NORSK REGNESENTRAL

NORWEGIAN COMPUTING CENTER

USER'S EXPERIENCE WITH THE SIMULA LANGUAGE

by

Haavard Hegna, Ole Johan Lund
and
Kristen Nygaard

106

NORWEGIAN COMPUTING CENTER
FORSKNINGSVEIEN 1B
OSLO 3 - NORWAY
JUNE 1968

USER'S EXPERIENCE WITH THE SIMULA LANGUAGE

by

Haavard Hegna, Ole Johan Lund
and
Kristen Nygaard

PREFACE

The SIMULA I language and its compiler for the UNIVAC 1107 and 1108 computers have been available since the beginning of 1965. The system has been used in a large number of jobs on a wide range of phenomena.

This report shall serve a double purpose: to give a survey of some jobs analyzed by the aid of SIMULA and to give our experience with the compiler and the teaching of the language. In addition some features of the new SIMULA 67 language is presented.

The job survey is very incomplete since we are not regularly informed about SIMULA work at other computer sites. Also, the information on the jobs reported is often very sketchy.

Haavard Hegna   Ole Johan Lund   Kristen Nygaard

TABLE OF CONTENTS PAGE

		PAGE
1.	INTRODUCTION	1
2.	TRAINING IN THE USE OF SIMULA I	2
3.	DEBUGGING AND TESTING SIMULA I PROGRAMS	3
4.	DATA ANALYSIS AND PRINTOUT	5
5.	COMPILATION AND EXECUTION SPEEDS	6
6.	THE SIMULA 67 LANGUAGE	9
7.	PROBLEMS BEING SOLVED BY SIMULA I USERS	11
7.1	Production	12
7.2	Administrative Problems	14
7.3	Inventory and Warehousing	15
7.4	Transportation	15
7.5	Harbours and Shipping	17
7.6	Planning and Scheduling	18
7.7	Renewal	19
7.8	Computer Operation and Monitor Design	19
7.9	Computer Hardware Design - Debugging	20
7.10	Social Systems	21
7.11	Statistics	22
7.12	Communications	23
7.13	Biological Systems	23
7.14	Associative Networks	24
7.15	Market Research	25
7.16	Railway Marshalling Yards	26
7.17	Evaluation of Production Planning Schemes	27
8.	References	27

1. INTRODUCTION

SIMULA I is an ALGOL 60 based simulation language, developed by Ole-Johan Dahl and Kristen Nygaard at the Norwegian Computing Center 1961 - 1964. A SIMULA I compiler has been working on the UNIVAC 1107 since January 1965.

A SIMULA I manual (1) appeared in May 1965. It has reached its fifth printing and has been spread in about 3000 copies. A Russian version of the manual appeared in 1967. The compiler is released for distribution by UNIVAC. SIMULA I is used at a number of UNIVAC 1107 and 1108 installations. The compiler has now worked well for more than three years, and no difficulties have been experienced in transferring it to other UNIVAC 1107 or 1108 configurations.

The authors of SIMULA I and Bjørn Myhrhaug developed a new general programming language named SIMULA 67 (3). This language, now being implemented on various computers, is a major extension of the SIMULA I referred to in this paper. SIMULA 67 has also extended power as a simulation language over the present SIMULA I.

This paper will discuss the experience gained by the use of SIMULA I. It presents some of the problems solved by SIMULA I users and also the current work on SIMULA 67. The paper is an updated and extended version of the earlier "Report on the Use of SIMULA up to December 1965". (7).

2. TRAINING IN THE USE OF SIMULA I

Since the first printing of the SIMULA I manual about 12 one week courses with ten to thirty pupils have been given at NCC. In addition, one week courses have been given in Zurich, Stockholm, Moscow, Novosibirsk, Leningrad and St. Paul. A large number of shorter presentations have been given in Norway and abroad.

SIMULA I is taught at the Royal Institute of Technology in Stockholm, and also at the Norwegian Institute of Technology. It is treated in several courses on Computer Science both in Europe and in the United States.

The pattern of the one-week courses given by NCC personnel has now been well established. The participants normally have a very varied background. Some are new to simulation, others have only some knowledge of ALGOL and programming.

The first day is used to give an intuitive understanding of the simulation technique and the SIMULA I approach to system description. The next two days are used for a precise definition of the basic concepts and the most important language statements and procedures. Through exercises in the classroom, an understanding of the event sequencing and referencing facilities is usually obtained.

At the end of the third day, the first exercise in system description is given. The fourth and the fifth day are used for an explanation of the parts of the manual which have not yet been discussed, as well as for presentation of a number of jobs having been solved by SIMULA I. It is important through these presentations to give the participants an understanding of how a simulation language should be used as a research tool in various situations.

Our experience is that the participants after this course should go with a medium sized system description (150-250) program cards), but that most of them will need guidance from experienced users of the language. This "first job" is offered as part of the course: NCC will suggest suitable jobs, aid the participants and provide free debugging and some production runs of the computer. NCC also offers an additional one-week course giving more intensive training.

During their first job, newcomers usually make all the standard blunders, learn to avoid them and are then rather well prepared to continue on their own afterwards.

3. DEBUGGING AND TESTING SIMULA I PROGRAMS

Debugging of a simulation program may be divided into two phases:

1. Obtaining a program that is syntactically correct and contains no routine errors.

2. Modifying the program to obtain a correct description of the system under study.

Simulation programs are generally considered difficult to write, whether the system description is correct or not. We have, however, observed that people using the SIMULA I language only seldom seem to need any assistance from

NCC specialists once their first job has been successfully completed. Establishing a syntactically correct program seems to require only few test runs, even for complex programs (e.g. 3 - 6 test runs for programs consisting of 400 - 600 cards).

The compiler is presently built around an already existing one-pass ALGOL compiler. The need for more comprehensive error messages in this debugging phase is manifest, and may be included in future implementations.

The analysis and modification of the syntactically correct program is the second and often more difficult stage in developing a valid simulation model. The standard approach to this phase is to run the program with sufficient printouts to get a clear picture of what is happening during executions of the program, i.e. trace the sequence of actions in the model, within as well as between events. In the SIMULA Tracing System,(2), this tracing is to a large extent automatic, and may at the same time be controlled by a number of system procedures in order to obtain the most useful printout.

The Tracing System operates in three modes:

1. Full trace: Trace information will be printed whenever a traceable action occurs.

2. Selective trace: Trace information will be printed only when the traceable action involves a process that has been "marked". Marking and unmarking of processes are under program control.

3. No trace: All tracing is suppressed.

Use of the SIMULA Tracing System requires more memory space and computer time and are normally used only in the debugging phase.

If the system is well defined, 2 - 3 test runs may be sufficient in the second debugging phase.

One of the main reasons why the debugging seems to be rather fast is the fact that the SIMULA I syntax and run-time system makes it impossible to refer to data which have disappeared or to a data structure under the assumption that it belongs to another class than expected.

For a number of research workers, the learning of a simulation language also implies the learning of an algorithmic language. We feel that the ease with which SIMULA seems to be picked up to a large extent is due to ALGOL. Our experience is that ALGOL is very easy to learn and use, contrary to beliefs commonly held in some countries.

4. DATA ANALYSIS AND PRINTOUT

A simulation language should provide extensive facilities both for generating data and for collecting statistics during simulation. Facilities for displaying the results should also be available. SIMULA I contains presently eleven procedures for generating random variables from most of the common probability distribution, such as the

Uniform, Discrete, Normal, Poisson, Erlang and Exponential distribution, as well as distribution based on observed data.

For data analysis, there are three procedures: "HISTO", which will update an array containing a histogram, "ACCUM", which computes time-integrals, and "ARRINFO", which computes various statistical measures from a two-dimensional set of observations. Extensions have now been postponed in the advent of the SIMULA 67 compiler.

SIMULA I has the ordinary UNIVAC 1107 ALGOL-printout facilities available. A completely new set of input/ output procedures has been defined in SIMULA 67, allowing easier editing and generation of reports.

The display facilities consist of a routine for direct histogram-printing and a set of procedures to produce paper tapes from SIMULA I acceptable for off-line drawing by the KINGMATIC 1215 Automatic Drawing Machine (produced by Kongsberg Våpenfabrikk, Kongsberg, Norway) (4). These include flexible routines for curveplots, histograms andoother diagrams.

5. COMPILATION AND EXECUTION SPEEDS

In this section a few typical jobs will be described and their average computation and execution times given. More information will be found in section 7.

In a plant layout analysis, the factory consisted of 22 machines groups (1 - 9 similar machines in parallel in total more than 60 machines) with batch-processing of 17 different order types (each with its own defined or preferred path through the factory). Components for the same order entered the system at two separate inlets being matched later on. The machines were served by 50 operators, grouped in 8 classes according to combinations of skills. The number of program cards was 340, compiling time was 28 seconds, execution time for 10 weeks' simulation 390 seconds.

The largest SIMULA program run at the NCC 32 K computer is an analysis of a new job foundry for a Norwegian firm. The object was to find the proper arrangement and dimensioning of cranes, storage points, etc. The number of program cards was 1130, the compile time about 50 seconds. Execution time for simulation of 2.5 days' operation was approximately 22 seconds (250 products and approximately 200 processes giving service to the products.)

SIMULA I programs are run under control of the EXEC II Monitor at the UNIVAC 1107 and 1108 Computers (4 micro-seconds and 0,75 microseconds memory cycle time respectively), Programs are compiled directly into object code. In the total time used on the machine, the standard monitor operations in handling a SIMULA I program will be included. SIMULA I will also be available with the EXEC 8.

The SIMULA I Compiler is at NCC located on Magnetic Drum, at other installations on magnetic tape. The extension to the ALGOL Compiler is approximately 1000 instructions. The SIMULA I Library with the Run-Time System and System Routines is presently located on a magnetic tape which has to be mounted when a SIMULA I program is to be run. The size of the Library is presently approximately 4000 instructions. The time used in reading the tape is approximately 5 seconds (with 90 Kc/s tapes).

The compilation speeds observed lie between 12 and 22.5 program cards per second for the jobs on which data are available. The average is 16.8 cards per second. This is the same speed as that of our current UNIVAC ALGOL Compiler.

The execution speed will of course depend on the number and nature of the events. Some SIMULA I events may contain a large amount of computation, other events may consist of only a few statements. Hence it is not possible to find a standard "time per event".

The "time per event"-values observed lie between 2 and 10 milliseconds.

A SIMULA I-version of the job shop example in the SIMSCRIPT manual was tested on data provided from a SIMSCRIPT test run. Number of program cards: 48. Compilation time: 4 - 5 seconds. Execution time: 0.6 seconds. Total time on the machine: 43 seconds, mainly consisting of reading the SIMULA I Library from the 25 KC tapes previously used at NCC.

A simple job was programmed both in SIMULA I and GPSS II (on an IBM 7090). The SIMULA I version used 1.75 minutes (compilation 14 seconds), the GPSS II version 4.7 minutes. A large fraction of the execution time was consumed by data transfers to secondary store, using approximately the same amount of time in the two versions.

A standard SIMULA example, "Airport Departure System" (5), used in the one-week course, was also programmed in FORTRAN IV and run on the UNIVAC 1107. The number of cards, compilation time and execution time for the SIMULA I and FORTRAN versions, respectively, were 150 and 325 cards, 11 and 16 seconds, and 5.0 and 5.6 seconds respectively.

As problems become more complex, the advantages of SIMULA show up also in execution speed. SIMULA I run-time efficiency is of course critically dependent on the efficiency of the UNIVAC ALGOL 60 system. Since that system is rapid in compilation, but rather poor in execution speed, this is reflected in SIMULA I execution.

The new SIMULA 67 system will contain a new ALGOL 60 compiler whose generated code is expected to be approximately twice as fast as the current one. This will have a marked effect on SIMULA 67 execution speed.

6. THE SIMULA 67 LANGUAGE

SIMULA 67 is a general purpose programming language with a built-in simulation language capability similar to, but stronger than that of SIMULA I. It has been developed at the Norwegian Computing Center and compilers for this language are now being implemented on a number of different computers. Other compilers are in the planning stage.

A main characteristic of SIMULA 67 is that it is easily structured towards specialized problem areas, and hence will be used as a basis for special application languages.

A complete description of SIMULA 67 is given in (3), while (6) contains a survey. User's Manuals and textbooks will appear later.

Since a main use of SIMULA 67 will be to serve as basis for application languages, this concept should be discussed briefly.

Until now, the computer has been a powerful but frightening tool to most people. This should be changed in the years to come, and the computer should be regarded as an obvious part of the human environment. More and more people should get their capabilities increased through access to "know-how" and the data they need.

This development depends on the condition that the demands on the computer user are reduced, which implies that communication between man and computer is made easier.

Know-how is today to a large extent made operative through "application packages" covering various fields of knowledge and methods. But these packages are in general not sufficiently flexible and expandable and also often require specialist assistance for their use.

The future seems to be "application languages" which are problem-oriented, perhaps in the extreme. Such languages may provide the basic concepts and methods associated with the field in question and allow the user to formulate his specific problem in accordance with his own earlier training. At the same time, such languages should be expandable in the sense that new knowledge acquired should be easily incorporated, even by the individual user.

The need for application languages are only apparently in conflict with the desire for non-proliferation of languages and for general purpose programming languages.

The SIMULA 67 solution is to design the general purpose programming language to serve as a "substrate" for the application languages, by making it easy to orient towards specialized fields and to augment by introduction of additional aggregated concepts useful as "building blocks" for programming.

By making SIMULA 67 highly standardized and available on many types of computers, the application languages also become easily transferable, and at the same time the software development costs for the computer manufacturers may be retarded from the present rapid increase.

SIMULA 67 contains a simulation language as a part of the basic language definition. This simulation language will serve as basis for application languages in electronic circuitry analysis, warehouse layout analysis, job shop analysis, etc. Other application languages will be designed for use in various problem areas in technical and scientific computation, in data processing, etc.

7. PROBLEMS BEING SOLVED BY SIMULA I

This survey of problems solved or being solved by SIMULA I users necessarily has to be incomplete. NCC has no longer complete information on all jobs done in SIMULA I since SIMULA I is used on a number of UNIVAC 1107 and 1108 computers. Also some jobs cannot be mentioned because of the user's security regulations.

7.1 Production

A number of jobs have been done in production planning. The largest program run at the NCC to date is an analysis of the layout of the new job foundry at Raufoss Ammunisjons-fabrikker. The object was to find the proper arrangement and dimensioning of cranes, storage points, etc. The number of program cards for the final version of the SIMULA program was 1130 cards. The compilation time: approximately 50 seconds.

Execution time for simulation of 2.5 days' operation: approximately 22 seconds (250 products and approximately 200 processes giving service to the products).

A series of jobs have been done by the Swedish firm ASEA. The first one analyzed the layout of a new factory with 22 machine groups (1 - 9 similar machines in parallel, in total more than 60 machines) with batch-processing of 17 different order types (each with its own defined or preferred paths through the factory). Components for the same order entered the system at two separate inlets, being matched later on. The machines, each kind having its own set of decision rules for operation, were served by 50 operators, grouped in 8 classes according to combinations of skills. The number of program cards was 340, compilation time was 28 seconds, execution time for 10 weeks simulation 390 seconds.

Larger jobs have been analyzed later on by ASEA's own people.

A smaller analysis of a metal foundry has been conducted to point out bottlenecks in the production and to determine available areas. The simulation also gives information regarding pressure of work with varying manpower and production. Program size is 400 cards.

A SIMULA program was used to optimize the size of the unit production equipment in a detergent plant, taking into account the mutual impact between units.

Actual pieces of equipment to be optimized was number and size of blowing towers, intermediate storage capacity and packing machines.

The program calculated the flexibility of the production system, number of shifts necessary and average stock levels. Alternative operations of the factory and alternative planning systems were tested.

The program generated sales figures for all products, calculated stock levels and by specified planning routines production quantities were calculated. The time unit was 1 hour, and the total simulation period was 3 months operation.

Generation of sales figures and production capacities was based on a statistical investigation of empirical data. Actual statistical distributions were normal and truncated normal.

Number of instructions were roughly 500. Running time on the computer per alternative was, including compilation time, 3/4 minutes. Lines printed per alternative were 930.

A SIMULA program to study different sequencing and production line balancing policies in general job-shop situations has recently been developed but no results of applications are reported as yet.

7.2 Administrative Problems

A simulation of job handling in an advertising agency has been made for a market research institute. The simulation takes into account customers, job requests (for campaigns), consultants, creative teams, the agency's traffic handling office and review boards, and analyses the flow of jobs (possibly in loops between the customer and the agency or within the agency). The purpose of the simulation is to find suitable job handling procedures and more suitable information exchange between customer and agency. Program size: 450 cards. Compilation time: 15 seconds. Execution time: 20 seconds (with 5 consultants and 5 creative teams).

At the Royal Institute of Technology of Sweden, an administrative gaming model has been developed by J. Bubenko. It is used in the Institute's regular SIMULA I course. The model is partially under control of the "game controller" and partially under individual control from the participants who independently may write their own SIMULA I subprograms describing their "company policy". All subprograms are pooled together with a control program during the gaming.

SIMULA I is currently used in hospital planning. Details are not available.

7.3 Inventory and Warehousing

As part of several larger jobs in inventory control simulation of alternative reordering policies and forecasting techniques have been made.

An analysis of the optimal layout of a warehouse has been made by a consulting firm for a large hardware store. In this connection a simulation program to study the effect of different layouts was written. Approximately 50.000 different items are in stock. The purpose is to find a layout giving fast service combined with minimum cost of goods handling. Information is sought on the proper number of handcars, trucks or movable cranes and the positioning of the main groups of items. Program size: 504 cards. Compilation time: 15 seconds. Execution time: 3 minutes (2 days with 250 orders totally containing 2000 items per day.)

This project is now being extended to a study of possible ways of making a general system to aid in warehouse planning by creating a SIMULA 67 special application language.

7.4 Transportation

A large study has been made of the transportation, preproduction and storage system of a Norwegian paper mill.

The paper mill gets all its deliveries by trucks according to a very complicated arrival pattern, depending on hour or the day, day of the week, week of the year (snow conditions, etc.). There are two kinds of raw material, each with their own handling characteristics.

The deliveries are (if space is available) placed on log decks before 2 chippers or at a round timber store, necessitating later transportation to the log decks. From the log decks the timber is automatically brought to the chippers and the chips are either placed in a large outdoor chip store or (from chipper 1) to the bins which supply the factory.

The chippers are different in capabilities and complex decision rules have to be taken into account. The fan of chipper 1 is also used to transfer chips from the chip store to the bins. The knives of the chippers have to be changed according to a probability distribution depending upon the accumulated actual use of the knives. Chipper breakdowns of various kinds also must be considered.

The object of the study was to determine the desirable capacity of the chipper system when production was increased, to evaluate an extension of the log decks and to determine the transfers to the round timber store.

The program consisted of 500 program cards, programming and debugging 80 working hours (the system was rather well known from earlier work for the firm). Compiling time was 24 seconds, execution speed approximately 160 seconds for simulation of 1 year's operation (approximately 64.000 events).

7.5 Harbours and Shipping

2 harbour simulations have been made in SIMULA at NCC. In the first analysis the effects of loading capacity and boat sizes were studied, and more than one boat could be accommodated at the same time. Programming and debugging: approximately 60 working hours. Number of program cards: 300. Compilation: 22 seconds. Execution time: 37 seconds for 1 year's simulation (900 boats).

In the second job the effects on shipping service of a doubling of a factory's production was studied. Programming and debugging: 45 hours. Number of program cards: 498. Compilation: 33 seconds. Execution time: less than 20 seconds for simulation of 1 year with 130 boats.

The Norwegian Institute of Transport Economics is conducting a research project for UNCTAD (United Nations Committee for Trade and Development) concerning the choice between different investment policies for large international ports. Presently the harbour under study is Casablanca, and the aim is to find a general means of measuring the effectiveness (based on the in-port delay of ships) of different investment plans. A SIMULA model of Casablanca harbour is used to simulate the effect of varying the number of tugboats, cranes, docks, pilots, etc. for a given traffic. The model will later be made independent of any particular harbour.

At the Norwegian Institute of Technology SIMULA is used to analyze ship transport systems, especially regularly scheduled sailings. The operation of a fleet of a given line is simulated for a reasonable period to predict the effects of different policy- and decision rules, technical characteristics, etc.

The main purpose of the simulation is:

a) To decide between ship and fleet characteristics and replacement policies,

b) to investigate line structures and service standards,

c) to analyze sensitivity of decisions to the reliability of forecasts of future transport demand.

The SIMULA program is part of a larger system and consists of 725 cards.

7.6 Planning and Scheduling

A major bus transportation firm requested an analysis of the possiblity of setting up personnel allocation plans on computers. Only some experimental studies have been conducted. The program is able to allocate (slightly better than the present manual procedure) 14 drivers to one of the buslines in 2 seconds. Number of program cards: 90.

A simple PERT program has been written in SIMULA. Number of program cards: 103.

The program was written to evaluate the effect of introducing the individual completion times as stochastic variables. When random drawings were used, 12 sample runs showed a distribution of total completion time which was not in accordance with the generally used criterion of adding up variances. We also found a sample of possible critical paths instead of just one. Execution time for a system with 24 activities and 12 runs: 5 seconds.

A simulation model of the programmers' punching room at NCC was used to analyze the possible benefits of installing additional machines or changing the queueing rules. Several machines are available to the programmer, but he is only allowed to use a machine a certain number of minutes at a time. Program size is 190 cards.

7.7 Renewal

Det norske Veritas, a ship classification society, used SIMULA in a pilot study of the reliability of complex control systems. The object was to evaluate the best combination of duplication, scheduled substitution and inspection of the components.

7.8 Computer Operation and Monitor Design

NCC has made a simulation study of its own UNIVAC 1107 Computer System Operation. The study has been presented in the report "Simulation of a Computer Installation" by Bjørn Myhrhaug, G.A. Wold (NCC) and T. Strøm (Royal Institute of Technology, Stockholm, Sweden). The analysis comprises the input of programs, the operators, the controller (or "scheduler"), the I/O devices, tape decks, drum files and the computer.

The object was to evaluate principles for classifying and handling jobs, ensuring an effective use both of programmers and the computer. Number of program cards: 530. Compilation time: 32 seconds.

A "skeleton description" (lacking data collection statements) of an advanced real time system was simulated as a NCC test job. The program studied the performance of alternative layouts and operating principles for a hypothetical system controlling the behaviour of many complex devices through a telecommunication network and a group of centrally located computers.

Two other SIMULA programs simulate alternative computer monitor systems.

7.9 Computer Hardware Design - Debugging

The LOGIC package is an aid in the field of digital computer design. The package is used in the design stage of debugging. (Thus taking the place of a first prototype.) When the design stage is complete and the logic elements are given physical addresses, the package will be used to prepare either wirelits for a wireman or a control tape for an automatic wiring machine, or both. It is intended to extend the package for preparation of logic drawings and users'/maintenance manuals.

The feasibility study for this package was made using SIMULA to determine the benefits of simulation digital systems. The digital systems consisted of basic logic elements interconnected to give a desired logic function.

The program contained approximately 270 cards requiring 18 seconds compilation time and execution time of 20 - 30 seconds. The execution time consisted of 10 - 15 seconds, printing a sophisticated output. The test run was made on a logic function of a total of 100 elements of 7 basic types. The program was in no way optimal and much time was used on data handling.

A second program has since been written for use as a teaching aid at Norway's Technical University in Trondheim. The advantages of using SIMULA in this application is that the student has to consider his problem very carefully when writing his activities. It forces him to research into why his simulation does not at first give the results that he expected. Upon careful study of the results, he is normally required to modify his activities. Thus the student obtains a much deeper understanding of the subject than by carrying out practical studies where he can usually get around difficulties without even understanding why they occur.

The short writing and debugging times for SIMULA have encouraged its use in the initial study phase and even development phase of a project. Techniques and solution methods are easily and quickly experimented with the use of SIMULA.

The LOGIC system has been the basis for the LOGIC application language currently being marketed by the Norwegian Computing Center in various countries. LOGIC allows the user to introduce into the system new components which he describes himself in SIMULA I. Standard components and user-oriented macro statements are part of the basic LOGIC language.

7.10 Social Systems

The "Opinion Transmission" program is a simulation of social diffusion: A finite population is assumed. Each person is characterized by his attitude towards a proposal advocated, by his "exposing power", his "stubbornness", his average number of contacts per time unit. The persons contact each other, trying to convince the other persons to accept their own point of view if not in agreement. The exchange of opinion is continued until a stable state of a majority approval (or disapproval) of the proposed action is established.

A second program, "Epidemic", reproduces the main characteristics of the spreading of an epidemic in a population. A certain "life cycle" for the disease, contact frequencies in the population, treatments of various kinds, (operating also on the "environment" of the infected persons), are assumed.

Four programs have been made to simulate different models of message diffusion within a finite population.

Model 1: This model is the simpler one. In each time interval a person who knows the message brings it to some other randomly chosen person whom he has not yet told the message.

Model 2: Same as model 1 except that the distance between the individuals is taken into account.

Model 3: Same as model 1 except that the individual stops sending the message after a certain time.

Model 4: This model differs from the other models with respect to the existence of groups. The message is only brought to persons belonging to the same group as the messenger. Individuals may of course belong to more than one group.

7.11 Statistics

The random drawing facilities of SIMULA have been used to estimate probability distributions. One project was to check the probability model for the distribution of blood discs (erythrocytes) in a liquid.

7.12 Communications

The Norwegian Civil Defence Authority has conducted an investigation of their communication system for reporting radioactive fall-out. The system under study consisted of permanent and mobile report stations and differences between an "actual" radioactive fall-out area as reported in the field and as viewed from central headquarters were studied. The program consisted of 470 cards and execution time was 15 seconds for 30 - 40 stations.

A major newspaper has just finished a study of a telephone exchange. The aim of the study is to find out what will result from an expected increase in the calling frequency. The service efficiency depends on the number of switchboards, whether the call is local or not, the way the system of queues of calls is handled, and the effectiveness of the operators. Experience gained in this study will later be used in connection with expansions of the newspaper's services.

Studies have been made of traffic handling in country-wide military telephone systems. Program size 165 cards, compile time 45 seconds (includes reading the SIMULA Library from tape), typical execution time 10 minutes.

7.13 Biological Systems

A certain type of neurons in the spinal narrow (DSCT-cells) receives nerve-impulses from the stretch-receptors in the muscles and the information is transmitted to the cerebellum. The pattern of such impulses have been studied experimentally The sequence of impulses has characteristics that differs from what is found in other neurons.

The problem was to find whether the impulse pattern can be explained from those mechanisms that have been shown to be present in other cells. The problem was attacked by formulating the theories of the integration of nerve impulses and DSCT-cells in SIMULA and running a series of simulations with parameters drawn from experimentally determined distributions.

With certain choices of the parameters one was able to reproduce the impulse pattern found experimentally in the DSCT-cells. Some of these parameter values have later been found to agree with results of direct observation in the cells.

Further plans is to extend the model to take into account the inhibition known to exist from neighbouring muscles. Inhibition can take place in two principally differing ways, the simulation will probably point out the correct one. The model consists of 130 cards, compilation time is 10 seconds, execution time 3 - 4 minutes.

7.14 Associative Networks

Mrs G. Eckblad at the Institute of Phychology at the University of Oslo is carrying out studies of associative networks. Among psychologists engaged in the simulation of human information processing today, one will find a difference of opinion concerning the amount of central control to be incorporated into the models. In the General Problem Solver of Newell, Shaw and Simon there is one central executive instance with full control of the sequencing of events in the system. In the problem solving model Argus by Reitman, Grove and Shoup on the other hand, there are in addition to a still powerful central executive instance also other "dynamic centra" in the form of an "active semantic network", from which action can be initiated and the on-going process interrupted. The present project while essentially in agreement with Reitman's point of view, represents an attempt to explore a still more extreme possibility: a system which at least starts out without any central executive instance at all. The simulations will be carried out in SIMULA I.

As the project has barely started and will have to undergo nearly continuous revision in the future, specific programming details are of small interest. The main component of the system is a growing set of active semantic units (modelled upon those in Argus), that are capable of building up an associative network among themselves. At present, the system has no problem solving ability, and no spatial or temporal structure except for that which may result from the free interplay of the semantic units themselves.

7.15 Market Research

F.O. Falch of the Research Projects on Economics of Distribution at the Norwegian School of Business Administration, has made a SIMULA model to study the interaction between customers and stores. This project is part of a larger investigation aimed at studying the structure and dynamics of market systems through computer simulation. The model takes into account different customer characteristics such as consistency, their reaction to market shares, sales effort, etc. and is presently being extended to include several stores and the effect of chaining and central ordering policies.

The program size is 300 cards, compile time 22 seconds.

S. Braaten at the Swedish firm IMAS is together with K. Nygaard at the Norwegian Computing Center are developing tools for analysis of marketing campaigns. The models integrate recent results in social psychology, diffusion theory and communication theory. An initial model (the "SIMCOM" model) is completed in SIMULA I. Future work will continue in SIMULA 67.

7.16 Railway Marshalling Yards

From J. Achermann, working for the Swiss Railways, the following account has been received:

"Simulation of a Railway Marshalling Yard

In a railway marshalling yard, goods trains are arriving at an entry queue. Here some work is to be done, such as controlling the documents and the waggons, loosening the couplings, etc. When this is completed, the goods train is ready to be pushed down the marshalling slope. There, each goods waggon rolls on its own to a direction road according to its destination. From those waggons in the direction roads, new goods trains are formed, which then leave for the destination marshalling yard.

A marshalling yard can be treated as a system of waiting queues and service points. All waiting queues have a limited capacity. The problem is to simulate on the computer road systems, activities, goods trains and goods waggons which interact with each other.

By means of the simulation language SIMULA, the marshalling yard at Biel was modelled on a UNIVAC 1107. SIMULA proved to be a very good tool. Thanks to the great flexibility of SIMULA, it was possible to model accurately and in great detail all the activities in the marshalling yard and, so far, to compare the simulated events of one day with reality using about 80 arriving and departing trains. The agreement was satisfactory.

The Swiss Railways want to concentrate and optimize the goods traffic according to a new marshalling concept. The marshalling work is intended to be undertaken at only a few marshalling yards. These yards have to be more effective. The program "Simulation of a Railway Marshalling Yard" is intended to model this marshalling work in advance and to show whether the expected effectiveness could be achieved without creating traffic jams. With new marshalling yards, it can test the proposed operating concepts." (Translated from German).

7.17 Evaluation of Production Planning Schemes

Dr. Brun, working for UNIVAC, Switzerland, has developed a program which evaluates the benefits to be expected from introducing certain computer programs for production control.

The evaluation consists of a comparison between results achieved with present methods, and with the new computer-based schemes.

8. REFERENCES:

(1) Dahl, Ole-Johan and Nygaard, Kristen:
SIMULA, A language for programming and description of discrete event systems. Introduction and user's manual, Norwegian Computing Center, May 1965.

(2) Dahl, Ole-Johan, Myhrhaug, Bjørn and Nygaard, Kristen:
SIMULA Tracing System.
Norwegian Computing Center, November 1967.

(3) Dahl, Ole-Johan, Myhrhaug, Bjørn and Nygaard, Kristen:
SIMULA 67, Common Base Language.
Norwegian Computing Center, May 1968.

(4) Kirkerud, Bjørn:
ALGOL/SIMULA Plot. Routines for the Kingmatic 1215.
Norwegian Computing Center, September 1967.

(5) Nygaard, Kristen:
Airport Departure System, a standard SIMULA example.
Norwegian Computing Center, May 1965.

(6) Dahl, Ole-Johan, Myhrhaug, Bjørn and Nygaard, Kristen:
"SIMULA 67, Basic Information"
Norwegian Computing Center, May 1968

(7) Nygaard, Kristen:
"Report on the Use of SIMULA up to December 1965"
Norwegian Computing Center, December 1965.