CONF-9009108--2

DE91 006591

SOME NEUTRONICS AND THERMAL-HYDRAULICS CODES FOR REACTOR ANALYSIS USING PERSONAL COMPUTERS*

W. L. Woodruff
Argonne National Laboratory
Argonne, Illinois, USA

To be presented at the

XIII International Meeting
on Reduced Enrichment
for Research and Test Reactors

23-27 September 1990
Newport, Rhode Island

*Work supported by the U. S. Department of Energy under Contract No. W-31-109-ENG-38.

SOME NEUTRONICS AND THERMAL-HYDRAULICS CODES FOR REACTOR ANALYSIS USING PERSONAL COMPUTERS

W. L. Woodruff
Argonne National Laboratory
Argonne, Illinois, U.S.A.

ABSTRACT

Some neutronics and thermal-hydraulics codes formerly available only for main frame computers may now be run on personal computers. Brief descriptions of the codes are provided. Running times for some of the codes are compared for an assortment of personal and main frame computers. With some limitations in detail, personal computer versions of the codes can be used to solve many problems of interest in reactor analyses at very modest costs.

INTRODUCTION

It has been recognized that many reactor operators have limited computational capabilities in terms of available codes and access to main frame computers. The original collection of codes for main frame computers was distributed to various U.S. reactor operators and some foreign countries that have joint agreements with the RERTR program. These codes have also been used for several IAEA sponsored training courses; at first only on main frame computers, then both on a main frame and as a backup in personal computer (PC) versions at the Australian course, and finally exclusively as PC versions at the Argonne National Laboratory (ANL) course. The PC versions of the codes will be addressed in this paper. A brief history and description of the codes in this package are provided in the next section. Some timing comparisons are presented for various generations of PCs. Finally, the advantages and limitations of the codes are discussed.

DESCRIPTION OF THE CODES

This collection of codes consists of the LEOPARD/LINX/UM2DB package for neutronics applications, the codes NATCON and PLTEMP for steady-state thermal-hydraulics analysis, and the PARET code for both steady-state thermal-hydraulics and transient analysis. The LEOPARD/LINX/UM2DB package was adapted for research reactor applications by the University of Michigan for the analysis of the first LEU demonstration core[1] and upgraded under the RERTR program at ANL. The NATCON code was developed at ANL for the analysis of plate type reactors cooled by natural convection. The PLTEMP code was developed by Mishima[2] while at ANL as part of the Japanese and ANL/RERTR program joint study, and later the code was upgraded at ANL. The PARET code was adapted for research reactor applications at ANL. The PC versions of these codes were generated at the Australian Nuclear Science and Technology Organization[3] as a backup option for an IAEA training course. The codes have now been upgraded at ANL to match their current main frame counter parts.

The LEOPARD[4] code provides burnup-dependent cross-section data in 2 and 4 energy groups based on ENDF/B-IV data. This non-dimensional lattice code supports a simple unit/super cell in either pin or plate geometry, and uses the MUFT/SUFOCATE scheme to produce a spectrum for the collapse of the 172 group thermal/54 group epi-thermal library data to broad group data. The burnup dependence is provided by limited but adequate decay chains with fitted correlations for fission products that now have been modified for research reactor applications. The binary file of burnup-dependent cross-section data produced by LEOPARD interfaces with the LINX code.

The LINX code provides a link between the LEOPARD and the UM2DB and UM3DB codes by processing the binary burnup-dependent data produced by LEOPARD into a binary library in either 2 or 4 energy groups. This library can then be read by either UM2DB or UM3DB.

The UM2DB (UM3DB) code[5] provides a 2 (3) dimensional finite difference solution of the neutron diffusion equations with depletion and shuffling of the fuel with burnup. The code reads burnup-dependent data generated by the LINX code from a binary file, while non-depleting cross-section data are input as card image data. The finite difference solution in the code is based on center of mesh differencing, but the code has now been modified to provide edge of mesh extrapolation for power peaking estimates. The code now also includes modifications to support internal boundary conditions for control rod modeling, etc. UM3DB has not been implemented for the PC at ANL.

The NATCON code[6] provides a capability for the analysis of the thermal-hydraulics of plate-type research reactors cooled by natural convection. The code includes steady-state estimates of what are often considered safety margins. These estimates may include power peaking factors, hot channel factor components, and laminar flow entrance effects. The code now includes only light water properties, but heavy water properties could easily be substituted.

The PLTEMP code[2] was developed for the thermal-hydraulic analyses of the Japanese KUHFR and KUR reactors. The code can model the entire active core including bypass flow, a single element, or now, after modifications at ANL, a single plate with it adjacent flow channels. The code provides steady-state estimates for departure from nucleate boiling (DNB) based on a selection of correlations. The code allows the application of separate hot channel components, radial power peaking factors, and an axial peaking factor. The ANL version of the code now includes an option for imposing an axial distribution on each channel and providing estimates for the heat flux and temperatures for selected channels. The code is well suited to providing steady-state estimates for safety margins in plate-type research reactors with forced flow cooling. The ANL version of the code now shares the use of the PARET code properties libraries.

The PARET code[7] provides a steady-state capability for thermal-hydraulic analysis, and a coupled point kinetics, hydodynamics, and heat transfer capability for transient analysis. The code was originally written for the analyses of the SPERT III series of reactivity insertion transients which included pin-type geometry and pressures more typical of power reactors. The ANL version of the code now contains correlations more appropiate for research reactors, and the properties library now has revised data to cover a range of pressures more typical of research reactors. Properties libraries may be generated for either light water or heavy water applications. Results from the code have been compared to the experimental results from the SPERT I transients[8] and the heavy water SPERT II transients[9] with a reasonably high level of agreement. In addition to the analysis of reactivity insertion transients, the code is capable of analyzing loss of flow transients including flow reversal and the establishment of natural convection[10,11].

The ANL version of these codes have been compiled with a Microsoft Fortran compiler (version 4.1) with a math co-processor under DOS 3.2. It was assumed that the PC would have 640 Kbytes of memory, but the dimensions on UM2DB were reduced to allow the executable code to fit on a 360 K floppy disk for transportability. The other codes have fixed dimensions and fall below this 360 K limit.

PC AND MAIN FRAME COMPARISONS

Computing times have been determined for some of the codes in this collection for an assortment of PCs and main frame computers. These should serve as a reference for other computers. The main frame computers in this comparison are a VAX 8700, IBM 3033, and IBM 3084, and the PCs included are IBM-XT, IBM-AT, and IBM PS/2 model 70. Some of the PC characteristics are provided in Table I. These IBM PC models were chosen as representative but not the fastest available. Later models of the IBM-AT have clock speeds of 8 MHz, and the IBM PS/2 is available in 20 and 25 MHz models. Most IBM compatibles are driven at clock speeds higher than those quoted in Table I. A comparison with an IBM-AT compatible with a clock speed of 8 MHz is provided in a later table.

Table I. IBM Personal Computer Characteristics

Type	CPU/Co-processor	Clock Speed, MHz	Word Length, bytes
XT	8088/8087	4.77	8
AT	80286/80287	6.0	16
PS/2	80386/80387	16.0	32

Since the codes LINX, NATCON, and PLTEMP do not have timing routines included, computing times for these codes will not be included in this comparison. It suffices to say that running times for these codes are relatively short even on the slowest PC. Table II provides a comparison of CPU times for the various PCs and codes. The IBM-AT is about a factor of two times faster than the IBM-XT due largely to word length differences. The IBM PS/2 is about 3.5 to over 4.6 times faster than the IBM-AT due to differences in both word length and clock speed. The CPU times quoted in Table II for both LEOPARD and UM2DB are per burn step. The full LEOPARD run with 8 burn steps ran 9.25 minutes on the IBM-AT. The same case on the XT stopped with a run time error after four burn steps. The PARET loss of flow case also failed on the XT with a run time error.

Table II. PC CPU Timing Comparison

Code Case	CPU Running Time, minutes IBM-XT	IBM-AT	IBM PS/2
LEOPARD[a]	2.40	1.17	0.342
UM2DB[a]			
Test Problem	4.93	2.30	0.660
Benchmark	2.89	1.47	0.567
PARET			
Benchmark $1.5	23.05	14.08	3.16
Loss of Flow	---	176.5	37.72

[a] CPU time per burn step

It may be of some interest to compare the PC run times to typical run times on main frame computers. Table III provides a comparison of three main frame computers with the IBM-AT running times for the PARET cases in Table II. The VAX 8700 runs about 38 times faster than the IBM-AT. The IBM 3033 runs about 1.2 times faster than the VAX. The IBM 3084 ran about 1.6 times faster than the 3033 machine and more than 77 times faster than the IBM-AT.

Table III. CPU Running Times for Main Frame Computers and PC

Code Case	VAX 8700	CPU Run Time, minutes IBM 3033	IBM 3084	IBM-AT
PARET				
Benchmark $1.5	0.38	0.31	0.19	14.08
Loss of Flow	4.61	3.66	2.27	176.5

Typically IBM compatibles run as fast or more often faster than the true IBM PC. Table IV provides a comparison of an IBM-AT at 6 MHz and an IBM compatible running at 8 MHz. The CPU running time for the compatible is just a little better than the 8/6 difference in clock speeds.

Table IV. Comparison of IBM-AT and IBM Compatible

Code Case	CPU Running Time, minutes IBM-AT at 6 MHz	IBM Compatible at 8 MHz
PARET		
Benchmark $1.5	14.08	10.45
Loss of Flow	176.5	130.8

It is also important that the codes give reasonably accurate answers to the problem assigned. The test problem for UM2DB is the same problem referenced in the 2DB User's Manual[5], and the eigenvalue that one should expect to compute is given as 0.986701. The IBM PCs each give the same eigenvalue for each iteration and converge to a value of 0.9867171. The VAX 8700 converges to a value of 0.9867166, the IBM main frames each give a value of 0.9867009. The convergence criterion on the eigenvalue in each case was 0.000001. While the agreement is quit good in each case, the IBM PCs agree better with the VAX main frame than the IBM main frame values, but the IBM main frame values agree best with the predicted value in the manual. The other PC results agree well with the main frame results, and there appears to be no significant loss of accuracy in using the PC version of any of the codes.

PC CODE LIMITATIONS

In addition to the obvious slower running times on a PC, some of the codes have further limitations. Some of the limitations may be machine dependent, some may be code dependent, and some may be dependent on the compiler used to generate the executable code (including the math co-processor). The long running time for some problems may be a real limitation.

One limitation common to all of the codes is the form of the output from the codes. The output file from each code is still set up for printing on a line printer not for display on the screen of a PC. Each line of output may extend over the full 132 columns allowed by most line printer plus a carriage control character in some cases. Most PC displays are limited to 80 characters per line, and some editors or word processors may either truncate longer lines or display the line in wrap around mode. A simple file lister is available that displays about 80 columns without wrap around and allows the user to shift the viewing window to the right (and back left) to view the remaining columns of data. This file lister also allows the user to search on selected key words or phrases in the output. Most of the codes in this package display some useful information on the screen as the code is executing on the PC.

The current versions of the PC codes are machine limited to the extent that the codes were each compiled under Microsoft Fortran 4.1 which in turn requires a level of DOS of 3.0 (or higher) and with a math co-processor installed. The codes will not run without a co-processor. The codes were all run on machines with at least 640 kbytes of memory, but this may not be a limitation. The IBM-XT failed to successfully execute for two of the problems in this mix of cases. These same problems executed successfully on the other PCs. It is not at all clear what caused these failures. These failures could even be due to limitations of the compiler used to generate the code. The Australians[3] highly recommended the Lahey F77L compiler, and judging by the amount of modification that was required to successfully compile their source code with the Microsoft compiler this compiler is at least much more forgiving.

The UM2DB code has at least two limitations. The first is common also to the main frame version of the code. With a depletion problem where burnup-dependent cross-section data are read from a binary file created by LINX, UM2DB is unable to properly read these data when four energy group data are requested. The code runs successfully with two energy group data, and when data is entered as card images in the input. It is not clear at this time whether the problem is with the LINX code or with the UM2DB code. Most of the use of the code has been with two group data. The second limitation is a limit in the amount of storage available with the dimensions fixed in the code. As noted earlier, the dimension of the container array was set not at its maximum value but such that the executable code would fit on a low density floppy disk. The container array dimension is now set at 20,000 words and the code occupies almost 347 Kbytes. Thus, with 640 Kbytes of RAM the size of the code could be increased by about 293 Kbytes (less the storage occupied by the resident part of the operating system). The current container array dimension can be increased significantly. Since the memory requirements are different for each problem, it is difficult to define the maximum problem size that can now be run, but perhaps the IAEA benchmark core will give some perspective on size. The benchmark problem with 2 energy groups, 19 X 16 mesh intervals, and 81 zones requires almost 11,000 words of the 20,000 available. This problem assumes symmetry, and only a quarter of the core is modeled. The container array dimension can be increased to match the available storage in RAM, but the complexity of the problem that can be run on a PC is still clearly limited.

The limitations of the LEOPARD code are more fundamental than related to the PC. As noted earlier the geometries that can be represented are limited to simple pin or plate unit cells (fuel, clad, and coolant) with an extra region to give a supercell model. The code is also limited to a choice of 2 or 4 energy groups total with only one thermal group, and scattering is limited to from one group to the next adjacent group. The number of materials in the working library is fixed at 35 materials, and Boron-10 is the only control rod material in the library. Although the library contains heavy water as one of the materials, this material should not be used as the coolant in the unit cell model, and its use even as a reflector material is questionable. New materials can be added to the library, but only with great difficulty. The code may be adequate for simple excess reactivity and fuel cycle scoping type computations, but even the four group data with only one resonance group and one thermal group is inadequate for computing reactivity feedback coefficients. Other available codes, such as the WIMS code, should be considered as alternatives.

The limitation of the PARET code on a PC is one more of patience and the stability of PC and its electrical supply. The loss of flow transient in Table II, for example, ran for almost three hours on an IBM-AT with over 4200 time steps. Some slow ramp reactivity insertion cases have run almost ten times this number of time steps (>30,000). The code could run for more than a day to complete some problems. The most recent version of the code now has a restart capability where a restart file is written after a selected number of time step alternately to two separate units (see ref. 9), and problems can easily be stopped and restarted to run in selected segments of time.

CONCLUSIONS

Some very useful analyses can be done with personal computers where main frame computers are unavailable or too expensive to use. The running times of the same problems on various PCs and some main frame computers were compared. The codes run slower on PCs, and some problems can run for hours on a PC. Where both PCs and main frame computers are available, the user may wish to weigh the loss of the use of the PC against the higher cost of running on the main frame. The relative running times of some of the codes on PCs suggest that a true IBM-XT may be too slow to be practical for some of the cases, but since many XTs are upgraded with accelerator boards or 80286 chips, this may not be of concern. Of greater concern is the failure of the XT to successfully execute two of the problems of interest. The other PCs had no difficulty executing these same problems. This difficulty may be unique to the XT used in this study. The cause of these run time errors was not determined. The reliability of the PC and its source of electricity must be high for the longer running problems.

The accuracy of the answers obtained on the PCs appears to be as good as that obtained with the main frame computers for the same problem. The quality of the answers from the PC with the UM2DB code may be limited by the detail that can be included within the available storage on the PC and by the quality of the cross-section data provided. The quality of the cross-sections provided by the LEOPARD code may not be acceptable for some applications. The LEOPARD code is very limited in the number of materials available in its library, in the geometries allowed, and in the number energy groups and the choices of energy group structure for the broad group library.

The LEOPARD/LINX/UM2DB package can provide some very useful results for some neutronics applications but has some limitations as noted above. The NATCON and PLTEMP codes are very useful for steady-state thermal-hydraulic analysis for both natural convection and forced flow cooling modes. The PARET code can provide both a steady-state thermal-hydraulic capability and a coupled point kinetics, hydrodynamics, and heat transfer capability for transient analysis.

ACKNOWLEDGMENTS

The contributions of Dr. Brian E. Clancy and his associated at the Australian Nuclear Science and Technology Organization in providing the initial PC versions of Fortran source code is gratefully acknowledged.

REFERENCES

1. David C. Losey, Forrest B. Brown, William R. Martin, and John C. Lee, "Core Physics Analysis in Support of the FNR HEU - LEU Demonstration Experiment," pp.450-498, ANL/RERTR/TM-3, CONF-801144 (Nov. 1980).

2. K. Mishima, K. Kanda, and T. Shibata, "Thermal-hydraulic Analysis for Core Conversion to the Use of Low-enriched Uranium Fuels in the KUR," Research Reactor Institute, Kyoto University, KURRI-TR-258 (Dec. 1984).

3. Brian E. Clancy, Australian Nuclear Science and Technology Organization, Private Communication.

4. R. F. Barry, "LEOPARD - A Spectrum Dependent Non-spatial Depletion Code for the IBM-7094," Westinghouse Electric Corp., WCAP-3269-26 (Sept. 1963).

5. W. W. Little,Jr. and R. W. Hardie, "2DB User's Manual," Battelle Memorial Institute - Pacific Northwest Lab., BNWL-831 (Aug. 1969).

6. R. S. Smith and W. L. Woodruff, "A Computer Code, NATCON, for the Analyses of Steady-state Thermal-hydraulics and Safety Margins in Plate-type Research Reactors Cooled by Natural Convection," Argonne National Lab., ANL/RERTR/TM-12 (Dec. 1988).

7. C. F. Obenchain, "PARET - A Program for the Analysis of Reactor Transients," Phillips Petroleum Co.,.IDO-17282 (Jan. 1969).

8. William L. Woodruff, "A Kinetics and Thermal-hydraulics Capability for the Analysis of Research Reactors," Nucl. Tech., 64, pp. 196-206, Feb. 1984.

9. W. L. Woodruff, "Additional Capabilities and Benchmarking with the SPERT Transients for Heavy Water Applications of the PARET Code," International Meeting on Reduced Enrichment for Research and Test Reactors, Sept. 10-14, 1989 Berlin, Germany, to be published.

10. J. E. Matos, K. E. Freese and W. L. Woodruff, "Comparison of Safety Parameters and Transient Behavior of a Generic 10MW Reactor with HEU and LEU Fuels," Tokai, Japan, pp.270-288, JAERI-M 84-073 (Oct. 1983).

11. R. S. Smith and W. L. Woodruff, "Thermal-hydraulic Aspects of Flow Inversion in a Research Reactor," pp.449-460, ANL/RERTR/TM-9, CONF-861185 (May 1988).