AN IMPROVED SIMULATION LANGUAGE APPROACH FOR LARGE SCALE SIMULATION AND TIME CRITICAL OPERATION

WILLARD M. HOLMES and ALEX C. JOLLY
Army Missile Command
Advanced Simulation Center
Redstone Arsenal, Ala. 35809

ABSTRACT

This paper describes the present effort to implement a CSSL type language for large scale system application in the Army Missile Command's Advanced Simulation Center. The Center's function is to develop multiple realtime simulations using a complement of hybrid computer operations with a CDC 6600 digital computer, seven analog computers, seven mini computers, two intermediate size hybrid computers with CDC 6600 interaction. Additionally, the Center includes three environmental simulator cells i.e., Infrared Simulation System (IRSS), Electro-Optical Simulation System (EOSS) and the Radio Frequency Simulation System (RFSS). These cells are used to allow appropriate sensors to be included in the simulations as hardware-in-the-loop. The capability exists to include one or more of the cells in any one simulation.

The primary goal for using the simulation language approach for ASC's simulation tasks is to reduce both manpower and leadtime required to achieve a fullup large scale simulation, second, but an equally important fact, is to establish some practical guideline for documenting the simulation programs as they evolve through the stages of mathematical modeling, simulation models, digital computer programs, hybrid computer simulation, and finally to time critical hardware-in-the-loop operation.

INTRODUCTION

Simulation has played a continually expanding role in the air defense weapons development community during the past two or more decades. This expanded use of simulation has been paralleled by the development of more complex simulation tools and sophistication of modern weapon systems. The increased use of simulation has evolved from a pure analog computation of the type initiated in 1956 and results reported in 1962 by Biggs and Cawthorne(1). A very similar development of a tactical missile development using simulation was reported by Driscoll and Gilbert in 1972(2). The significant difference in the reported operations of these two efforts is the expanded application of simulation with a hybrid computer with hardware-in-the-loop operation. Inherent in this added simulation tool is the requirements for a comprehensive operational software package. The software developed for hybrid computer operation is generally unique to the particular hybrid computer configuration. However, for simulations requiring an all digital operation within the digital computer, some flexibility in the available software does exist in the form of simulation languages. The development and use of a simulation language for large scale all digital simulation has been reported by Gauthier and Mitchell(3).

THE SIMULATION ENVIRONMENT

The Army Missile Command's Advanced Simulation Center (ASC) has been established to perform multiple, large scale realtime continuous system simulations of air defense missile systems during the design, development and production stages of these missiles. Emphasis is placed on the inclusion of hardware in the simulation loop with specially designed facilities for the purpose of generating appropriate environments for the missile guidance sensors. These facilities are used for sensor testing in open loop operation in addition to their closed loop simulation capabilities. Extensive hybrid computational hardware is used to interconnect the environmental facilities and to close the guidance loops by simulating the missile in flight. Figure 1 shows a pictoral representation of the simulation facilities.

It is the function of ASC to develop the techniques and methodology which permit multiple realtime simulations to be performed simultaneously by the hybrid computer and environmental facilities, in a manner analagous to multiprogramming in a large scale digital computer. Thus, a critical need exists for the tools, primarily software, which improve the ease of use of the equipment, offers improved productivity and/or reduce costs associated with a particular simulation and, in addition, provide a basis for simple and accurate documentation.

Missile system simulation has traditionally been oriented to the design stages of a missile's evolution, with the simulation models being used to optimize design parameters and calculate the system's sensitivity to various design changes. Flight testing of the hardware was the usual method of validating the system design. Because of the high cost of flight testing, efforts are being made to develop alternate capabilities to achieve more for the dollar or accomplish the same tasks at reduced cost. The comprehensive application of simulation has emerged as a viable tool to complement flight testing with a resultant cost saving(4). This is in large measure the motivation for elaborate hardware-in-the-loop simulations. Test flights will be used to provide validation or development of the simulation models for this role.

Environmental Simulation Facilities

The environmental simulation facilities in the ASC consist of Infra-Red, Electro-Optical and Radio Frequency simulation systems respectively known as IRSS, EOSS, RFSS. Each system is characterized by an elaborate target generator, a flight motion simulator and some computer hardware independent from the centralized hybrid computers.

The Infra-Red Simulation System(IRSS) . The IRSS is a simulation tool for the design, development and evaluation of infrared sensor systems applicable to surface-to-air and air-to-air missiles. Sensors in the 0.3 to 0.7 and 1 to 5 micron bands are hybrid-computer controlled in six degrees of freedom during the target engagement sequence. A gimballed flight table provides pitch, roll, and yaw movements to the sensor airframe. A target generator simulates a variety of target/background combinations which include tailpipes, plumes, flares, and fuselages in single or multiple displays against clear sky, dark clouds, overcast sky, and sunlit cloud backgrounds. These are then displayed in azimuth, elevation, range, and aspect by the target projection system through a mirror/lens network, a display arm, and a display mirror. Simulation capability ranges from open loop component testing to closed loop total system simulation. A pictorial representation is shown in Figure 2. The target generation system is nonimaging and consists of an assembly of equipment and components, which provide for generation of simulated aircraft targets, backgrounds, and countermeasures. The purpose of this assembly is to present to the guidance unit under test suitable radiation sources to simulate the physical, radiometric, and dynamic characteristics of targets, backgrounds, and countermeasures. These characteristics are designed to be manually or automatically controlled. Local instrumentation provides manual control, while automatic programmed control is provided through either the local hybrid computer consisting of DEC PDP-11 and AD4 or from the central hybrid computer facility.

The Electrooptical Simulation System (EOSS). The EOSS shown in Figure 3 provides realistic and precisely controlled environments for the nondestructive testing of a wide variety of ultraviolet, visible and near infrared sensor systems. Actual sensors are hybrid-computer controlled in six degrees of freedom while viewing targets under controlled illumination levels (10 to 10 fc) in an indoor simulation chamber, and under ambient conditions of an outdoor test range. Three-dimensional target simulation is provided by a 32- x 32-foot terrain model/transporter which features a variety of topographical and man-made complexes at 600:1 and 300:1 scales, removable model sections, and fixed and moving targets at any desirable scale. A moving projection system provides 2-D representation.

The terrain model is transported on the longitudinal system and provides a series of straight line contrast areas, bland topography with a variety of contrasting targets, and servo-controlled moving target models. The moving targets provide dynamic tracking capability against changing background scenes. The target model can be tilted to an infinite number of positions from 0 to 30 degrees from the horizontal so that various geometries and altitudes can be accommodated. When the target model is horizontal, it can be rotated in azimuth and secured at each 90 degree point presenting different aspect angles to the seeker.

Radio Frequency Simulation System (RFSS). The RFSS shown in Figure 4 simulates a missile's total mission from launch to intercept in RF and ECM environments, and is designed to enhance capabilities in all phases of missile system research, design, development, and engineering. The primary application is evaluation of RF active, semiactive, passive, and command terminal guidance systems for surface-to-surface, air-to-air, and surface-to-air missiles. Guidance sensors and flight control systems will perform in an environment where aerodynamic moments, angular motions, and electromagnetic signals are realistically produced. The RFSS is a multilevel facility comprising a number of closely integrated rooms. A shielded anechoic chamber simulates a free space environment for the radiation of signals from an array of 550 antennas to a guidance sensor projected through an aperture at the opposite end of the chamber. The guidance sensor is mounted on a Three-Axis Rotational Flight Simulator (TARFS-1). A second TARFS simulates angular motions for the autopilot gyros and a Control System Aerodynamic Loader (CSAL) simulates aerodynamic moments on control surface shafts.

The RF generation equipment consists of four target generators, a reference generator, two denial ECM sources and fuze selection and attenuation. The equipment operates in the 2- to 18-GHz spectrum and provides for the control and generation of RF signals suitable for stimulating the electromagnetic characteristics of airborne targets and environments to be encountered by a wide variety of advanced guidance systems. Control of the RF target generator is performed by an array of seven mini-computers which may operate in a stand-alone mode, or on-line to the central hybrid computer. Communications with the central hybrid computer is available through conventional analog and discrete channels and also via direct memory transfer. The latter mode is known as Direct Cell.

Central Hybrid Computer. A CDC 6600 digital computer with 131K words of 60-bit core memory and 20 peripheral processors comprises the digital computer portion of the hybrid system. A pool of analog computers is provided for assignment to any of the simulator systems. Three (3) AD-4's are currently available with one additional planned. The ASC hybrid computing system (Figure 5) provides realtime computing support for operation of the Center's three environmental simulators. The system's design permits assignment of needed computing hardware to individual simulators in a manner that allows easy reconfiguration for changing requirements. The digital computer's multiprocessing capability provides simultaneous operation of simulators where software memory requirements and hardware timing are compatible with the computer's capabilities. Ports for Direct Discrete/Analog Input/Output (PDDAIO) is the operating system for executive control of the real time simulations. A unique feature is the direct digital/digital links between the CDC 6600 computer and the dedicated digital computers in the simulator cells. These direct links allow digital word transmission at rates up to 1 MHz.

An interconnection and setup subsystem provides cross-patching control for analog and discrete channels throughout the simulation facility. In addition, this subsystem provides the operating system for automation of the setup, documentation and checkout process for the analog computers. A Centralized Multivariable Arbitrary Function Generator allows the hybrid-computer generation of functions of one to three variables.

METHODOLOGY OF SIMULATION DEVELOPMENT

The details of actual developing, implementing, validation and operating a hybrid computer simulation with hardware-in-the-loop operation will be somewhat different for each facility. The standard approach to development of a missile system simulation within the ASC is outlined in Figure 6. Simulation objectives are carefully defined, in conjunction with the customer or simulation user. Following the mathematical modeling and simulation design, implementation takes place along three paths which are not necessarily concurrent in time. These paths consist of an all-digital, an analytical hybrid (i.e. all-software, real-time), and hardware-in-the-loop time-critical simulations.

An all digital simulation is considered to be the essential first step in the computerized simulation process and is used to check the mathematical models and also to provide a means of generating dynamic check data and scaling information for the hybrid programming. The hybrid simulation is generated using the results and program from the all-digital simulation as far as possible; the hybrid program contains all the system components in model form and executes in realtime or faster. Hardware-in-the-loop implementation is the final step in the total simulation process and requires the substitution of hardware components for the corresponding models, insertion of interfacing equipment between computers and hardware and integration of the target generation system into the total simulation. The hybrid simulation is used as a checkout tool for the hardware-in-the-loop simulation; it is possible to run the hybrid and the hardware components in a parallel mode during simulation development, with the hardware open loop to avoid the chance of damage to expensive equipment, and check one against the other.

Not shown in the procedure above, but inherent in the process as used in the ASC is the requirement for documentation. This has been one of the more overlooked aspect of efficient simulation development and utilization as reported in the open literature. As most any simulation developer can attest, documentation is a costly aspect of the simulation development and utilization cycle. Surprisingly, the most important cost factor is associated with not having systematic documentation throughout the development and utilization cycle. When the major problem of the documentation is completed after the simulation is developed, this is too late for efficient use of resources directed toward documentation. While the intermediate milestone for documentation will vary, some firm functional milestones do exist. The documentation of the math models to be used for the initial digital simulation development is required before effort is initiated toward developing the digital program. The documented updated math models and digital program must be complete before the hybrid computer simulation development effort is initiated. Likewise, the hybrid simulation must be documented before the hardware-in-the-loop effort is initiated. In a later section, it will be shown how the use of a simulation language plays a significant role in achieving the documentation objectives.

SIMULATION DEVELOPMENT USING A HIGHER LEVEL SIMULATION LANGUAGE

Since the publication of a recommended structure for continuous system simulation language (CSSL) in 1967(5), a variety of packages have been made available to the simulation developer(6). Continuing interest in a CSSL type language centers on a language which would permit the unification of the approach outlined above. It is logical to consider the development of software tools to complement the extended simulation tools described in the simulation physical facility. Comparisons have been made between an all digital simulation developed using a Fortran language and a high level simulation language(7). These results show that typically an all digital simulation can be developed using a high level simulation language with reduced lead time and more efficient use of manpower. This is consistent with the studies conducted in the ASC.

SIMULATION LANGUAGE FEATURES

It is highly desirable that a simulation program written in this language be structured for eventual use in realtime hardware-in-the-loop (HWIL) operations. To accomplish this would require that with minor modification, it be able to operate in all modes described in the methodology of simulation development. Desirable features of the language also include those which relieve the user of a large number of tedious details such as I/O operations, programming of integration routines, conversions from s-plane to time domain differential equations, perform interpolation in tabulated functions and provide common non-linearity simulations as macro or function calls. It should also be capable of providing a simple and flexible means for calculating static and dynamic checks for simulation portions assigned to the analog section of the hybrid simulation, and be capable of providing output of these sections, at the source program level, which can be used by an automatic analog programming and scaling program.

Use of the language in a realtime mode requires the capability of hybrid input/output (i.e. analog-digital and digital-analog conversion plus discrete binary) and the connection of various program sections to external computer interrupts. Differences in the bandwidths and integration step requirements of the analog and digital computers implies the desirability of multi-rate independent integration procedures; this requirement, coupled with the trichotomy of hardware-in-the-loop simulation into digital, analog and hardware sections, leads to the conclusion that some form of modularity in the language use is highly desirable. Interfacing between modules then becomes an important factor which requires consideration of global and local variables, argument transmission and module hierarchy. A further benefit of a modular concept is that modules representing subsystems within the simulation model may be checked independently and assembled into the overall simulation subsequently. Two other desirable features include the ability to interact with a data base or data management system, and some standard processes for simple statistical calculations. The former feature has important applications in the area of incorporating measured test data into a simulation and the latter is required for simulations which contain Monte Carlo sampling and analysis procedures. A further useful feature is the calculation of partial derivatives of a simulation solution (missile trajectory, for instance) with respect to all the state variables by perturbation methods: the resulting matrix of partial derivatives may then be used in the covariance approach to simulation of stochastic systems, which is an alternative to Monte Carlo sampling.

SIMULATION LANGUAGE UTILIZATION

The experience and use of simulation languages in the background of the ASC has been, for the most part, of the CSSL type. The progress in continuous system simulations languages development has been with few exceptions, a variation around the original specification with different implementation for the same language operation. A variant of CSSL known as RSSL (Raytheon Scientific Simulation Language(8) is presently being used for digital simulation development. As an all digital simulation language, the 1967 specification is a sub-set. This language presently does not include any realtime capability for hybrid simulation or hardware-in-the-loop operation. While the language operation is confined to a single derivative section, many of the other language features previously described are included for use in simulation development.

Presently RSSL is used for the twofold requirements of; an all digital simulation and determining analog/digital partitioning prior to initiating hybrid computer implementation as indicated in Figure 6. As a further step to minimize leadtime and achieve efficient use of manpower, a generic modular missile simulation program has been developed using RSSL (9). The simulation has been developed using the modular structure of the physical system as shown in Figure 7. In addition, the format and documentation as outlined in this document is used to meet the progressive documentation requirements for this part of the simulation development.

With the present limitation of a single derivative section, the analog/digital partitioning determination is accomplished with the use of Fortran subroutines. Since RSSL is totally compatible with Fortran, that part of the model identified as residing in the digital computer for hybrid operation is Fortran programmed. The single derivative section is used to represent the analog computer. Since RSSL does not have a realtime capability, the Fortran subroutines can be used directly in the realtime digital program for hybrid simulation and HWIL operations. Errors associated with the hybrid computer interfaces and special HWIL interfaces can be conveniently introduced into the all digital simulation. In addition, execution time for subroutines associated with the digital portion of the hybrid are available.

SIMULATION LANGUAGE DEVELOPMENT

The simulation language features pointed out in a previous section included both the characteristics hoped for and the operations typically available for supporting hybrid computer simulation development. A variant of the 1967 CSSL specification, an Advanced Continuous Simulation Language (ACSL) (10) has been obtained by the ASC and is presently being modified and extended by the developer. The ACSL extension will give the package a time-critical hybrid computer simulation orientation and still retain the basic structure and capability of the simulation language. The developers of the 1967 CSSL specification gave some indication of the need for flexibility and extensibility in the language specification, but no attempt was made to specify the requirements for time-critical application. The specifications included the concept of multiple derivative sections, however there has been no formalized relation established between the derivative sections, for this and other reason, there has been limited implementation of this part of the specification.

The first extension of ACSL was to include multiple derivative section capability. This allows the simulation developer to view his physical system in terms of the range of dynamics that may be involved in the system operation. The system may be segregated accordingly with different integration step sizes. In addition to the independent step size for each derivative section, a a separate integration algorithm can be selected. The details are not described here, but the execution of the multiple derivatives is driven from an event list based on the greatest distance from the beginning of the next communication interval. The issue of global versus local variables has not been addressed directly; presently all variables are implicitly global to all elements in the simulation program. To some extent, this is undesirable when considering a modular structured simulation with independently written and tested program elements. Including local variables which have meaning only within a particular derivative section remains under consideration. Internal control of the multi-integration process requires the extension of the usual state vector update procedure by inclusion of an event control table which points to the next state vector to be updated, as previously indicated. It is not required that all derivative sections actually evaluate derivatives, but may simply contain "housekeeping" or algebraic operations.

The extended lantuage will include a set of hybrid computer drivers to control hybrid input/output, analog mode and component control and integration with the realtime monitor of the digital computer. This interaction includes interrupt connections and release of source program blocks in the form of derivative sections with facilities for realtime data recording. When realtime operation is envisaged, the macro file is translated to a separate subroutine which can be tied to a separate interrupt, which in turn says that each derivative section can be interrupt driven.

A comprehensive benchmark program is under development for using as many as twelve derivative sections and including combinations of different step sizes and integration algorithms. One or more analog computers will replace one of the derivative sections for hybrid computer operation. A second derivative section will be replaced with the actual missile system hardware located in the RFSS. Other derivative sections will be used to represent the IRSS and EOSS for multiple environmental cell interaction and operation. The digital portion of this benchmark development will be a second generation generic modular missile.

REFERENCES

(1) Biggs, A. G. and Cawthorne, A. R., "Bloodhound Missile Evaluation". Royal Aeronautical Society Journal. Vol 66, No. 621 PP 571-587, Sept 1962.

(2) Driscoll, Thomas R. and Gilbert, William H., "Tactical Missile Development Using Computer Simulation", PROC. of 1972 Summer Computer Simulation Conference, Vol. 1, PP 680-695, July 1972.

(3) Gauthier, Joseph S. and Mitchell, Edward E. L. "Large Scale Simulation in the Raytheon Scientific Simulation Language", PROC of the 1975 Summer Computer Simulation Conference, Vol 1, PP 133-137, July, 1975.

(4) GEN Deane, Jr., John R. "Single Integrated Development Test Cycle Policy", Directive Letter, Department of the Army, AMC. Jan 1975.

(5) Simulation Councils, Inc., "The SCI Continuous System Simulation Language (CSSL)", Simulation Vol 9, PP 281-303, Dec 1967.

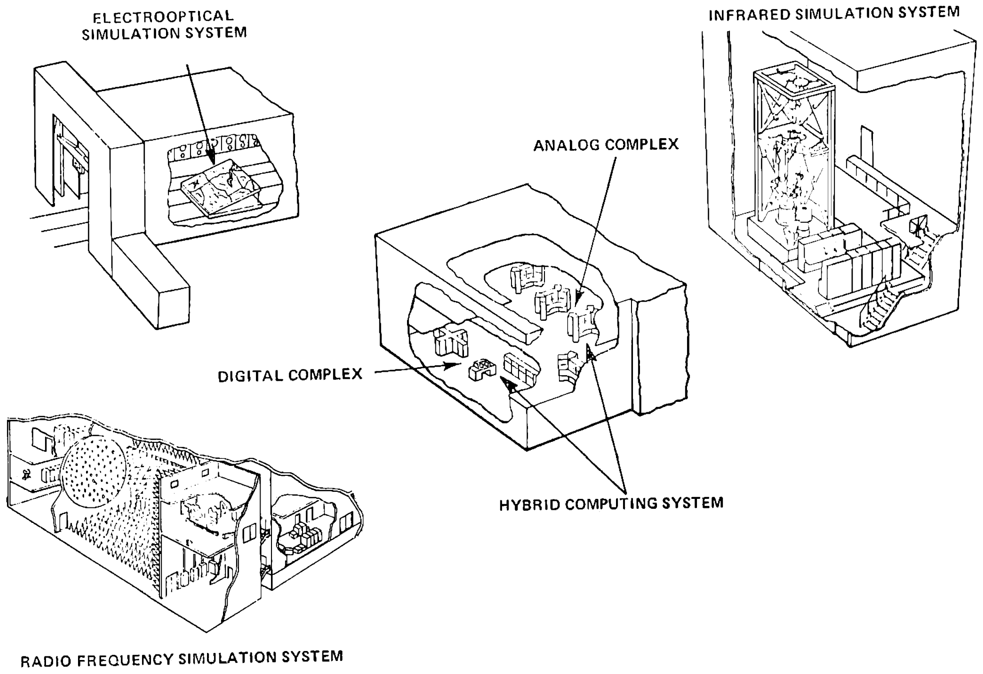
(6) Nilsen, Ragnar N., PhD and Karplus, Walter J., PhD "Continuous-System Simulation Language: A State-of-the-Art Survey". Annales de l' Association Internationale Pour Le Calcul Analogique - No. 1 - PP 18-25, Janulier,1974.

(7) Hart, Roger, "Two Steps Forward - SCSL & CSSL II" (or 'How to Improve the Credibility of Simulation Language?'), PROC. of 1975 Summer Computer Simulation Conference, Vol. 1, PP 150-157, July 1975.

(8) Raytheon Co., RSSL Raytheon Scientific Simulation Language User Guide/Reference Manual, BR-7468.

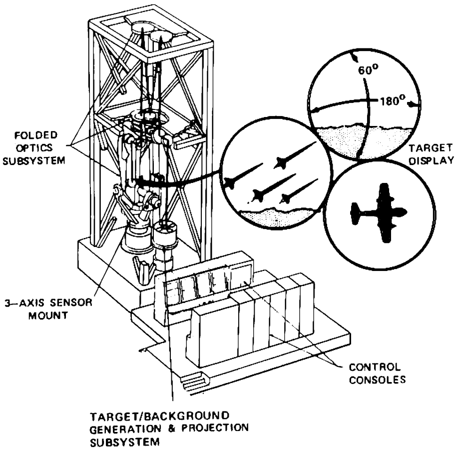
(9) Holmes, W. M., Jolly, A. C., Kesting, J. W., Merriman, D. B., Paley, B., and Webb, J. A., "Modular Missile Simulation Model", U. S. Army Missile Command Technical Report RG-75-21. 13 Dec 1974.

(10) Mitchell and Gauthier, Assoc., Advanced Continuous Simulation Language User/Guide Reference Manual, 1975.



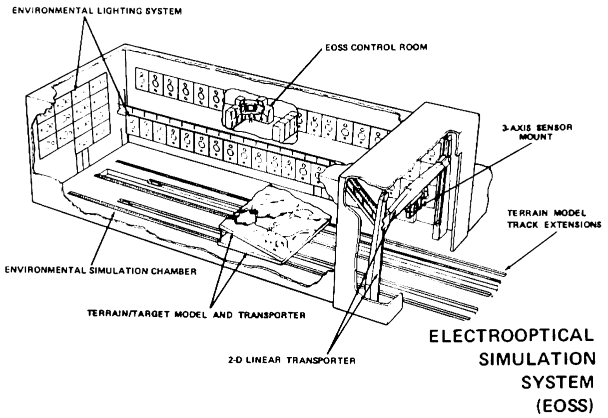
THE ADVANCED SIMULATION CENTER's ENVIRONMENTAL SIMULATION FACILITIES

FIGURE 1



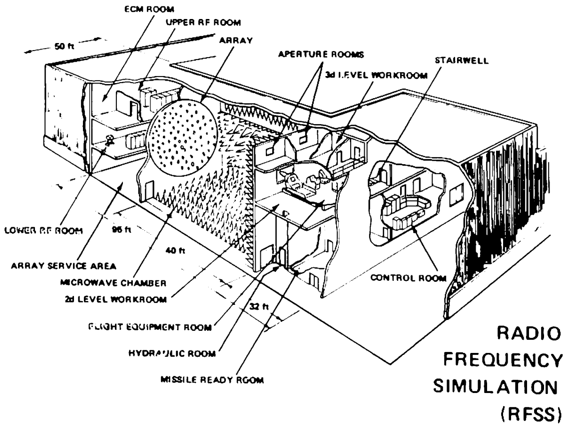
THE INFRA-RED SIMULATION SYSTEM (IRSS)

FIGURE 2



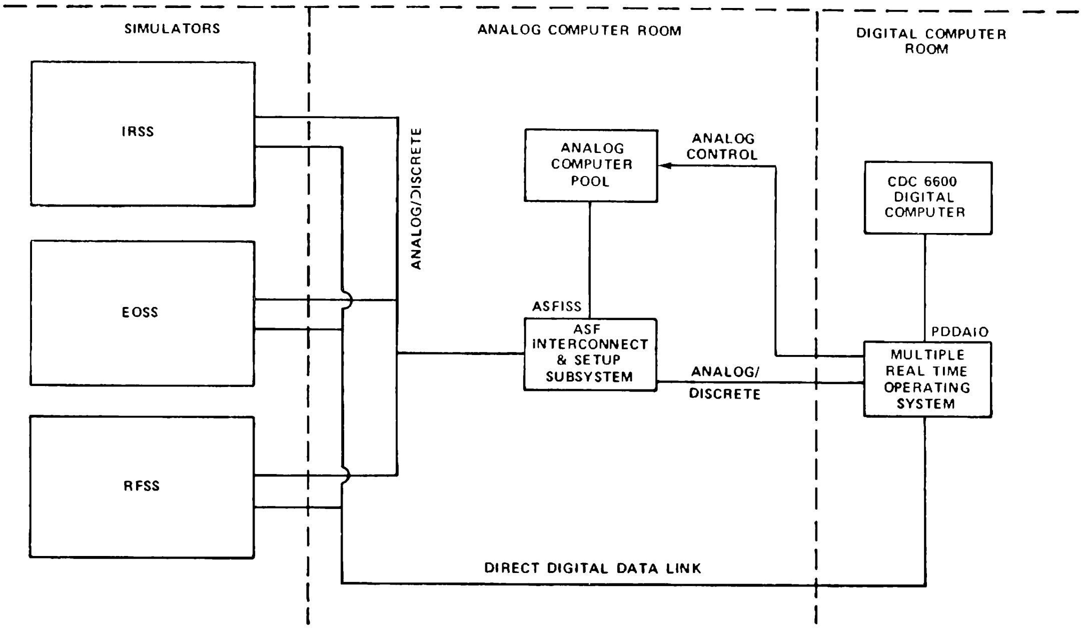
THE ELECTROOPTICAL SIMULATION SYSTEM (EOSS)

FIGURE 3



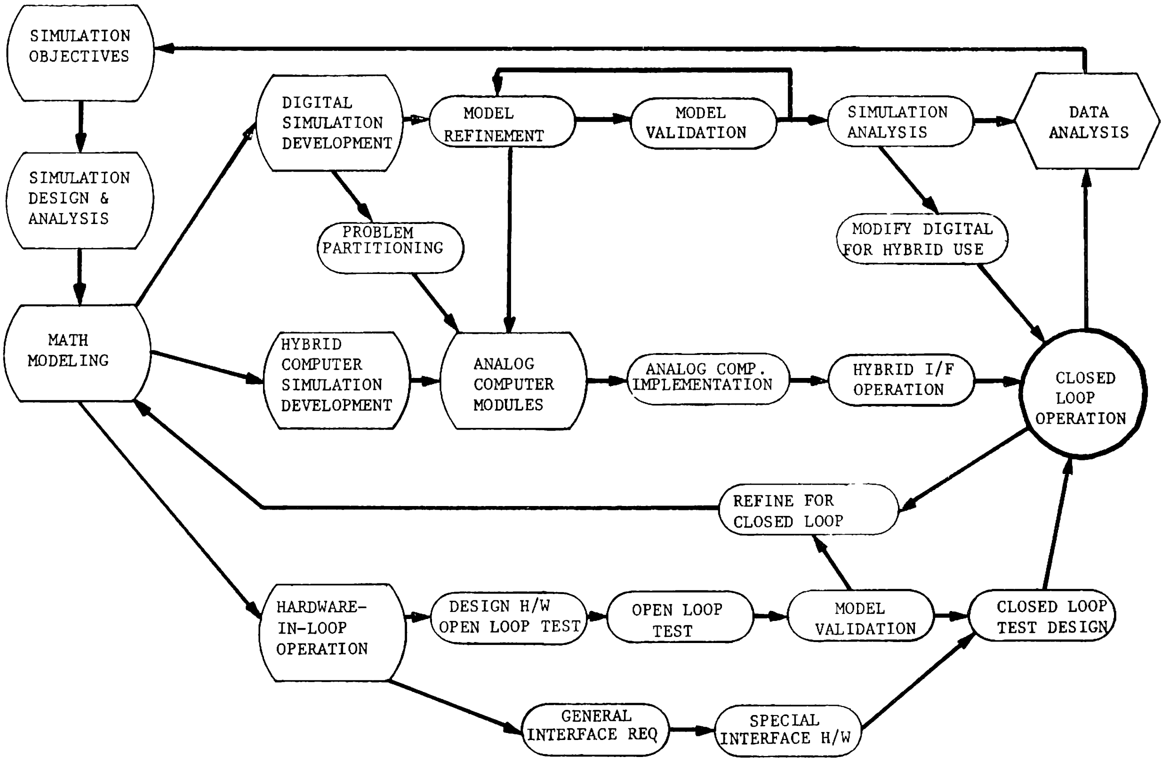
THE RADIO FREQUENCY SIMULATION SYSTEM (RFSS)

FIGURE 4



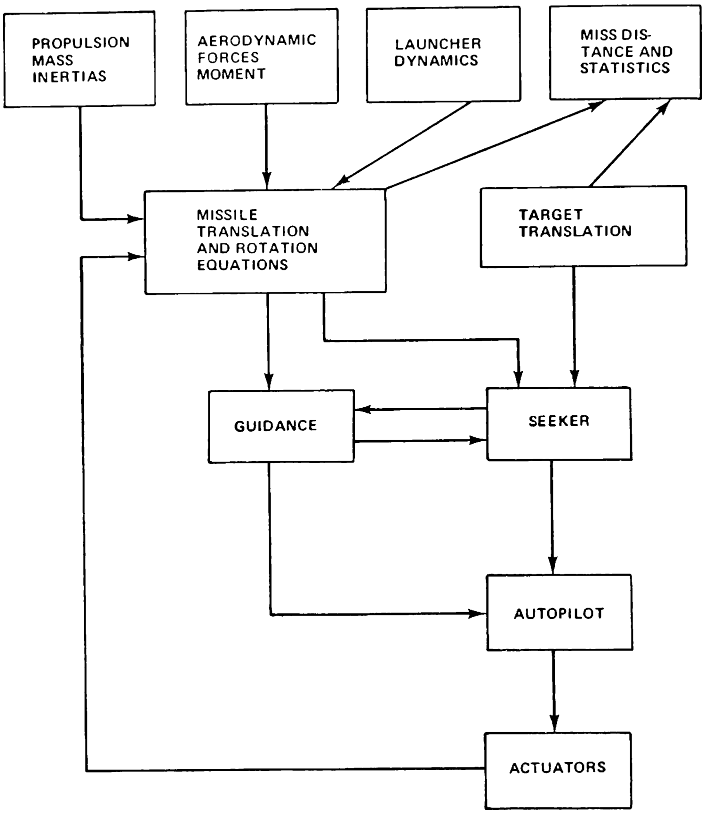
THE ADVANCED SIMULATION CENTER's HYBRID COMPUTING SYSTEM

FIGURE 5



METHODOLOGY OF SIMULATION DEVELOPMENT

FIGURE 6



SIMULATION PROGRAM MODULES

FIGURE 7