EGG-M-88524
CONF-890173--4

RELAP5 Desktop Analyzer*

R. J. Beelman
W. H. Grush
G. A. Mortensen
D. M. Snider
K. L. Wagner

Idaho National Engineering Laboratory
EG&G Idaho, Inc.
P.O. Box 1625
Idaho Falls, Idaho
83415

ABSTRACT

The previously mainframe bound RELAP5 reactor safety computer code has been installed on a microcomputer. A simple color-graphic display driver has been developed to enable the user to view the code results as the calculation advances. In order to facilitate future interactive desktop applications, the Nuclear Plant Analyzer (NPA), also previously mainframe bound, is being redesigned to encompass workstation applications. The marriage of RELAP5 simulation capabilities with NPA interactive graphics on a desktop workstation promises to revolutionize reactor safety analysis methodology.

INTRODUCTION

The arrival of powerful 32-bit microprocessors and inexpensive semiconductor memory have made it feasible to run the advanced reactor safety codes on desktop computers. Although the speed of execution is currently an order of magnitude slower than mainframe computation, there are times when this performance is not only acceptable, but preferable. The principal advantage is the user can run the code on his desktop at his own discretion and at no mainframe computational expense. Towards this end RELAP5/MOD2 [1] has been extensively modified to run on the Definicon 780+ coprocessor board for the IBM PC/AT [2,3] as proof of concept. Desktop RELAP5 was first successfully demonstrated at the Fifteenth Water Reactor Safety Information Meeting (WRSIM) in October 1987 [4]. The calculational results were graphically displayed as they were produced via a special purpose HALO graphics driver to facilitate analysis of results and alleviate output mass storage limitations. The data could also be repeatedly replayed through the driver.

Efforts are now underway to develop a portable version of RELAP5 applicable to microcomputers as well as supercomputers. In a parallel effort the Nuclear Plant Analyzer (NPA) [5], also previously mainframe bound, is currently being rewritten to encompass workstation applications. This paper is an account of developmental efforts to integrate RELAP5 engineering simulator capabilities with online NPA graphical analysis techniques to produce a cost-effective, general purpose RELAP5 desktop analyzer.

BACKGROUND AND MOTIVATION

Historically, the productivity of a numerical reactor safety analyst has been hampered by several factors: computer turnaround for problem setup and checkout; CPU allocation, accessibility and availability for problem initialization and transient simulation; job throughput; lost output; and validation of numerical results. Within the constraints of budget and schedule these restrictions have traditionally left little margin for error or allowance for further investigation when funds are limited or delays are encountered. Yet, ironically, the analyst most often finds himself waiting for CPU access, job completion, or printout.

Furthermore, as the complexity and severity of reactor transient analyses accomplished via computer simulation have escalated, the tools available to the analyst to decipher the results have remained relatively unchanged. The analyst today still relies primarily on the printed numerical output of a code run to discern the nuances of the calculation and accomplish the analysis. Much of his time is still spent paging through mounds of printed output or producing x-y plots trying to interpret the numerical results, while also trying to objectively validate the results.

*Work supported by the U.S. Nuclear Regulatory Commission, Office of Nuclear Regulatory Research, under DOE Contract No. DE-AC07-76ID01570.

Clearly, an economical engineering workstation capable of running the reactor safety codes interactively, and of simultaneously displaying the results in a coherent graphical fashion as they are produced, would alleviate many of these concerns. It would also reduce analysis costs and enhance productivity. Although not yet capable of real time simulation, this desktop analyzer could be used for input model development and validation, problem setup and initialization, scoping analyses, interactive operator guideline evaluation, drill scenario development, simulator qualification playback of mainframe calculations, etc. It could also serve as a word processor, file server, editing and plotting terminal.

RELAP5 ENGINEERING SIMULATOR CAPABILITY

The Reactor Excursion and Leak Analysis Program (RELAP) is a large thermal-hydraulic computer code, roughly 80,000 Fortran source statements, that has long been used by government and industry to analyze transient behavior of pressurized water cooled nuclear reactors (PWRs). The code models multiphase flow of fluids in piping networks; heat transfer between the fluid, the fuel, heat exchangers and structural components; neutronic feedback effects between the fuel and the fluid in the reactor core; and control system interactions with system components. RELAP5 is the fifth generation of this long-standing code development effort. It has been extensively assessed and applied by numerous companies and government agencies throughout the United States and abroad in evaluating the safety of nuclear power plants over the range of plant conditions encountered during hypothetical and actual accident scenarios.

RELAP5/MOD2, employing six phasic conservation equations, solves coupled transient two-phase hydrodynamic, multi-region heat conduction and space-independent neutronic equation sets formulated on a one-dimensional, two-velocity, nonequilibrium flow model. The hydrodynamic model also transports and accounts for the effects of noncondensibles and boron solute. The constitutive package uses mechanistic correlations for interphase mass, energy and momentum transfer, convective heat transfer, wall drag, form induced head loss and hydrodynamic choking.

The code provides process component input models for modelling piping networks, plenums, pumps, valves, tees, turbines, accumulators, steam separators, point reactor kinetics, control systems, actuation signals and structural component mass. The code also allows online interactive manipulation of process component models to the extent provided during input model development.

RELAP5/MOD2 was originally designed to run on mainframe computers because of speed, precision, memory and mass storage requirements. The code was initially developed on the CDC 176 computer at the Idaho National Engineering Laboratory (INEL) and incorporated many design features unique to that machine to enhance runtime performance. The CDC 176 was extremely fast for its day but had limited core memory. Therefore, the RELAP5 code was designed with elaborate memory overlays and extensive word packing in order to conserve memory. These and other techniques allowed RELAP5 to run large problems in an acceptable timeframe on a CDC 176 computer. The code has since been installed on CRAY and IBM mainframes and a Definicon coprocessor board installed in an IBM PC/AT.

DEFINICON CONVERSION

The Definicon 780+ is an IBM PC XT/AT compatible coprocessor board equipped with a 20 MHz Motorola 68020 central processing unit (CPU), a 20 MHz Motorola 68881 math processing unit (MPU) and up to 16 Megabytes of fast dynamic random access memory (DRAM). This board, with 4 Mbytes of DRAM, has been configured in an IBM PC/AT equipped with 100 Mbytes mass storage, a streaming tape drive for file archival/backup purposes and an Ethernet board for transferring files to and from the mainframe. The requisite software includes a Silicon Valley Systems (SVS) Fortran 77 compiler, Definicon's own loader, linker, symbolic debugger and assembler, and Mortice Kern Systems' (MKS) UNIX toolkit. The board and the software are installed and operated under the resident MS-DOS operating system.

Since the Definicon 68020 CPU uses 32-bit words, whereas the CDC 176 CPU uses 60-bit words, a problem immediately arose when integer variables were equivalenced to real variables which were declared REAL*8, i.e., double precision 64-bit words, on the Definicon in order to attain the necessary precision. Since RELAP5 was developed on a CDC 176 computer with 60-bit floating point precision, the coding made no distinction between the memory space occupied by single precision integer or real variables. In the Definicon version, differences in word size between integer and REAL*8 variables were accounted for by double dimensioning the integer variables that were equivalenced to real variables, i.e., item(i) becomes item(2,i), such that equivalenced integers referenced the bottom 32 bits of the 64-bit REAL*8 word.

The majority of the changes necessitated in the Definicon conversion consisted of accommodating the AND, OR, XOR, SHIFT, MASK, and NOT constructions that were used extensively in the CDC 176 coding to accomplish bit manipulations in the packed 60-bit words.

These constructions were converted to corresponding function calls, e.g., DAND, DOR, DXOR, DSHFT and DNOT. In addition, the extra four bits encountered in converting CDC shifts from 60-bit words to the 64-bit words were accounted for by adding four to the shift count. The CDC MASK function was replaced by a reference to a one-dimensional array that contained the bit mask generated by the CDC compiler.

These 60-bit manipulation routines were first written in Fortran by invoking the MIL-STD-1753 bit manipulation routines, e.g. IAND, IOR, IBITS, IBSET, IBCLR, MVBITS, etc. Most of these routines were later converted from Fortran to 68020 assembly language to speed up execution.

Numerous other problems were encountered justifying and reinforcing efforts to develop a portable version of RELAP5 applicable to microcomputers as well as supercomputers. Most of these problems are alleviated by the forthcoming release of RELAP5/MOD3. MOD3 adheres to the ANSI Fortran 77 standard and avoids machine dependent coding. Preliminary results with the unpacked developmental version of MOD3 also indicate significantly improved runtime performance on the Definicon 780+.

RUNTIME COMPARISONS

Benchmark calculations have been run using the desktop RELAP5 code on the Masscomp 5600 and Definicon 780+. Some timing comparisons with the CDC 176 and other pertinent information are presented for each of these cases. With the exception of the final case these test cases are distributed with the code transmittal.

Edwards' Pipe Problem

The Edwards' pipe problem was one of the first well instrumented, two-phase blowdown experiments. Conducted in England in the late 1960s by Anthony Edwards, this experiment has been used ever since for assessing thermal-hydraulic computer codes. The experimental test apparatus consisted of a 4 m length of 38 cm diameter pipe filled with water at 7 MPa and 502 K. The water is exhausted out the end of the pipe when a diaphragm is ruptured. This case tests the hydrodynamic coding and includes pipes, single junctions, and time dependent volumes.

The RELAP5 model consists of 20 control volumes with the rupture diaphragm modeled as a single junction connected to a time dependent volume. The transient is run out to 0.5 seconds. The Edwards' pipe case results are presented in the following table.

RELAP5 EDWARDS' PIPE CASE RESULTS

Computer	Clock (MHz)	Compiler	Runtime (CP sec)	Ratio
CDC 176	36.	FTN5 Opt2	18.6	1.0
Masscomp 5600	16.7	F77 Opt3	495.	27.
Definicon 780+	20.	SVS F77	382.	21.

Typical PWR Case

The typical PWR case represents a four loop Westinghouse type PWR in which a four inch pipe ruptures in one of the loops. The RELAP5 model consists of 100 fluid volumes describing the primary and steam generator secondaries in which the pressurizer loop and the two unaffected loops are lumped into a single loop and the affected loop is modelled separately. The model also includes numerous heat structures, representative control system modelling, homologous reactor coolant pump (RCP) treatment, emergency core coolant (ECC) system modelling, and time dependent feedwater and main steam line boundary conditions modelling. It is a sample problem. The results for a 30 s small break loss of coolant accident (SBLOCA) calculation are shown in the following table.

RELAP5 TYPICAL PWR CASE RESULTS

Computer	Clock (MHz)	Compiler	Runtime (CP sec)	Ratio
CDC 176	36.	FTN5 Opt2	219.	1.0
Masscomp 5600	16.7	F77 Opt0	10651.	49.
Definicon 780+	20.	SVS F77	7260.	33.

The timing ratio for the initialization phase of this problem was over 60 for the Definicon 780+. If we adjust the runtime ratio to account for the latency of the initialization phase, the actual transient runtime ratio is about 25 times slower than the CDC 176.

Real Plant Problem

Calvert Cliffs is a two unit Combustion Engineering (CE) type nuclear generating station located in Maryland 45 miles southeast of Washington, D.C., on Chesapeake Bay. The RELAP5 Calvert Cliffs 1 input model has been used with the Nuclear Plant Analyzer (NPA) on the CDC 176 to conduct emergency response preparedness training exercises at the United States Nuclear Regulatory Commission (USNRC) [6]. It was also the example problem demonstrated at the unveiling of Desktop RELAP5 at the 15th WRSIM. This input model is a full featured scoping analysis model consisting of 100 hydrodynamic volumes with complete nuclear steam supply system (NSSS) description and full balance of plant modelling with the exception of main turbine modelling. Although the nodalization scheme is coarser than the parent licensing audit model, it has been optimized to be fast running without sacrificing predictive capability. It retains full interfacing control system modelling, homologous RCP and feedwater pump treatment, point kinetics feedback modelling, engineered safety features modelling and structural mass distributed stored energy modelling. It is a full blown real plant problem. After the initialization phase, this deck runs close to real time on the CDC 176 computer.

The results for a loss of feedwater accident (LOFA) are presented in the following table at problem initialization and 300 seconds transient advancement. Once again the Definicon initialization time was disproportionately longer than the transient advancement runtime compared to the CDC. The runtime ratios shown have been adjusted by subtracting the initialization time from the elapsed CPU time in order to account for the latency of the initialization phase. These corrected transient runtime ratios are more consistent with the previous RELAP5 cases.

RELAP5 CALVERT CLIFFS LOFA CASE RESULTS

Computer	Clock (MHz)	Compiler	Runtime (CP sec)	Ratio
CDC 176	36.	FTN5 Opt2		
(Initialize)			17.9	0.0
(Run 300 s)			374.9	1.0
Definicon 780+	20.	SVS F77		
(Initialize)			1739.	0.0
(Run 300 s)			9172.	20.8

The comparison results indicate that the Definicon version of RELAP5/MOD2 currently runs 20-30 times slower than the CDC 176 version. This means that a problem which would normally require 30 minutes of execution time on a CDC 176 computer could be run overnight on the microcomputer. Since a 30 minute RELAP5 run on the CDC 176 would normally be run with overnight turnaround anyway, the analyst would be able to review the results the next morning whether it was run on the mainframe or the microcomputer. The effective elapsed wall clock time to complete the calculation is therefore the same.

NUCLEAR PLANT ANALYZER

The Nuclear Plant Analyzer (NPA) is a computer graphics based reactor safety analysis tool developed at the INEL to assist an analyst in simulating and evaluating the transient response of nuclear power plants [7]. The NPA integrates reactor safety codes with interactive graphics routines allowing interactive code execution and online analysis of resulting data or replay of past simulations by means of graphic displays. The replay capability allows the user to review the results of a previous calculation by redisplaying the data at real time, or at a user selectable rate faster or slower than real time, and by allowing him to move forward or backward in the calculation at will. Both modes allow online plotting of parameters.

The NPA was also originally developed on a CDC 176 and incorporated machine specific coding to enhance its performance. In the absence of a hierarchical file system a huge relational database was developed to contain the elements of the graphic display system, provide for a central data repository, and allow database editing and maintenance. Access was accomplished via network communications through dual computer terminals controlling separate color graphics and simulation tasks requiring a master-slave interprocess communications protocol on a CDC 176. Data flow-control communications and RELAP5 interactive execution were accomplished through formatted page input prompting and menu option selection on the master ADDS Viewpoint 60 console while NPA color graphics was displayed on the slave Tektronix 4125 graphics display terminal. All processing was accomplished on the CDC 176 where the NPA resided.

NPA interactive simulation was expensive because it required dedicated mainframe allocation at interactive access rates to be effective. Replays could be accomplished at a fraction of the cost because interactive RELAP5 execution was not required. Finances, not efficiency or productivity, therefore dictated the mode in which the NPA was used. However, despite the cost, NPA interactive simulation has been used extensively to aid understanding of transient phenomena in applications where the need was immediate or simulated operator intervention was required [8].

RELAP5 DESKTOP ANALYZER

With the introduction of coprocessor boards that fit in an IBM PC, it appeared that an entry level NPA could be built using a Definicon 780+, an enhanced graphics adapter (EGA) card and an EGA color monitor. After a rather lengthy debugging process, RELAP5 finally successfully executed on the Definicon 780+ coprocessor board in the IBM PC/AT. A specialized graphics display processor was developed so that the output of a RELAP5 run could be viewed graphically as the code was running. Since the EGA card allows a maximum of only 16 colors and a display resolution of only 640 x 256, a high fidelity display was not possible. Nevertheless a primitive display was produced to view the results of the code calculation as it advanced. In this capacity the Definicon 780+ CPU ran the RELAP5 code in the background and the host PC CPU ran the graphics display processor in the foreground. Since MS-DOS is not a multitasking operating system, this feat was accomplished using a special background loader supplied by Definicon. The display processor can also repeatedly playback the results of a previous RELAP5 calculation in the replay mode.

A higher resolution graphics board and monitor would be needed in order to obtain the functionality and fidelity of the current Tektronix 4125 NPA graphics displays. One possibility was the Definicon GVR-1024 graphics card consisting of an Intel 82786 graphics processor on a board which piggybacks to the 780+ CPU board via the unused 132-pin paged memory management unit (PMMU) socket. This graphics board provides 1024 x 768 resolution with 256 choices from 16 million colors. In this configuration the host 80286 CPU would perform executive functions and control disk operations, the Definicon 780+ CPU would run RELAP5, and the Definicon GVR-1024 CPU would drive the graphics display. It would result in an inexpensive but impressive RELAP5 desktop analyzer. However, the new class of 32-bit UNIX windowing workstations provides an even more promising integrated platform with many additional hardware and software features.

To this end the NPA software is currently being rewritten to take advantage of the UNIX windowing workstation environment. The new NPA design will adhere to established industry-wide standards and protocols. It will be an integrated tool allowing graphic display development, automated driver production, and interactive code execution via a user-friendly graphical interface using pop-up menus and a mouse.

SUMMARY AND CONCLUSIONS

The ability to interactively execute RELAP5 and simultaneously analyze the results on a desktop workstation heralds a new era of desktop reactor safety analysis capabilities. Dependence on mainframe computers to accomplish input model development and problem setup, checkout and initialization has been reduced. As CPU speed and code efficiency increase and real time desktop simulation becomes a reality, a natural migration path from the mainframe to the desktop will exist for all code applications. For the present, desktop applications such as scoping analyses, code development and assessment, emergency procedure development and validation, drill scenario development, simulator qualification, and playback of mainframe calculations are already possible. Eventually, all but the most stringent safety related applications will be performed on the desktop at the user's discretion and at no mainframe computational expense. In the interim interactive RELAP5 execution on the mainframe with online graphical presentation of results on the workstation, or playback of batch mainframe calculations, will continue to greatly enhance comprehension of results, increase analyst productivity and dramatically reduce analysis costs.

ACKNOWLEDGMENTS

The authors wish to gratefully acknowledge the technical contributions, assistance and encouragement of Mr. Dick Wagner, RELAP5 code architect, in converting RELAP5 to a desktop computer application. We also wish to thank Dr. Don Solberg of the USNRC for his support of the NPA redesign effort.

REFERENCES

1. V. H. Ransom, et al., RELAP5/MOD2 Code Manual, NUREG/CR-4312, EGG-2396, August 1985, EG&G Idaho, Inc., Idaho Falls, Idaho.

2. "The Definicon 68020 Coprocessor", BYTE Magazine, July and August, 1986.

3. Definicon Systems, Inc., 1100 Business Center Circle, Newbury Park, CA 91320.

4. "RELAP5 Desktop Workstation", R. J. Beelman, et al, Transactions of the Fifteenth Water Reactor Safety Information Meeting, October 1987, U.S. Government Printing Office, Washington, D.C.

5. E. T. Laats, et al, User's Manual for the U.S. Nuclear Regulatory Commission's Nuclear Plant Analyzer, EGG-RTH-7515, February 1987, EG&G Idaho, Inc., Idaho Falls, Idaho.

6. "Engineering Simulator Applications to Emergency Response Training", R. J. Beelman, M. A. King and F. S. Jaggar, to be published, 1989 Eastern Simulation Multiconference, Tampa, FL, March 1989, Society for Computer Simulation, San Diego, CA.

7. "Nuclear Plant Simulation Using the Nuclear Plant Analyzer," R. J. Beelman, E. T. Laats and R. J. Wagner, Proceedings Of The International Conference On Power Plant Simulation, Cuernevaca, Mexico, November 1984.

8. "Interactive Operator Guideline Evaluation Using RELAP5", R. J. Beelman and J. N. Curtis, Transactions of the American Nuclear Society, 45, 1983 ANS Winter Meeting, San Francisco, CA, November 1983, American Nuclear Society, La Grange Park, IL.