

HYBRID EXPERIMENTING SYSTEM AS AN EXTENSION OF SIMULATION LANGUAGE SIMCOS

Borut Zupančič, Matjaž Jekl, Rihard Karba

Faculty of Electrical and Computer Engineering
University of Ljubljana, Slovenia

A cost effective and efficient hybrid simulation system with the connection of analog processor EAI 2000 and IBM PC compatible computer is developing on the Faculty of Electrical and Computer Engineering, University of Ljubljana, Slovenia. The analog processor is connected to the digital computer by means of general purpose process interface PCI 20000 (Burr-Brown) with A/D, D/A and digital I/O channels. The software equipment for the hybrid system is based on the digital simulation language SIMCOS. The main idea was to expand the existing concept model-method-experiment to the concept model-method-experiment-type of simulation. So a model can be simulated with analog, digital or hybrid simulation. In the paper the mentioned hybrid simulation system with some experiments' implementations that elucidate the usefulness of the hybrid system is presented.

KEY WORDS Digital simulation, simulation language, hybrid system

1. INTRODUCTION

In spite of the very fast development of digital computers and parallel processing, true hybrid computers (the connection of analog and digital processors) represent still useful and valuable tool on particular fields and in particular applications [2]. Let us list some simulation tasks and simulated process characteristics for which hybrid simulation could bring some advantages:

- demands for high-speed simulation,
- real-time or faster simulation,
- control system design, evaluation and testing,
- man-in-the-loop simulation (direct interaction between user and simulation computer),
- hardware-in-the-loop simulation (computer connected to a pilot or a real plant),
- need for equal simulation speed for small and large models,
- simulation of complex, time variable, non-linear processes,
- presence of discontinuities (depending on states and on a time),
- highly stiff systems (large differences in time constants),
- algebraic loops in model equations.

Besides, the important field of analog computers is surely educational since their properties enable very illustrative representation of the modelling and simulation approach. Finally some principles, which originate in analog computation (e.g. scaling), can also be applied to other areas [3].

On the other hand we cannot overlook several disadvantages of hybrid simulation systems, among which the two most important are:

- extremely high price and
- inadequate (insufficient) software equipment.

So the main aim in developing the new experimenting system was to use our existing hardware and software equipment:

- analog-hybrid computer EAI 2000,
- process interface PCI 2000 Burr Brown,
- digital simulation language SIMCOS

and to develop a cost effective experimenting system that would retain the most of the mentioned advantages and dispatch the mentioned disadvantages as well. Using the mentioned equipment we were aware of the fact, that the simulation speed would be limited.

In addition the brief description of used equipment and its integration into an efficient hybrid experimenting system will be given.

2. HYBRID SIMULATION SYSTEM

The former version of our analog computer EAI 2000 enables the communication with the digital part only through the serial line RS 232. This communication enables setting and reading of parameters (model coefficients, simulation run control parameters), controlling of analog computer modes as well as reading of simulation results. Because of very limited speed (ASCII data, additional data for handshaking, transfer rate 9600 bit/s, ...) it is usually possible to use this feature only before simulation run or between two successive simulation runs but not during particular simulation run. For the latter possibility the parallel communication was implemented. Process interface PCI 20000 Burr Brown, which we use also for the implementation of digital control algorithms for the laboratory set-ups, was used. The cost effective solution of our hybrid system is shown in Figure 1.

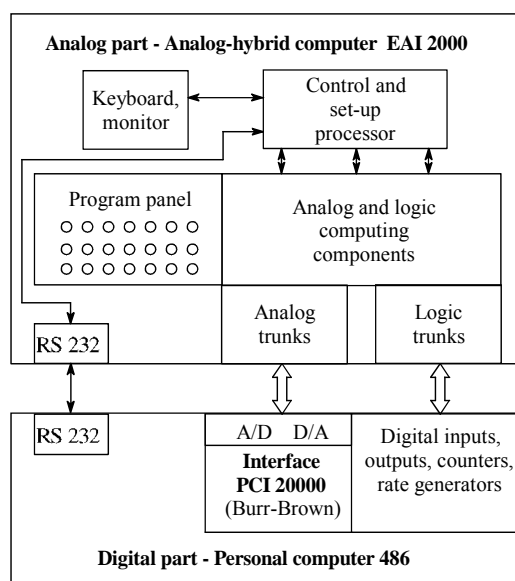


Figure 1 Hybrid system

The parallel communication enables to exchange analog (voltage) signals and digital (TTL) signals. As analog and logic trunks of analog part already generate signals $\pm 10V$ and TTL respectively, these signals can be directly fed to A/D and D/A channels and to digital inputs and outputs of the PCI 20000 process interface. Analog signals are used mainly as simulation variables' signals. On the other hand, digital signals are used primarily for the fast controlling of analog computer modes as well as for more sophisticated simulation experimentations (e.g. indication of constraints during optimization).

3. SIMULATION LANGUAGE SIMCOS

The simulation language SIMCOS [3, 5, 6, 7] is a CSSL - type equation oriented language that was developed at the Faculty of Electrical and Computer Engineering in Ljubljana. It works as a compiler. The model, which is coded in CSSL syntax or is described by a graphical block oriented simulation scheme, is processed by the compiler into FORTRAN modules and a model data base. The FORTRAN modules are further processed by a FORTRAN compiler and subsequently linked with appropriate libraries into an executable simulation program. The supervisor program automatically handles all the above procedures and is able, together with a highly interactive user interface, not only to simulate the model but also to perform simple experiments (e.g. change of model constants, output specifications, function generators' breakpoints) and complex ones (e.g. parameter studies, optimization, ...).

All basic possibilities for a model description are available: The built in nonlinearities and signals, multidimensional function generators, the procedural block, basic and advanced univariable and multivariable dynamical precompiled submodels (transfer functions, state space, lead-lag, ...) in continuous and discrete forms as well as different controllers with integral wind up protection are implemented. Using a special pre-processor a hierarchical modelling is also available. Well tested and previously developed programs for particular components can be reused as submodels in higher hierarchical levels. Because of sorting algorithm an user need not to take care about the order of a model statements. Many integration algorithms (single step, multistep, low order, high order, extrapolation, stiff, ...) cover all commonly appeared numerical problems giving the simulation tool an appropriate numerical robustness.

However several applications show, that still some additional features should be included, as:

- extension of continuous simulation concept to the discrete one by so called generalized simulation operator,
- extension of the language with hardware-in-the loop and man-in-the loop simulation possibilities implementing real time possibilities,
- extension of the simulation towards a complex experimentation system.

Generalized simulation operator

Sometimes it is very efficient to simulate or to experiment with a model in discrete or in difference equation form. Such models can be easily derived from continuous linear models using well-known discretization transformations. Especially in experiments that demand many simulation runs (e.g. optimization) a lot of time can be saved using the mentioned approach.

The generalization of the simulation was achieved by so called generalized simulation operator i.e. the integrator as the basic operator of continuous simulation was expanded to the generalized simulation operator which can be by the proper parametrization either the integrator or the discrete delay as the basic operator of discrete (difference equation) simulation. The approach is illustrated in Figure 2.

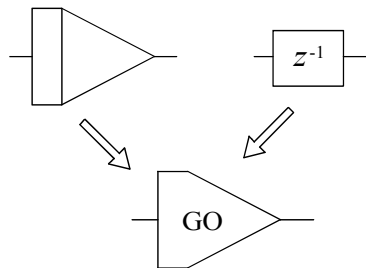


Figure 2 Generalized simulation operator (GO)

Real time possibilities

Conventional computer systems are very limited for real time applications because of insufficient hardware capabilities, redundant and too complex software equipment, inefficient programming and poor input-output facilities. As SIMCOS is general purpose simulation language which works on conventional hardware (PC) with standard operating system (MS-DOS) and cost effective process interface (PCI 20000), the simulation speed is of course limited. Using SIMCOS in hybrid configuration some other limitations occur so the speed is even more limited. But the use in slower simulations (or experimentations) is quite satisfactory.

The real time simulation was enabled by the following modifications and extensions:

- the generalized simulation algorithm was synchronized with real time,
- simple CPU low cost simulation algorithms were included (Euler integration algorithm, discrete simulation),
- the communication with real signals was enabled.

Program modules for data acquisition and transmission are specific for particular hardware equipment (PCI 20000). Because the language is opened to the user, it is not difficult to implement a specific hardware.

Experimental environment

It is well known, that simulation run is only one useful experiment in different studies. Modern tools include also more complex experiments as optimization,

linearization, parameter study etc. Some experimental implementation concepts are known (e.g. the concept model - method - experiment, [1]) but compiler orientation of SIMCOS does not give the possibility to fully implement them. For this case we introduce three experimental sections (INITIAL, TERMINAL and DYNAMIC). These sections are not the model sections as is the case in ASCL. INITIAL and TERMINAL sections are two FORTRAN modules that specifies the operations before and after a simulation run. They are included in the main simulation program during FORTRAN compilation after the experiment selection. Section DYNAMIC must be written in SIMCOS syntax and is used for the definition of operations which must be executed during the simulation in parallel with model equations but functionally do not represent a model but an experiment (e.g. a criterion function or constraints in optimization, input signals, simulation control parameters, ...). So each model is compiled for in advance selected experiment. Each higher level experiment allows also the execution of the pure simulation. The concept of described experimental environment is shown in Figure 3. An experiment is treated as an execution of a particular method (simulation, optimization, linearization, parameter study, user experiment) on a model.

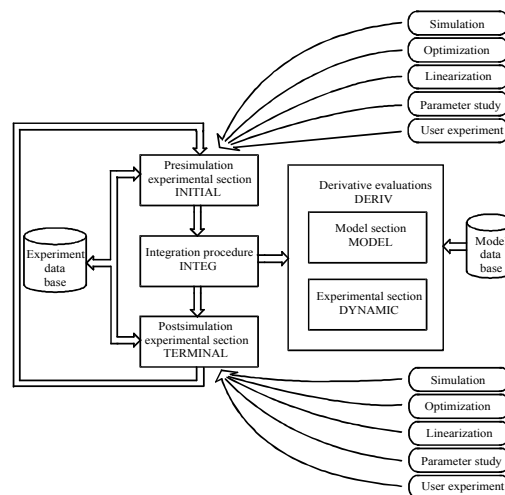


Figure 3 The concept of the experimental environment in SIMCOS

Optimization is a very useful experiment in many applications. An user should define a criterion function, constraints and parameters to be optimized. Linearization is used to obtain the linear model (in state space form) from a non-linear one. Parameter study is an experiment in which the influence of a parameter to simulation results can be efficiently studied. All these experiments are realized with pre-programmed INITIAL and TERMINAL sections. User experiment enables a user to create his own experiment.

4. CONCEPT OF HYBRID SIMULATION SYSTEM

The goal was to completely supervise the hybrid system by the digital part i.e. by a PC computer. Important modifications and expansions of the simulation language SIMCOS were implemented for this purpose. The extension with real time possibilities was already mentioned. Besides the controlling of analog computer

modes, the reading and setting of the potentiometers and the exchange of analog and digital signals represent the basic extensions.

As we intend to study digital, analog or true hybrid models with our hybrid system and to simulate each model on digital, analog or hybrid way the concept of digital simulation language SIMCOS was extended from the concept model-method-experiment [1] into the concept model-method-experiment-type of simulation [7]. Model means the descriptions of two parts: digital part with appropriate algebraic - differential equations and analog part with precise description of all used analog components and their topological connections as well. Method means an algorithm which defines, what must be done with a model. Simulation, parameter study, optimization or user defined experiment can be performed with our hybrid simulation system. Experiment means an use of a particular method on a particular model. Type of simulation, which can be analog, digital or hybrid, means a tool on which the simulation is implemented. The analog simulation is implemented on the analog computer, the digital simulation is implemented on the digital computer and the hybrid simulation is implemented with parallel operation of both computers. The concept of hybrid simulation processing is shown in Figure 4.

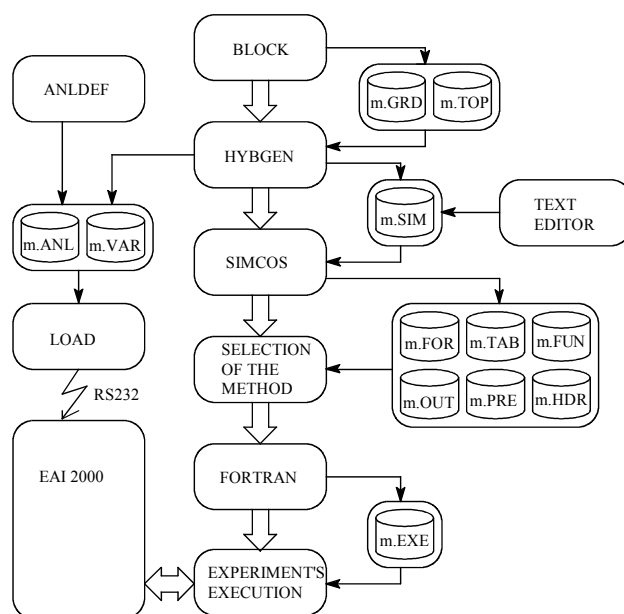


Figure 4 The concept model-method -experiment-type of simulation

All phases are automatically performed by the aid of a supervisor program. An user must select a model at the beginning. A new model can also be defined. The model is described by the graphical simulation scheme by the aid of the graphical editor BLOCK. Each icon (block) has particular attributes. Some blocks have additional attribute which defines whether they must be realized on the analog or on the digital part (e.g. integrator (INT), summer (SUM), coefficient (GAIN)). The default value of this attribute means the digital part. Analog part of the model should be of course patched on the analog computer patch panel. The graphical scheme defined in graphical editor BLOCK is of great help in patching procedure. After model description an interface program HYBGEN enables a user to specify the demands for simulation results presentation. Only the variables from the digital part can be directly observed. If a particular variable from the analog part should be observed, it must be

connected to a digital part through analog trunks and A/D channels. The simulation control parameters like the duration of initial condition, operate and hold periods, the value of the master time scale, the sampling time (frame time) of analog variables and some other control parameters of digital simulation program are also defined by the aid of HYBGEN. When exiting the HYBGEN the following files are generated: *model.SIM* which defines the digital part of the model in CSSL syntax, *model.ANL* in which the structure of analog part of the model is defined and *model.VAR* with some additional parameters for analog part. The digital and analog part can be also generated without graphical pre-processors BLOCK and HYBGEN. The digital part *model.SIM* can be defined with CSSL standard language by an arbitrary text editor. The analog part can be defined by a special program ANLDEF. The described files define also the type of simulation. Hybrid simulation demands both parts (files). Analog or digital simulation demands only one part. After model description the digital part is compiled into several FORTRAN modules and some data base files. The parameters of analog part are then transferred to the analog computer through the serial line. Then a user must select a method i. e. simulation, optimization, parametrization or a user defined method. After that the digital part is compiled into executable program. In the case of pure analog model a pre-programmed code is used for simulation. Then the experiment is executed starting the generated or pre-programmed code. Model variables can be shown on the screen or recorded in the file.

Hybrid simulation

The flow chart of hybrid simulation is shown in Figure 5.

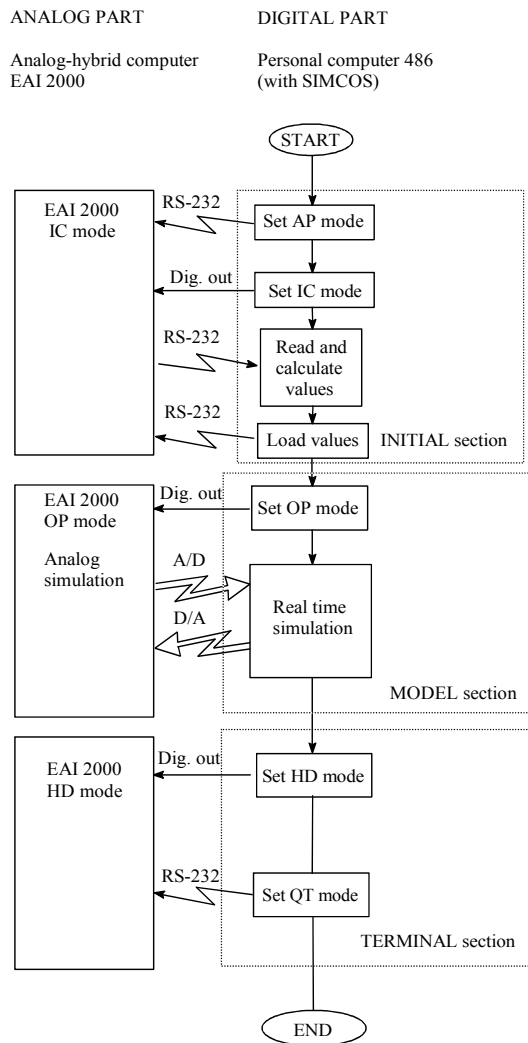


Figure 5 Hybrid simulation

At the beginning (in the INITIAL section) the analog computer is set into AP mode through serial line. This mode enables to use logic lines to adjust the analog computer modes. The analog computer is then set into IC (initial condition) mode. According to the defined duration of simulation run in the simulation program, the appropriate time constants like master time scale and local time scales for integrators are loaded into the analog computer, what enables the synchronized work of analog and digital part of hybrid simulation system. At the beginning of the first SIMCOS communication interval (time=0) the analog computer is set into OP (integration) mode. During simulation both parts exchange data through PCI 20000 interface lines. After the simulation run (time=time_final) the analog computer is set into HD (hold) mode and then into QT (stand by) mode. If model variables are stored in a file during simulation, the achieved frame time is app. 50 ms. However it can be significantly shortened if model variables are written in RAM.

Hybrid optimization

The parameters, that should be optimized, are implemented on the analog computer with digitally set coefficients. It is recommended to realize the whole simulation model on the analog part. A criterion function and constrains can be realized on the

analog or on the digital part. The flow chart of hybrid optimization is shown in Figure 6.

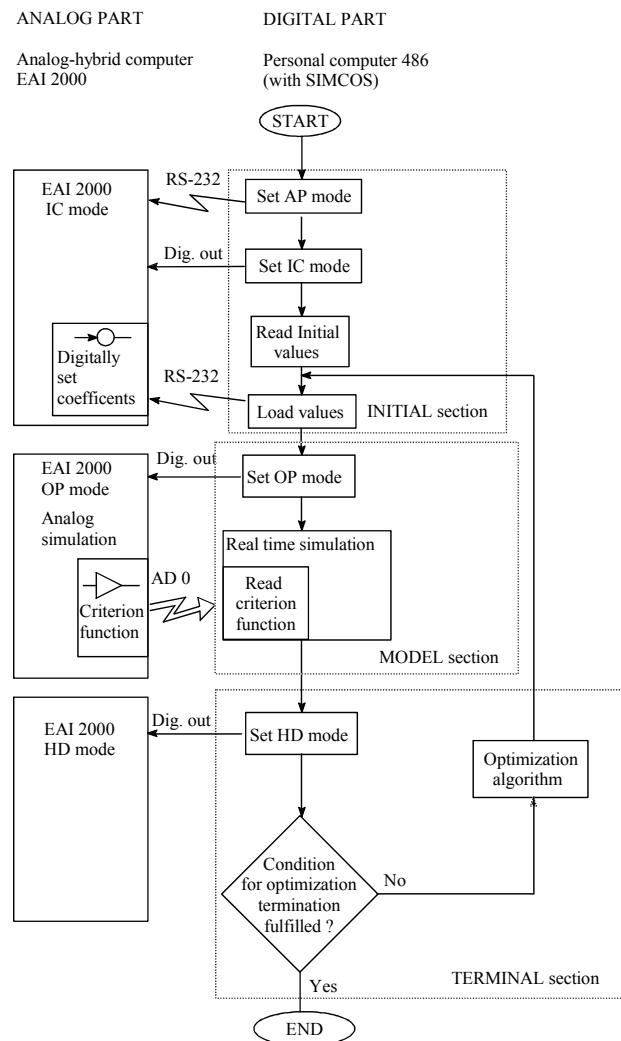


Figure 6 Hybrid optimization

At the beginning (in the INITIAL section) the analog computer is set into AP and then through logic lines into IC mode. The digitally set coefficients are loaded by initial values of parameters that should be optimized. The transfer to the MODEL section defines the initial simulation time (time=0) when also the analog computer is set into OP mode. During simulation run the criterion function is read. After simulation run (time=time_final) (in TERMINAL section) the analog computer is set into HD mode. Then the conditions for optimization termination are tested. If no one of the conditions is fulfilled, the new optimal parameters are evaluated by an optimization method. These values are loaded into digitally set coefficients after returning into INITIAL section. This procedure is repeated until one of the optimization termination conditions is accomplished.

Hybrid parametrization

From the implementation view this method is similar (and easier) to the optimization one. Only one parameter, which is again realized by digitally set coefficient, is swept during several simulation runs.

5. CONCLUSIONS

All parts of the described concept except HYBGEN interface have been already realized. With the hybrid system an efficient simulation system, used above all in the educational process, was obtained. The studies with hardware-in-the loop and man-in-the loop can be efficiently performed. Some extensions e.g. automatic scaling would additionally improve the applicability of the system.

References

- [1] Breitenecker, F., D. Solar: Models, methods and experiments - modern aspects of simulation languages, *Proceedings of the 2nd European Simulation Congress*, Antwerpen, Belgium, 1986, pp. 195-99.
- [2] Breitenecker, F.: Need for Hybrid Simulation, *Proceedings of the 3rd European Simulation Congress*, Edinburgh, Scotland, 1989, pp. 421-25.
- [3] Matko, D., B. Zupančič, R. Karba: *Simulation and Modelling of Continuous Systems - A Case Study Approach*, Prentice Hall Int., London, UK, 1992.
- [4] Jekl, M.: *Synthesis of the hybrid simulation system for the simulation and optimization of control systems*, MSc. Thesis, Faculty of Electrical and Computer Engineering, Ljubljana, 1993 (in Slovene).
- [5] Jekl, M., B. Zupančič, R. Karba: Hybrid simulation system based on digital simulation language SIMCOS, *Proceedings of the 8th symposium Simulationstechnik*, Berlin, Germany, 1993, pp. 171-74
- [6] Zupančič, B., D. Matko, R. Karba, M. Šega: SIMCOS - digital simulation language with hybrid capabilities, *Proceedings of the 4th symposium Simulationstechnik*, Zürich, Switzerland, 1987, pp 205-12.
- [7] Zupančič, B., D. Matko, R. Karba, S. Divjak, M. Jekl: The Concept of a new Hybrid Experimenting System, *Proceedings of the 13th IMACS World Congress on Computation and Applied Mathematics*, Dublin, Ireland, 1991, pp. 1318-19.