BNL-NUREG--42998

DE89 016802

LWR POWER PLANT SIMULATIONS USING THE AD10 AND AD100 SYSTEMS*

W. Wulff (BNL), H. S. Cheng (BNL), C. J. Chien (INER), J. Y. Jang (TPC), H. C. Lin (TPC), A. N. Mallen (BNL), and S. J. Wang (INER)

Brookhaven National Laboratory (BNL), Upton, NY 11973,
Institute for Nuclear Energy Research (INER), Lung-Tan, Taiwan,
Taiwan Power Company (TPC), Taipei, Taiwan

ABSTRACT

Boiling (BWR) and Pressurized (PWR) Water Reactor Power Plants are being simulated at BNL with the AD10 and AD100 Peripheral Processor Systems. The AD10 system has been used for BWR simulations since 1984 for safety analyses, emergency training and optimization studies. BWR simulation capabilities have been implemented recently on the AD100 system and PWR simulation capabilities are currently being developed under the auspices of international cooperation.

Modeling and simulation methods are presented with emphasis on the simulation of the Nuclear Steam Supply System. Results are presented for BWR simulation and performance characteristics are compared of the AD10 and AD100 systems. It will be shown that the AD100 simulates two times faster than two AD10 processors operating in parallel and that the computing capacity of one AD100 (with FMU processor) is twice as large as that of two AD10 processors.

*Work performed in part under the auspices of the U.S. Nuclear Regulatory Commission (BWR simulation on the AD10 system).

1. INTRODUCTION

Brookhaven National Laboratory (BNL) has successfully used Applied Dynamics International's simulation technology and equipment to predict thermohydraulic transients in nuclear power plants for over eight years.

Two AD10 computers, operating in parallel, were programmed in MPS10, starting in 1981, to simulate the nuclear steam supply system of a Boiling Water Reactor (BWR), and later the Balance of Plant, consisting of steam lines, turbine-generators, condensers, condensate and feedwater pumps and feedwater preheaters. The BWR simulation encompasses now the entire plant, including the three control systems for pressure, power and feedwater regulations and all the engineered safety systems for reactor scram, for safety and relief valve operations, emergency cooling and residual heat rejection. The MPS10 program developed for the BWR simulation is called the High-Speed Interactive Plant Analyzer or HIPA-BWR code and constitutes, along with the two AD10 computers and associated peripheral equipment, the BNL Engineering Plant Analyzer (EPA) for BWR plants. The EPA is now being used in remote access mode via commercial telephone lines at BNL for the Nuclear Regulatory Commission (NRC) and also by the NRC from Washington, D.C., by the Consejo Seguridad Nuclear in Madrid, Spain, by Public Service Electric and Gas and by the New York Power Authority.

An AD100 is being programmed, since June 1988, in ADSIM to simulate nuclear power plant thermohydraulics in Light Water Reactors (LWR), that is in both BWR and Pressurized Water Reactors (PWR). This simulation development work is a cooperative program, with Taiwan Power Company (TPC) in Taipei, the Institute of Nuclear Energy Research in Lung-Tan, Taiwan, ADI and BNL being the active participants. The modeling and software development is now completed for BWR plant simulations on the AD100. The same models as previously used in the AD10s were reprogrammed in ADSIM. Developmental assessment and validation work for the AD100 simulation of BWR is near completion. More work is required to provide the same interactive graphics capabilities as are available on the AD10-based EPA for on-line control of plant parameters simulating operator actions and component failures. A new modeling and program development has been started to simulate a U.S. four-loop and a TPC three-loop Westinghouse PWR.

It is the purpose of this paper to present modeling and computing methods which evolved from the nuclear reactor simulation at BNL and which are considered to be useful also in general for simulating on the AD100 peripheral processor systems that are governed by partial differential equations. Secondly, it is the objective of this paper to present simulation results from the AD10 and AD100 systems and to compare their performance in the field of reactor simulation.

2. SIMULATION METHODS FOR REACTOR HYDRAULICS

2.1 Problem Formulation

In order to describe the methods developed at BNL for reactor simulation, we establish first two distinctive characteristics of reactor simulation.

2.1.1 Two-Phase Flow Modeling

The computer simulation of nuclear power plants encompasses the modeling of conventional components, such as pumps, valves, heat exchangers, turbines, motors and control systems. But the central problem of simulating nuclear power plants is the modeling of nonhomogeneous, nonequilibrium two-phase flow, for the purpose of predicting the coolant dynamics in the nuclear steam supply system. Light-water moderated and cooled reactors, i.e., Boiling Water Reactors (BWR) and Pressurized Water Reactors (PWR), contain water at 70 and 155 atmospheres, respectively. The water may boil and flash (phase change due to depressurization) and form a mixture of vapor and liquid. The vapor and liquid phases have in general strongly differing densities ρ_v and ρ_ℓ, different temperatures T_v and T_ℓ and different velocities w_v and w_ℓ. In two-phase flow analyses, the flow is called nonhomogeneous when the velocities w_v and w_ℓ differ; the mixture is in nonequilibrium when at least one of the two phasic temperatures, T_v or T_ℓ, differs from the saturation temperature $T_s(p)$, corresponding to the pressure p. The ability of the two-phase mixture to cool the nuclear fuel depends strongly on the volume concentration of the vapor, which is called the vapor void fraction α, $0 \leq \alpha \leq 1$. The cooling capability decreases markedly as the void fraction approaches unity. At the same time, however, the ability of the two-phase mixture to slow down neutrons and to maintain the fission power diminishes. Thus, the reactor shuts itself off as the cooling capability of the two-phase coolant deteriorates. This shows that it is extremely important to compute accurately the vapor void fraction in the reactor core and the separation of vapor and liquid in the reactor cooling system. Two-phase flow modeling is therefore the most important aspect of nuclear reactor simulation. The central issues of two-phase flow modeling are the prediction of phase separation by gravity or inertia, the formation and motion of mixture levels (discontinuities!), then counter current flows with flooding limitation, and nonequilibrium mass transfer between the phases.

The governing equations of two-phase flow models are the conservation equations of mass, momentum and energy. These describe the time rates of change of the principal flow variables (pressure, temperature, velocity) and must be supplemented (for closure) by constitutive relations for material properties and for transfer processes, by initial and by boundary conditions. Two-phase flow models have between three and six conservation equations. The simplest, three-equation model is the Homogenous Equilibrium Model (HEM) and applies only for well-mixed phases, in thermal equilibrium at saturation conditions. HEM applications are limited by the maximum possible of three restrictions: both phases must have equal temperatures, equal pressures and equal velocities. As the number of conservation equation is increased, the number of restrictions decreases. In the limit, the most fundamental model is the six-equation Two-Fluid Model (TFM). It implies no restrictions but calls for the greatest computational effort. All the models with five or fewer con-

servation equations are obtained from the TFM by adding pair-wise the phasic mass, momentum or energy equations.

The six-equation TFM contains the total of 35 variables and requires therefore the total of 35 equations [Ishii, 1975], i.e., six conservation equations, three interfacial transfer laws, the axiom of continuity and 23 constitutive equations for closure. Constitutive equations are obtained from experiments, and the need for specifying 23 constitutive equations for nuclear reactor conditions cannot be met with currently available instrumentation. Measurements needed for developing constitutive equations are difficult or impossible because the phasic interfaces between bubbles, vapor slugs, and liquid and between droplets and vapor deform chaotically in most prevailing flow regimes. The exception is the separated flow (horizontal stratified flow, film flow, etc.) where measurements are possible and where the TFM is used with success. As the number of conservation equations is reduced from six to two, the number of needed constitutive relations for mass, momentum and energy transfer between each phase and the solid structures decreases from seven to two.

The two-phase model which is best supported by currently available experimental data is the four-equation Drift Flux Model [Zuber, 1967; Ishii, 1977]. It has a vapor mass balance and three conservation equations for mixture mass, momentum and energy. The model accounts fully for nonequilibrium nonhomogenous flow phenomena in two-phase mixtures. The Drift Flux Model (DFM) is used in the BNL Engineering Plant Analyzer, because it provides the best-possible balance between simulation fidelity and computational effort. The DFM is also used in the frequency domain code NUFREQ-NP [Peng, Podowski and Lahey, 1984].

2.1.1 Governing Partial Differential Equations

The conservation equations for all transient two-phase flow models are in principle either hyperbolic or parabolic partial differential equations [Wulff, 1987, Section 2.4]. A fundamental choice [Wulff, 1987, Section 5.1] must be made for the numerical integration of partial differential equations [Wulff, 1987, Section 5.1]: either one converts the partial differential into implicit finite difference equations by replacing both time and space derivatives by difference quotients, and then solves a large system of nonlinear equations by iteration, involving matrix inversions, or one converts partial into ordinary differential equations with respect to time and then integrates these numerically by explicit integration. The principal difference is the choice between explicit and implicit integration methods. Techniques for converting partial differential equations for two-phase flow into either finite difference or ordinary differential equations can be found in [Wulff, 1987, Chapters 7, 8 and 9]. Selection criteria for implicit and explicit integrations are found in Section 3.6 of the same reference.

The BNL Engineering Plant Analyzer integrates, for coolant dynamics simulation, in the AD10 and AD100 these four conservation equations of the Drift Flux Model [Wulff, Cheng, Lekach and Mallen, 1984, p. 3-39], i.e. for:

Vapor Mass: $$\partial(\alpha\rho_v)/\partial\tau + \nabla \cdot (\vec{j}_v \rho_v) = \Gamma_v \qquad (1)$$

Mixture Mass:
$$\nabla \cdot \vec{j}_m + \frac{\alpha}{\rho_v} \frac{D_v \rho_v}{D\tau} + \frac{(1-\alpha)}{\rho_\ell} \frac{D_\ell \rho_\ell}{D\tau} = \frac{\rho_\ell - \rho_v}{\rho_\ell \rho_v} \Gamma_v \qquad (2)$$

Mixture Energy:
$$\partial(\rho_m u_m)/\partial\tau + \partial(j_v \rho_v h_v + j_\ell \rho_\ell h_\ell)/\partial z = q_w'/A \qquad (3)$$

Mixture Momentum:
$$\partial G_m/\partial\tau + \partial[\alpha\rho_v w_v^2 + (1-\alpha)\rho_\ell w_\ell^2]/\partial z =$$
$$-\partial p/\partial z - g_z \rho_m - f_{\ell o} \phi_{\ell o}^2 G_m |G_m|/(2\rho_\ell d_n), \qquad (4)$$

where α, ρ, p, u and h designate, respectively, void fraction, density, pressure, internal energy and enthalpy, g_z and G are the gravitational acceleration component in the direction opposite to the flow, Γ_v, q_w', $f_{\ell o}$, and $\phi^2_{\ell o}$ stand for vapor generation rate, linear wall heating rate (including direct gamma heating), wall shear and two-phase multiplier for wall shear, respectively. The symbols A and d_n are for flow cross-sectional area and hydraulic diameter, while j designates volumetric flux and w the axial mass velocity component. Subscripts ℓ, m and v designate liquid, mixture and vapor, respectively. $D_k/D\tau$ for $k=\ell,v$, is the Lagrangian derivative $\partial/\partial\tau + v_k \cdot \nabla$; z and τ are the axial coordinate and time. Equation (1) is the vapor mass balance of the fundamental Two-Fluid Model (TFM). Equation (2) is the sum of the two phasic mass balances of the TFM, while Equations (3) and (4) are obtained from the TFM, by adding the pairs of phasic energy and momentum balances and then area-averaging the results over the flow cross-section.

Equations (1) and (3) are transformed into ordinary differential equations by volume averaging over a computational cell with index (n), volume V_n and open flow boundaries designated by subscripts (n-1) and (n). Indicating volume averages by $\langle f \rangle = 1/V \int_V f dV$, one gets for one-dimensional flow from Equations (1) and (3):

$$\left(d\langle\alpha\rho_v\rangle/d\tau\right)_n = \left[(j_v\ \rho_v)_{n-1} - (j_v\ \rho_v)_n\right] A/V_n + \langle\Gamma_v\rangle \qquad (5)$$

$$\left(d\langle\rho_m u_m\rangle/d\tau\right)_n = \left[(j_v\ \rho_v\ h_v)_{n-1} + (j_\ell \rho_\ell h_\ell)_{n-1} - (j_v \rho_v h_v)_n - (j_\ell \rho_\ell h_\ell)\right] A/V_n$$
$$+ \langle q_w'\rangle/A\ . \qquad (6)$$

The products $\langle\alpha\rho_v\rangle$ and $\langle\rho_m u_m\rangle$ are state variables. Equations (5) and (6) are ready to be integrated with the built-in algorithms of the AD10 or AD100, once the flux terms at the cell boundaries, i.e. the terms proportional to the volumetric fluxes j, are calculated from the state variables. The source terms $\langle\Gamma_v\rangle$ and $\langle q_w'\rangle/A$ are computed from constitutive relations for vapor generation and from heat transfer analyses, respectively.

The phasic volume fluxes j_v and j_ℓ for vapor and liquid are computed from the mixture volumetric flux j_m and from the kinematic identities:

$$j_v = \alpha(C_o j_m + v_{gj}), \qquad j_\ell = j_m - j_v, \tag{7}$$

where C_o and v_{gj} are the void distribution parameter and the void-weighted area-averaged vapor drift velocity, respectively [Zuber, 1967]. These are computed from flow-regime dependent drift flux correlations. The mixture volumetric flux j_m is computed for each cell boundary from Equation (2). Thus, there remains only the need to compute the densities ρ_v, ρ_ℓ and enthalpies h_v, h_ℓ at the cell boundaries from the system state variables $\langle\alpha\rho_v\rangle$ and $\langle\rho_m u_m\rangle$.

These state variables and the system pressure p yield first the volume-averaged thermodynamic properties $\langle\rho_v\rangle$, $\langle\rho_v h_v\rangle$ and $\langle\rho_\ell h_\ell\rangle$ through the use of thermal and caloric equations of state. The conventional method for obtaining cell boundary values $(\rho_v)_m$, $(\rho_v h_v)_n$ and $(\rho_\ell h_\ell)_n$ from their corresponding volume-averaged values is the donor cell differencing method, which constitutes upwind-weighted zeroth-order interpolation. For the general volume-averaged variable $\langle y\rangle_n$ of volume V_n, the method means that one selects for the product $(jy)_n$:

$$(y)_n = \langle y\rangle_n \quad \text{for } j \geq 0\ , \qquad (y)_n = \langle y\rangle_{n+1} \quad \text{for } j_n < 0. \tag{8}$$

The method implies perfect mixing within a cell volume, such that the values at the exit equal the volume-averaged value, and it reflects signal propagation or advection of the property y in the direction of the flow. It is well known [Leonard, 1979; Wulff, 1987, Sections 8.3 and 8.4] that the "numerical" mixing of the zeroth-order, donor cell differencing method has strong numerical damping and requires therefore for accuracy a large number of small computational cells. Equation (8) is used for the BWR simulation. Since the PWR system is much larger, a new method was required and developed at BNL to reduce the number of needed cells and thereby the needed computational effort, while attaining low numerical diffusion. The method is the upwind-weighted quadratic interpolation for volume-averaged state variables and fashioned after a similar method derived by Leonard for discrete local, rather than cell-averaged, state variables [Leonard, 1979].

2.2 Newly Developed Computing Methods

2.2.1 Upwind-Weighted Quadratic Interpolation

The donor cell differencing with zeroth-order interpolation according to Equation (8) has the truncation error T.E., i.e. the difference between either Equations (1) and (5) or Equations (3) and (6), and therefore the numerical diffusivity ν, as given by:

$$\text{T.E.} = \frac{h}{2}\, j\, \frac{\partial^2 y}{\partial z^2}\ , \qquad \nu = \frac{h}{2}\, j\ , \tag{9}$$

where $h = V_n/A$ is the length of the flow segment with volume V_n and cross-sectional area A. The numerical diffussivity ν is of the order of 1 m^2/s, or five orders of magnitude larger than the turbulent diffusivity of water. Equation (8) produces strong and nonphysical damping.

The upwind-weighted quadratic interpolation is designed to reduce the truncation error and thereby the numerical diffusivity. A power polynomial $y(z) = a + bz + cz^2$ is passed through three adjacent cells: two cells with indices n-1 and n upstream, and one cell with index n+1 downstream of the cell boundary with index n. The polynomial is made to meet three conditions, namely that its three sectional averages:

$$\frac{1}{n} \int_{(k-1)h}^{kh} y(z)dz = \langle y \rangle_{(n+k)} , \quad \text{for } k = -1, 0, 1 \tag{10}$$

be equal to the averages as obtained from integrating the differential equations, such as Equations (2) or (3), for the three adjacent cells with indices n-1, n and n+1. The three conditions are solved for the leading polynomial coefficient $a = y(o) = (y)_n$. The result is:

$$(y)_n = \frac{1}{6} \left[-\langle y \rangle_{n-1} + 5\langle y \rangle_n + 2\langle y \rangle_{n+1} \right] , \quad \text{for } j_n \geq 0 \tag{11}$$

$$= \frac{1}{6} \left[2\langle y \rangle_n + 5\langle y \rangle_{n+1} - \langle y \rangle_{n+2} \right] , \quad \text{for } j_n < 0 \tag{12}$$

Equations (11) and (12) apply only for interior cells in a flow duct with N cells, i.e. for n = 2,3, ...,N-2. However, the same method as used for Equations (11) and (12), along with appropriate entrance conditions, produce similar expressions for entrance and exit cells. For example, if the flow duct emerges from a mixing plenum with $\langle y \rangle_o$ as its average, then one finds for the entrance:

$$(y)_o = \langle y \rangle_o , \quad \text{for } j_o \geq 0 \tag{13}$$

$$= \frac{1}{6} \left[11 \langle y \rangle_1 - 7\langle y \rangle_2 + 2\langle y \rangle_3 \right] , \quad \text{for } j_o < 0 \text{ and} \tag{14}$$

$$(y)_1 = \frac{1}{4} \left[-2\langle y \rangle_1 + 5\langle y \rangle_2 + \langle y \rangle_3 \right] , \quad \text{for } j_1 \geq 0 \tag{15}$$

$$= \frac{1}{6} \left[2\langle y \rangle_1 + 5\langle y \rangle_2 - \langle y \rangle_3 \right] , \quad \text{for } j_1 < 0 \tag{16}$$

The truncation error T.E. and the numerical diffussivity ν of Equations (11) and (12) are:

$$\text{T.E.} = \left| \frac{h^3}{12} j \frac{\partial^4 y}{\partial z^4} \right| , \quad \nu = 0 , \tag{17}$$

i.e., there are no first-order diffusion terms (proportional to $\partial^2 y/\partial z^2$), nor first-order dispersion terms (proportional to $\partial^3 y/\partial z^3$) and the second-order damping is very small, although negative. It is expected, however, that the

time integration provides still sufficient positive damping to insure numerical stability. It will have to be shown whether there is a need to introduce turbulent damping to achieve numerical stability.

2.2.2 Solution to Systems of Linear Equations on the AD100

The solution of Equation (2) for a network of connected flow loops requires the calculation of one time-dependent integration constant $j_m(o,\tau)$ for each closed loop. The integration constants are defined by the integral to Equation (4) which is written first as a loop momentum balance, one for each closed loop. Details of the loop momentum balance derivation and of the relation between the loop momenta and the integration constants $j_m(o,\tau)$ can be found in [Wulff, Cheng, Lekach and Mallen, 1984, pp. 3-51 to 3-53]. Since Equation (7) is linear in j_m, the momentum flux $G_m = \rho_v j_v + \rho_\ell (j_m - j_v)$ and therefore the system of N equations for the N integration constants $j_{m,i}(o,\tau)$ is also linear. For the simulation of a BWR with three reactor core channels N=4. Here, the system of equations could be solved explicitly by algebra and the result is simple to evaluate. For the simulation of a four-loop Westinghouse PWR with two core channels, however, N=6 and the algebraic solution would have required a prohibitively large number of arithmetic operations for its evaluation during a simulation, because of its complexity. We have therefore decided first to reduce the number N of equations from six to five, since that is simple and advantageous, and then to solve the remaining five equations by matrix manipulation.

We have evaluated the Gauss Elimination and the Gauss-Jordan Methods (Carnahan, Luther and Wilkes, 1969] for the purpose of minimizing the execution time required on the AD100 for solving five coupled linear equations with a non-sparse coefficient matrix. The execution time is determined by the number of required additions, multiplications and divisions (where the count of additions includes all the subtractions), and by the time these operations require. The result is given in Table 1 below for N, an arbitrary number of equations. For the given execution times of the AD100, as shown in the second column of Table 1, it is clear that up to N=5 the Gauss-Jordan method is faster because it requires fewer divisions than the Gauss Elimination method. For N=6, Gauss Elimination requires 28.3 μs, 0.1 μs less than the Gauss-Jordan method. For N>6 the number of additions and multiplications needed for the Gauss-Jordan method outweighs the greater number of divisions needed for the back substitutions of the Gauss-Elimination method.

The Gauss-Jordan method can be programmed in ADSIM for the AD100 more compactly than the Gauss-Elimination method because the former is reducible to N cycles, each one starting with a division and followed, for the first cycle, by N elementary column reductions, and then for each subsequent cycle by one column reduction less than for the previous cycle. All elementary column reductions are identical and applied to the coefficient matrix, augmented by the column vector which contains the known right-hand side B of the system AX=B. The last cycle produces the unknown X, whose components are in our case the five integration constants $j_m(o,\tau)$, times the cross-sectional area A, i.e., BTMxxyy = $Aj_m = \beta_m(o,\tau)$ (cf.Eq.(19)).

It must be recognized that the AD100 solves the system of linear equations without the overhead burden from computing addresses for subscripted

variables which arises in FORTRAN-programmed, general-purpose computers. The total execution time is computed for the AD100 entirely from the execution time contributions listed in Table 1. The ADSIM FUNCTION for column reductions and its application to the solution of five coupled linear equations is given in Figure 1.

Table 1 Number of Operations and Execution Time Required to Solve N Linear Equations on the AD100

Required Operation	AD100 Execution Time	Number of Operations for	
		Gauss-Jordan Method	Gauss Elimination Method
Addition	0.1 μs	$N(N^2-1)/2$	$N(N-1)(2N+5)/6$
Multiplication	0.1 μs	$N^2(N+1)/2-1$	$N(N-1)(N+4)/3-1$
Division	0.9 μs	N	2N-1
Total Execution Time for N = 5		17.9 μs	19.0 μs
10		113.4 μs	96.5 μs
100		100.6 ms	68.3 ms
500		12.51 s	8.37 s

```
!
FUNCTION COLRDCT(R1,R2,R3,R4,R5 = C1,C2,I1,I2,I3,I4,J1,J2,J3,J4)
!   Performes Elementary Matrix Row Reductions
!
        R1 = C1*C2
        R2 = I1-R1*J1
        R3 = I2-R1*J2
        R4 = I3-R1*J3
        R5 = I4-R1*J4
END FUNCTION
!
!
!
                    A = 1/S11
R12,R22,R32,R42,R52=COLRDCT(A,SDL,S22,SDL,SDL,DCL,SDL,SDL,SDL,S51)
R13,R23,R33,R43,R53=COLRDCT(A,SDL,SDL,S33,SDL,DCL,SDL,SDL,SDL,S51)
R14,R24,R34,R44,R54=COLRDCT(A,SDL,SDL,SDL,S44,DCL,SDL,SDL,SDL,S51)
R15,R25,R35,R45,R55=COLRDCT(A,SCR,SCR,SCR,SCR,S55,SDL,SDL,SDL,S51)
X01,X02,X03,X04,X05=COLRDCT(A, X1, X2, X3, X4, X5,SDL,SDL,SDL,S51)
                    B = 1/R22
T23,T13,T33,T43,T53=COLRDCT(B,R23,R13,R33,R43,R53,R12,R32,R42,R52)
T24,T14,T34,T44,T54=COLRDCT(B,R24,R14,R34,R44,R54,R12,R32,R42,R52)
T25,T15,T35,T45,T55=COLRDCT(B,R25,R15,R35,R45,R55,R12,R32,R42,R52)
X12,X11,X13,X14,X15=COLRDCT(B,X02,X01,X03,X04,X05,R12,R32,R42,R52)
                    C = 1/T33
U34,U14,U24,U44,U54=COLRDCT(C,T34,T14,T24,T44,T54,T13,T23,T43,T53)
U35,U15,U25,U45,U55=COLRDCT(C,T35,T15,T25,T45,T55,T13,T23,T43,T53)
X23,X21,X22,X24,X25=COLRDCT(C,X13,X11,X12,X14,X15,T13,T23,T43,T53)
                    D = 1/U44
V45,V15,V25,V35,V55=COLRDCT(D,U45,U15,U25,U35,U55,U14,U24,U34,U54)
X34,X31,X32,X33,X35=COLRDCT(D,X24,X21,X22,X23,X25,U14,U24,U34,U54)
                    E = 1/V55
BTMCRI1, BTMLPI1, BTMLPI2, BTMLPI3, BTMLPI4
                      =COLRDCT(E,X35,X31,X32,X33,X34,V15,V25,V35,V45)
!
```

Figure 1 ADSIM Coding for the Solution of Five Coupled Linear Equations

2.3 Newly Developed Modeling Principles

The development at BNL of nuclear reactor simulation capabilities has lead to the formulation of five important modeling principles which have not been used in any other computer code, simulating nuclear reactor systems, such as TRAC [TRAC-PF1/MOD1, 1986], RELAP5/MOD2 [Dansom, et al., 1986] or RETRAN [McFadden et al., 1981]. These principles are to a large extent responsible for the computing speed and economy achieved with the ADI technology in the BNL Engineering Plant Analyzer. Even though the last two of the five principles described below were stipulated specifically by the architecture of the AD10 and AD100 systems, all five principles are generally applicable to simulation on any digital computer.

In order to achieve maximum simulation effectiveness, one must optimize as a whole all three parts of simulation, namely: (i) the model formulation, (ii) the computing methods, and (iii) the computer architecture with the associated programming language. Adaptations of existing models or computing methods to new computers have consistently disappointed in the past. Moreover, computer programs have been proven to be either transportable to several computers, or they are efficient, but they are never both, because a portable code cannot utilize the specific strengths of any two different computers. Even execution times are machine and compiler dependent and influence the optimization of computing effectiveness, as shown in Section 2.2.2 above. Therefore, it is important to match simultaneously simulation objectives, model formulation, computing methods and computer architecture. This is the overriding requirement for effective simulation. The aim is to achieve the greatest modeling fidelity possible with the smallest number of arithmetic and logical operations.

Below are summarized the five modelling principles [Wulff, Cheng and Mallen, 1987], which evolved at BNL during the Engineering Plant Analyzer development. The five principles were applied to the entire simulation of reactor systems, but they are related here to the simulation of the coolant dynamics, as described in Section 2.1.2 above.

2.3.1 The Principle of Model Selection demands that one select the least complicated model which accommodates all available experimental information and all processes of interest.

This principle reflects the axiomatic fact, that no computer model can produce more information than is contained in, or can be extracted from, its data base of constitutive relations, no matter how many conservation equations the model has, or how complex it is.

For two-phase flow, one concludes from the Principle of Model Selection and from the recognition of currently available experimental data on two-phase flow, that:

(i) the Drift Flux Model (DFM) is the best choice when both phases are mixed and tightly coupled (almost all flow regimes in the reactor vessel),

(ii) the Two-Fluid Model (TFM) may be the best choice if the phases are separated and weakly coupled (stratified flow in long horizontal pipes, or film flow). Here the interfacial geometry is simple, measurements are possible and boundary layer analyses can be developed.

Ishii has extended the DFM even to separated flow [Ishii, 1977]. We have used his results and employee the DFM exclusively. It provides the most cost-effective modeling and the same modeling fidelity as is possible with the TFM and the currently available experimental information. The choices of the six-equation TFM in TRAC and RELAP-5 and the five-equation model in RETRAN are not optimal because the associated computing efforts are not matched by additional computing detail in the results from these codes, due to the lack of appropriate and sufficient information for closure.

2.3.2 The Principle of Model Simplification requires that, through an order-of-magnitude analysis, all irrelevant terms be eliminated from the model equations, while all important processes and phenomena are retained.

The indiscriminant retention of irrelevant terms in the governing equations increases simulation cost and time while it decreases significantly one's ability to manage and maintain the computer code.

As one example, acoustical effects everywhere except in the BWR steam line simulation, are omitted in the Engineering Plant Analyzer, but retained in all other systems codes. Acoustical effects are eliminated by setting the pressure gradient ∇p equal to zero in the density derivatives of Equation (2). As a result, this partial differential equation provides immediately the state equation for the system pressure, i.e.:

$$\frac{d\langle p\rangle}{dt} = \frac{\sum_{in}(Aj_m) - \sum_{out}(Aj_m) + \iiint_V \frac{\rho_\ell - \rho_v}{\rho_\ell \rho_v}\Gamma_v\, dV - \iint_A \frac{1}{\rho_\ell^2}\left(\frac{\partial \rho_\ell}{\partial u}\right)_p q_w''\, dA}{\frac{1}{\rho_g}\frac{d\rho_g}{dp}\iiint_V \alpha dV + \iiint_V \frac{1}{\rho_\ell}\left(\frac{\partial \rho_\ell}{\partial p}\right)_u (1-\alpha)dV} \qquad (18)$$

Here, q_w'' is the wall heat flux, all other terms are as defined below Equation (4). The first two terms of the numerator represent the excess of inflows over outflows, the other two terms the volume dilatations due to phase change and heating, respectively, while the numerator represents the system's compliance due to vapor and liquid compressibilities.

Equation (18) is an ordinary differential equation, since the right-hand side is obtained by summing and by spatial quadratures of known quantities. An equation like Equation (18) provides the pressure for all thermodynamic properties of the fluid in the primary or secondary system. Acoustic waves are unimportant and their motion is not modeled; acoustic disturbances are modeled instead to propagate instantly. Notice that all simplifications are clearly interpreted and quantitatively assessed.

2.3.3 The Principle of Analytical Integration requires that analytical solutions be evaluated dynamically by the computer, wherever possible, instead of numerical integrations.

This principle helps to strive toward effective simulation by eliminating the uncertainties from numerical diffusion, by avoiding the effects from numerical instability and the issues of convergence associated with numerical integration, and by providing the basis for quantitative estimates of uncertainties arising from any assumptions which might be implied in the analytical integration.

As an example, we offer the implementation of Equation (2) to find the distribution $j_m\ (z,\tau)$ of the mixture volumetric flux needed in Equation (7) for computing the local phasic volumetric fluxes j_v and j_ℓ in Equations (5) and (6). The analytical integration of Equation (2), carried out after substituting for the density derivatives pressure and energy derivatives (through the use of caloric and thermal equations of state) and after eliminating energy derivatives through the use of phasic energy equations, reduces this partial differential equation into simple spatial quadratures [Wulff, Cheng, Lekach and Mallen, 1984]. After setting $\beta_m = Aj_m$, one gets:

$$\beta_m(z,\tau) - \beta_m(o,\tau) = \int_o^z A \frac{\rho_\ell - \rho_g}{\rho_\ell \rho_g} \Gamma_v \, d\eta - \frac{d\langle p\rangle}{d\tau} \int_o^z A \left\{ \frac{\alpha}{\rho_g} \frac{d\rho_g}{dp} + \frac{1-\alpha}{\rho_\ell} \left(\frac{\partial \rho_\ell}{\partial p}\right)_u \right\} d\eta$$

$$- \int_o^z \frac{1}{\rho_\ell^2} \left(\frac{\partial \rho_\ell}{\partial u}\right)_p q_w'(\eta)\, d\eta, \qquad (19)$$

where the integration constant $\beta_m\ (o,\tau)$ is computed from Equation (4), and is the result of the matrix manipulations described above in Section 2.2.2. Since the pressure derivative is known from Equation (18), it is relatively easy to carry out the spatial quadratures and compute the right-hand side of Equation (19). This is far easier and more effective than the numerical integration of the partial differential equation given above as Equation (2).

Similarly, we have integrated analytically around every closed loop Equation (4), the momentum balance, to obtain one ordinary differential equation each for every closed loop. The closed-loop integration eliminates the pressure gradient term from Equation (4), and the loop momenta are driven by pumps and gravity effects, they are retarded by friction and form losses. The analytical integration of the momentum balance around closed loops not only simplifies the computational effort from that of integrating a momentum balance for each and every computational cell, to that of integrating one only for every closed loop, but it also avoids unphysical "ringing" and reduces drastically the system stiffness.

2.3.4 The Principle of "No Iterations" requires that all the iterations required for solving nonlinear equations systems be eliminated from the actual simulation process by pretabulating the results in terms of state variables which are known from integrating state equations, and then by interpolating the tabulated results during the simulation.

This principle reduces the number of arithmetic and logical operations during the simulation by orders of magnitudes, the more so the more complicated the nonlinear equations are. It should be obvious that the special AD10

and AD100 features for interpolating multidimensional tables invites the formulation of this principle. The number of applications for this principle is particularly large for nuclear reactor system simulation.

One example are the calculations of $\{\rho_v, (\rho_v h_v), T_v, \alpha, \rho_\ell, (\rho_\ell h_\ell), T_\ell\}$ from the state variables $\langle\alpha\rho_v\rangle$, $\langle\rho_m u_m\rangle$ and $\langle p\rangle$ (cf. Equations (5), (6), and (18)), and from the complex nonlinear thermal and caloric equations of vapor and liquid water. The symbol T designates here the temperature.

Another example is the reduction of complicated heat transfer correlations with complex heat transfer regime logic, to a simple sequence of linear interpolations, free of decisions, which produce directly wall temperatures and convective heat fluxes, as needed for displays and for the energy balance, respectively. AD10 and AD100 simulations of heat transfer require therefore a much smaller fraction of frame time than all other reactor systems simulations on general-purpose computers.

2.3.5 The Principle of Pretablulated Functions is used also to reduce the required number of arithmetic and logical operations during the simulation, by pretabulating complicated expressions and by interpolating the resulting multidimensional tables during the simulation.

Adherence to this principle alone has been shown to reduce the computing time by more than a factor of three.

Each part of the right-hand sides of all state equations is examined to find out whether elementary library functions or table interpolation produce shorter execution times. This minimization of execution time would be hopelessly difficult to achieve in general-purpose computers because execution time data are not made available to the programmer and because no systems software is available for efficient table interpolation.

Typical candidates for pretabulation are the entire last term of Equation (4), the vapor generation term $\langle\Gamma_v\rangle$, in Equation (5), the linear wall heating term $\langle q_w'\rangle$ in Equation (6), the products (αC_o) and (αv_{gj}) in Equation (7), and the property terms in the integrands of Equations (18) and (19).

2.3.6 Closing Comments

It is the purpose of this chapter to convey the simulation methods used in the Engineering Plant Analyzer at BNL and to contrast these methods with those used elsewhere, to the extent that such methods apply in general to the simulation of thermohydraulic systems and could therefore be of general interest. It must be pointed out, however, that the computer architectures of the AD10 and AD100 systems contributed the greatest part to the achievement of the high computing speed in the Engineering Plant Analyzer. The speed could not have been achieved with a general-purpose computer, programmed in FORTRAN to utilize the methods described in this chapter.

3. SIMULATION RESULTS FROM THE AD10 AND AD100 SYSTEMS

3.1 BWR System Simulation with the AD10

3.1.1 AD10-Based EPA Description

The Engineering Plant Analyzer (EPA) currently in active use for safety analyses at BNL employs two coupled AD10 processors, which are hosted by a PDP-11/34 computer and operated remotely via commercial telephone from an IBM-compatible Personal Computer (PC) (XT or AT). Analog to digital and digital to analog converters were used in the past, as interface to a control panel and to a storage oscilloscope. These components are now emulated by the PC.

The EPA uses 12,300 and 587,000 words of memory for instruction and tabulated data, respectively, 2,307 of 2,339 available words of ARP memory and it uses 789 out of 800 available data areas. Both consoles operate in parallel with the frame time of 8.6 milliseconds (14.2 milliseconds in sequential operation). The number of state variables is 296. There are more than 7,000 outputs. The number of function tabulations is 359.

The EPA responds on-line to 52 keyboard commands, without interruption of the simulation, unless a HALT is desired by the user. The keyboard commands serve to introduce changes to boundary conditions, namely postulated reactor system component failures and reactor operator actions. While two preselected parameters are displayed on the PC screen, 190 parameters are stored on hard disc for later processing. Under remote access operation, the simulation speed is approximately three times faster than real time.

The EPA is the only simulation facility available that simuates all closed-loop feedback processes in the BWR power plant, including the effects from control systems, engineered safety systems and containment.

3.1.2 Recent EPA Simulation of BWR Transients

The EPA for BWR simulation has been used since 1984 for safety analyses at BNL, at the Consejo Seguridad Nuclear in Madrid, Spain, at the USNRC and by a number of utilities. It has also been used for emergency training at the USNRC Emergency Operations Center.

More recently, the EPA has been used to analyze the BWR instability which occurred on March 9, 1988 at the LaSalle-2 power plant. The EPA simulations showed that the instability was excited by the combination of three destabilizing phenomena, namely: (i) loss of core flow due to an inadvertent recirculation pump trip, (ii) reactivity insertion due to a valving error, which eliminated some feedwater preheating, and (iii) fuel burnup, which caused a concentation of fission power near the core entrance.

The EPA facilitated the LaSalle instability simulation with preliminary data on plant conditions within a week (long before official NRC funding was authorized). Definitive simulations of the incident and over 30 postulated variations were simulated within less than four weeks after the receipt of reliable plant data. In December 1988, the EPA was still the only facility

which was able to simulate the LaSalle instability correctly, without resorting to artifical destabilization effects. The EPA was the first facility to show how much the mean fission power of the reactor increases, as the amplitude of the power oscillations grows after a postulated scram failure. The EPA is the only facility that simulates correctly all systems effects on the flow and power oscillations in the reactor.

General Electric (GE) has used a TRAC-BWR version to simulate the LaSalle instability, but could not obtain oscillations until a destabilizing rod withdrawal was simulated. GE claimed before the NRC in March 1989 that the mean fission power does not increase as the power amplitude increases. GE conceded later in May 1989 that the mean does increase. The final GE results on the LaSalle instability have not been documented as of May 27, 1989. However, it is known that GE's TRAC-BWR requires 120 hours of CRAY XMP execution time to simulate 10 minutes of the LaSalle transient; BNL's EPA requires three minutes. This shows the simulation capability of the EPA in contrast to that of the FORTRAN TRAC code, executed on a general-purpose supercomputer.

Figures 2 and 3 show the fission power as simulated for the LaSalle conditions. Figure 2 shows how the plant shut itself off at 118% of full power, Figure 3 shows the continuing growth of the power amplitude in the postulated event that the shut-off system (scram) fails. Figure 4 shows that the EPA simulation produces bifurcation at large power oscillations, which are expected on the basis of simple theoretical considerations [March-Leuba, Cacuci and Perez, 1986].

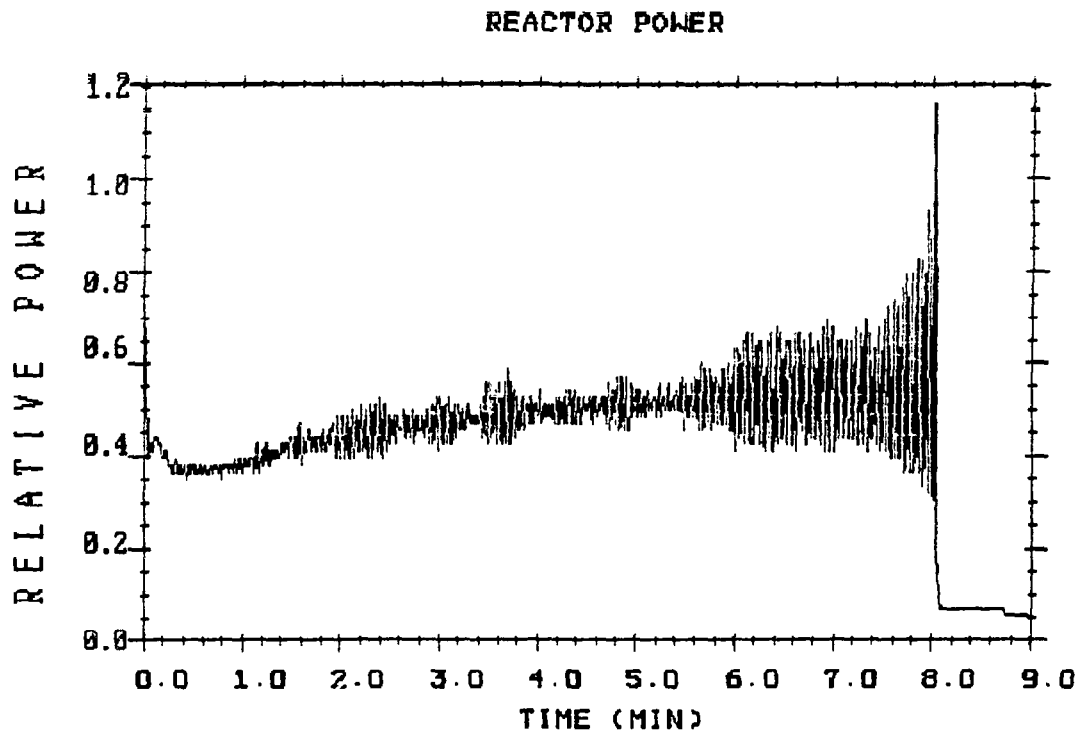

Figure 2 Power Oscillations Simulated with LaSalle Conditions and Automatic Shut-off at 118% of Full Power

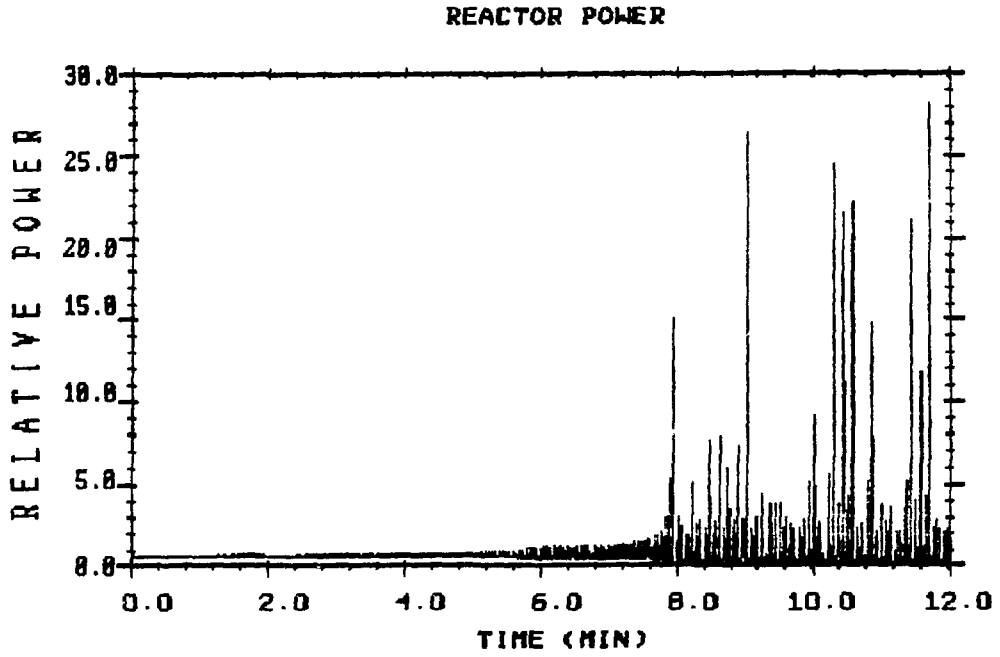

Figure 3 LaSalle Power Oscillations with Postulated Failure of Reactor Shut-Off System (Scram System)

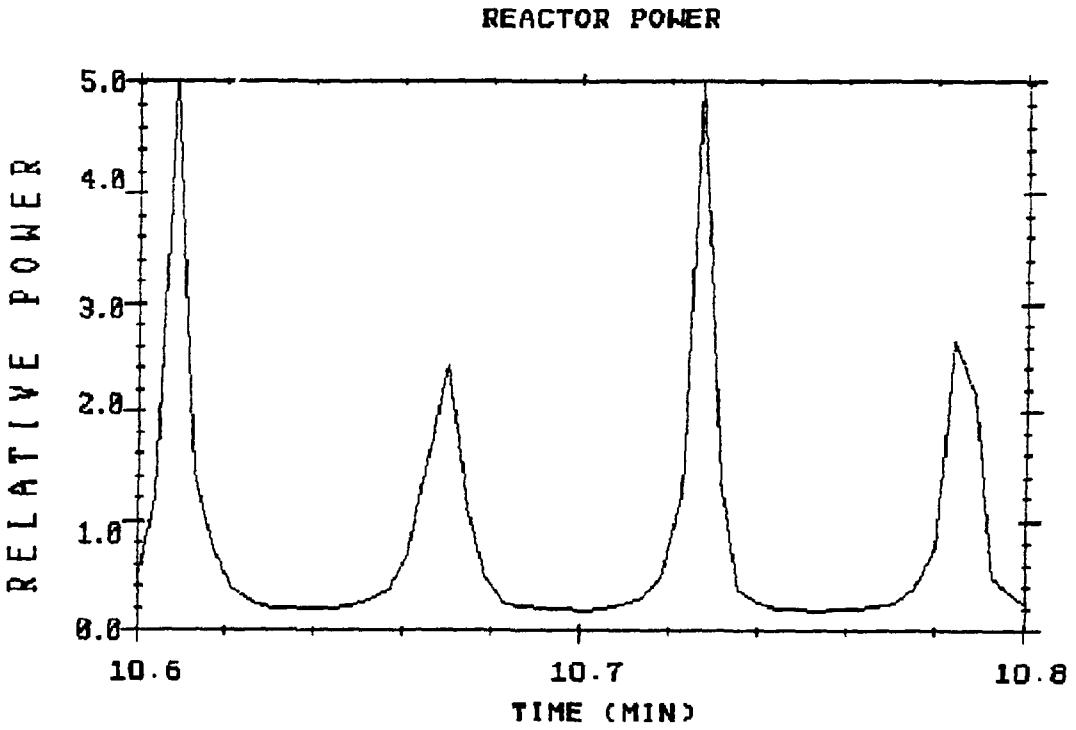

Figure 4 Simulated Oscillations with Expanded Time Scale Show Bifurcation: Two Amplitudes with Doubled Period

3.2 BWR System Simulation with the AD100

Three visiting scientists from Taiwan Power Company and from the Institute of Nuclear Energy Research have implemented the mathematical models as used in the AD10-based EPA, now in ADSIM for the AD100. The work is nearly completed. Developmental assessment is currently being performed by comparing the results from the AD100 simulation with plant data.

One AD100 console carries out now the simulation work of two AD10 consoles (cf. Section 3.1.1). Nearly 11,000 memory locations are used in the 65k-word STO processor. Instructions require 28,200 memory locations in the 65k word COM processor. There are 117 nonlinear functions tabulated. They occupy approximately 392,300 words of memory in the FMU processor of the AD100.

Three dynamic blocks are used in the ADSIM program, each having its distinct integration step. The frame time is approximately 5.7 milliseconds; the maximum speed-up factor is nine.

It is evident that the simulation capacity and speed of one AD100 console are about twice as large as those of two AD10 consoles, when the currently used BWR models are executed in both systems. The ratio of the computing capacity is determined by the capacity of COM memory.

Figure 5 shows the agreement obtained by comparing results from the AD10 with the results from the AD100. Shown is the pressure comparison for an Anticipated Transient without Scram (ATWS), caused by the Main Steam Isolation Valve (MSIV) closure. Similar agreement is obtained for such parameters as void fractions, core flow rates and fuel temperatures, etc. Preliminary calculations for the Chinshan BWR/4 power plant show the correct trend of key plant parameters. The causes of currently existing differences between plant data and simulation results have been identified as differences in valve set points which had been changed since the plant data were taken.

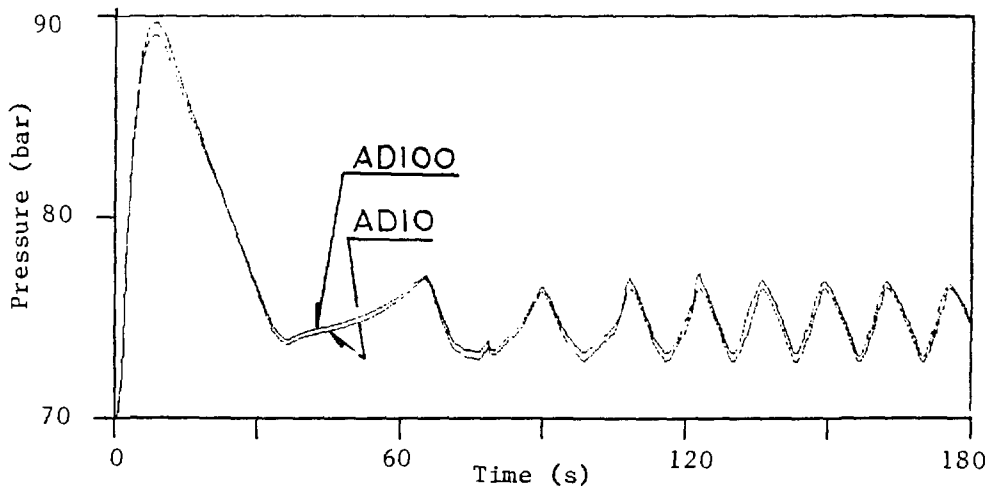

Figure 5 Comparison of AD100 and AD10 Results: Pressure for MSIV-Closure Induced ATWS

3.3 PWR Simulation with the AD100

Two BNL staff and one visiting engineer from the Taiwan Power Company in Taipei are currently cooperating to develop a PWR simulation capability. Both a four-loop (U.S., Indian Point 2) and a three-loop (Taiwan, Maanshan) Westinghouse PWR are being simulated.

The PWR simulation requires at least three times as many state variables as the BWR simulation. However, the same simulation methods as used for BWR plants are also being used for the PWR plants, with these four exceptions:

(i) both the vapor and the liquid phases are allowed to be in thermal disequilibrium for PWR simulation,

(ii) component-specific processes will be modeled for PWR components, primarily for pressurizer and U-tube steam generators, once-through steam generators, loop seals, etc.

(iii) individual control systems will be modeled for each steam generator,

(iv) PWR containment designs need to be modeled.

Currently, Equations (2) and (18) have been implemented to compute the two-phase flow distribution in coolant loops and reactor core channels, and to compute the system pressure. This requires 2,000 lines of ADSIM coding and 480 microseconds for execution. Progress in PWR simulation is slower than anticipated because the NRC is withholding its anticipated funding support.

4. CONCLUSION

Two-phase flow modeling is the most demanding part of simulating transients in nuclear power reactors. The AD10 and AD100 have been shown to outperform general-purpose computers, programmed in FORTRAN, in the simulation of nuclear reactor transients, with full accounting for nonequilibrium nonhomogeneous two-phase flow.

The AD100 was shown to have twice the computing capacity and twice the computing speed of two AD10 processors, working in parallel, and simulating a BWR plant, complete with Nuclear Steam Supply System, Balance of Plant, Control and Engineered Safety Systems, and the containment. ADSIM programming for the AD100 is easier and faster than MPS10 programming for the AD10.

Effective simulation of nuclear reactor transients can be achieved only by optimizing as a whole model selection, model formulation, computing techniques as well as computer architecture. The optimization must be aimed at clearly defined simulation objectives. Effective simulation cannot be achieved by adaptation alone of an unoptimized computer code to a newly developed array processor.

It has been recognized from the simulation of nuclear reactors at BNL, that the special-purpose computers AD10 and AD100 are superior for processing reactor model equations with their largely scalar structure, primarily because these computers are designed to utilize their processors' processing speed far better than general-purpose computers, without requiring that the modeling equations be programmed as independent streams of instructions for parallel processing. Computers designed to integrate state equations sequentially at high speed and low cost serve simulation purposes better than arrays of general-purpose microcomputers. This conclusion will not be altered in the future by the arrival of new microprocessors of lower cost and greater speed because the special-purpose and the general-purpose computers will benefit equally well from microprocessor improvements and because the superiority of the special-purpose computers for simulation derives from their architecture and programming language.

REFERENCES

Carnahan, B., Luther, A. H. and Wilkes, J. O. (1969), Applied Numerical Methods, John Wiley & Sons, Inc., NY.

Ishii, M. (1975), "Thermo-Fluid Dynamic Theory of Two-Phase Flow," Eyrolles, Pan's, France, p. 182.

Ishii, M. (1977), "One-Dimensional Drift Flux Model and Constitutive Equations for Relative Motion Between Phases in Various Two-Phase Flow Regimes," Argonne National Laboratory, Argonne, IL, ANL-77-47.

March-Leuba, J., Cacuci, D. C. and Perez, R. B. (1986), "Nonlinear Dynamics and Stability of Boiling Water Reactors: Part I-Qualitative Analysis," J. Nuclear Science and Engineering 93, pp. 111-123.

Peng, S. J., Podowski, M. Z. and Lahey, R. T., Jr. (1984), "NUFREQ-NP: A Digital Computer Code for the Linear Stability Analysis of Boiling Water Reactors," U.S. Nuclear Regulatory Commission, NUREG/CR-4116.

Wulff, W., Cheng, H. S., Lekach, S. V. and Mallen, A. N. (1984), "The BWR Plant Analyzer," Final Report, Brookhaven National Laboratory, Upton, NY, 11973, NUREG/CR-3943, BNL-NUREG-51812.

Wulff, W. (1987), "Computational Methods for Multiphase Flow," Second International Workshop on Two-Phase Flow Fundamentals, Rensselaer Polytechnic Institute, Troy, NY.

Wulff, W., Cheng, H. S. and Mallen, A. N. (1987), "Modeling and Numerical Techniques for High-Speed Digital Simulation of Nuclear Power Plants," Proc. ANS Topical Meeting on Anticipated and Abnormal Transients in Nuclear Power Plants, April 12-15, 1987, Atlanta, GA, Vol. 1, pp. II-3 to II-10.

Zuber, N. (1967), "Flow Excursions and Oscillations in Boiling Two-Phase Flow Systems with Heat Additions," Proc. Symp. Two-Phase Flow Dynamics, Vol. 1, p. 1071.