is the system Z implementable in the technology *T*? How much better is the system Z than the baseline system with respect to each of α, β, and γ? What is the value of the performance index ϕ for the system Z operating under the system experiment (f,x,t)? What is the expected value of the performance index ϕ for the system Z?

When the SDMS team and the client group has split into n factions, and the ith faction proposes its favorite solution Z^i, then the information system will be asked to compare the n systems with respect to the baseline system, each individual performance index, figure of merit, and merit ordering.

Perhaps, at some stage the SDMS team will be able to define a system design concept: a set of systems parametrizable by a small number of real variables. Then the information system may be asked to find the system in the given set optimal with respect to ξ^γ, the overall T/O figure of merit.

Perhaps initially, or at some stage, the SDMS team and client group will be completely stumped and will ask the information system to produce one, or two, ... systems in the I/O cotyledon, the technology cotyledon, or the feasibility cotyledon.

Perhaps the SDMS team will ask the information system for a comparison of several systems based on a different set of weights and utility functions for sensitivity studies.

In any event, the design of the information system incorporating the mo-els necessary for the design of the communications system will begin with the definition of the I/O specification for the information system, and the definition of the I/O specification will begin with the enumeration of queries that will be generated as input to the information system by the SMDS team and the client group.

The statement of the problem of the design of the information system continues with consideration of the technology available to build the information system. This technology will ordinarily be limited to the in-house capability of the SDMS team and of the clients groups themselves. The technology includes in a sense, also, the mathematical and computer techniques available to generate answers to the queries.

The I/O merit ordering for the design of the information system will undoubtedly be based on such concepts as: speed of response, accuracy of response, completeness of response, confidence in the responses, number of queries ignored, and so forth. These concepts can be given explicit and precise system theoretic definitions as well as operational, real-world definitions as part of the system test plan.

The technology merit ordering will be based on such concepts as: capital investment, operating and maintenance costs, total life cycle cost, cost per query, number of personnel required, personnel costs, computer costs in both time and money, reliability and maintainability, time to develop, probability of successful development, and so forth. These concepts can also be given explicit and precise system theoretic definitions as well as operational, real-world definitions as part of the system test plan.

Then the T/O between I/O and U/R must be defined as well as the system test plan.

It would be desirable if all the artifacts required for the complete statement of the problem of the design of the information system for the use of the SDMS team and the client in the solution of the problem of the design or analysis of a communications system could be written down and defined as precisely as has been done for the communications system problem itself. Perhaps as the output of a research project such an expression could be achieved in general enough terms to apply to many modelling situations yet specific enough to be useful in any particular one. Until such research is completed, however, in all but the most well-heeled projects, it is clear that "back of the envelope" kind of considerations must be utilized for the design of the information system. But just because the computations themselves are not precise does not mean that any consideration required by *T3SD* can be overlooked. *T3SD* ought to be harnessed into even the "back of the envelope" mode of consideration.

When the statement of the problem of the design of the information system has been achieved, then its design and development can proceed apace to arrive at models and their associated computer implementations necessary in the process of the design or analysis of the communications system. The design and development of the information system and the models of communications systems on wihch it is based are cost/effective within the range of considerations by the SDMS team and the client group because the problem of the design and development of the information system was stated first, comprehensively and as precisely as possible (within available resources) as an optimization problem where the objective to be maximized is a T/O between generalized effectiveness and generalized cost.

6. CONCLUSIONS AND RECOMMENDATIONS

It must appear that in order to adopt the elaborated problem-solving methodology described herein, the cost of any SDMS project might have to be inflated considerably. I assert that this is not so. I think that more of the resources available to any SDMS project must be allocated to the problem statement exercise, but I am sure that these resources can be more than recovered in savings further downstream in the avoidance of confusion and in the achievement of much better system designs and analyses much more quickly.

Another advantage of the methodology described herein, is the provwsion of a clearcut occasion and an arena for political activity and formation of policy and value judgments that become part of the statement of the problem. Everyone in the SDMS team and the client group has a chance to incorporate his pet prejudices into the statement of the problem. At some point, however, everyone must agree that the problem statement exercise is concluded (except for iterations that might be desirable later on technical grounds), and everyone signs off on the problem statement. From that point on, the project can become more purely scientific and technical without political interventions. In most SDMS projects, a source of confusion is the application of political considerations to what ought to be purely scientific or technical questions, throughout the project. The use of the elaborated problem-solving methodology would tend to minimize this source of confusion as well as the others discussed in previous sections.

My second basic recommendation after that of allocating more of a project's resources to the statement of the problem exercise, is that the problem statement be expressed in a precise, methematical, system theoretic, framework.

It seems that every project in large-scale, complex, man/machine system design or analysis starts from scratch; we seem to learn very little from previous experience and the wheel must always be reinvented. Part of the reason for this necessity in many large-scale, complex, man/machine system design or analysis projects lies in the lack of a mathematical theory within which design or analysis decisions can be made as in possible in system design or analysis projects where such theories (aerodynamics, electrical circuits) exist. A formal theory of system design can make up for this lack by providing a rogorous framework for the documentation of all decisions pertinent to a system design or analysis project. Future generations of system designers or analysts can then be informed as to the precise reasons underlying each decision and can therefore proceed to improve on the previous generations' work.

My final recommendation is that every modelling problem must be treated as the design of an information system. If not, there is no way of limiting the modelling activity, and, as usually happens, the modelling and computer simulation activity consumes all the resources available to the project without guarantee that any useful information will be provided in a timely way.

NOTES AND REFERENCES

1. The formulation given seems to cover most of the SDMS problem solving methodologies proposed and is roughly a combination of those to be found in the following two early references:

   Churchman, C., R. Ackoff, and E. Arnoff (1957), *Introduction to Operations Research*, John Wiley, New York.

   Hall, A.D. (1962), *A Methodology for Systems Engineering*, Van Nostrand, Princeton, New York.

2. Wymore, A.W. (1976), *Systems Engineering Methodology for Interdisciplinary Teams*, John Wiley, New York.

3. Driscoll, M. and N. Weiss (1976), An Application of Queueing Theory to Reservation Networks, *Management Science*, 22, pp. 540-546.

   Gopinath, B. and J. Morrison (1977), Single Server Queues with Correlated Input, *The Bell System Technical Journal*, Vol. 56, No. 9.

   Spragins, J (1977), Simple Derivation of Queueing Formulas for Loop Systems, *IEEE Transactions on Communications*, Vol., COM-25, No. 4, pp. 446-448.

4. See, for example:

   Gomory, R. and T. Hu (1964), Synthesis of a Communication Network, *J. SIAM*, 12, 2, pp. 348-369.

5. Reitman, J. (1971), *Computer Simulation Applications*, John Wiley, New York, pp. 279-296.

   Seaman, P. (1966), On Teleprocessing System Design: The role of Digital Simulation, *IBM Systems Journal*, pp. 175-189.

6. Dunn, T.S. (1977), *U.S. Army Telecommunications Management Data Base (Writer-to-Reader), Vol. V, System Summary and Projected Writer-to-Reader Message Times and Pre-AMME Writer-to-Reader Data*, Systems and Economic Analysis Division, Office of the Comptroller, Headquarters, US Army Communications Command, Fort Huachuca, Arizona, 85613.

7. Keeney, R. and H. Raiffa (1976), *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*, John Wiley, New York. See also the reference in Note 2.

8. Wymore, A.W. (1967), *A Mathematical Theory of Systems Engineering: The Elements*, John Wiley, New York.

   Wymore, A.W. (1969), Discrete Systems and Continuous Systems. In: *Advances in Mathematical System Theory*, P.C. Hammer (Ed.), Pennsylvania State University Press, University Park, Pennsylvania.

   Wymore, A.W. (1972), A Wattled Theory of Systems. In: *Trends in General System Theory*, G.J. Klir (Ed.), John Wiley, New York.

9. Wymore, A.W., *A Theory of the Design of Communications Systems: An Application of the Tricotyledon Theory of System Design*. (The book is being written).

10. Miller, G.A (1956), The Magical Number 7 + 2 Some Limits on Our Capacity for Processing Information, *Psychological Review*, 63, pp. 81-97.

11. Zeigler, B.P. (1976), *Theory of Modelling and Simulation*, John Wiley, New York.