Journal of the Korean Nuclear Society
Volume 27, Number 6, December 1995

Waveform Relaxation Method for Reactor Transient Analysis

Keon Woo Park and Nam Zin Cho
Korea Advanced Institute of Science and Technology
(Received May 24, 1995)

원자로 천이해석을 위한 파형완화법

박건우 · 조남진
한국과학기술원
(1995. 5. 24 접수)

Abstract

We investigate the concurrent solution of differential equations by the waveform relaxation (WR) method, an iterative method for analyzing linear and nonlinear dynamical systems in the time domain. The method, at each iteration, decomposes the dynamical system into several subsystems, each of which is analyzed for the entire given time interval. The method, when efficiently implemented, results in algorithms with a highly parallelizable concurrent fraction. In this paper, the waveform relaxation method is introduced and applied to two types of reactor dynamics problems. It is concluded that the WR method can be applied to reactor dynamics equations, but that its parallel performance on the KMRR dynamics is only modest.

요 약

반복계산법(iterative method)중의 하나인 파형완화법(Waveform Relaxation Method)을 이용하여 시간의 함수인 비선형 원자로 동역학(reactor dynamics)의 해를 병렬처리기법으로 구하였다. 파형완화법은 각 반복계산과정중에서, 먼저 전체 시스템을 작은 부시스템(subsystem)으로 나누고, 각각의 부시스템들은 주어진 각각의 시간간격(time interval)에서 해를 독립적으로 구한다. 만약 이 기법에 맞게 효과적으로 부시스템으로 나눌 수 있다면 병렬처리기법에 잘 응용될 수 있다. 본 논문에서는 파형완화법이 소개되었고, 두 종류의 원자로 동역학에 응용되었다. 결론적으로 파형완화법은 원자로 동역학에 응용될 수 있으나, 다목적 연구로에 응용한 결과는 병렬효과가 그다지 크지 않은 것으로 나타났다.

1. Introduction

In the numerical solution of large dynamical systems it has turned out to be advantageous to use iterations in time domain with the aid of parallel com puters. In such a method the system is decomposed into smaller subsystems, each of which is integrated numerically with its own processor reading inputs from other subsystems from the earlier iteration. If these subsystems are loosely coupled and a good starting value for the iteration is available, then the computing time may decrease significantly. The back-

ground of the method appears in electrical network simulation, and it is known there as waveform relaxation (WR) or dynamic iteration[1-3].

The objectives of this study are to investigate the waveform relaxation method by applying the method to the reactor dynamics in order to improve the computational efficiency. Two test problems are considered : one is the point kinetics equations with six groups of delayed neutron precursors equations, and the other is the Korea Multipurpose Research Reactor dynamics.

In this paper, a parallel computer network composed of four T-800 transputers is used to solve the reactor dynamics. T-800 transputer[4] is a message-passing type MIMD (multiple instruction multiple data) architecture. The transputer is a microprocessor with four serial links to be used in communicating with other transputers.

Speedup factor (S) is used to describe the performance of parallel processing. In a multi-processor system, the parallel speedup (S) achieved when using P processors to solve a problem of size N is defined by $S(N) = T1(N)/TP(N)$, where TP(N) is the time required to solve the given problem size N using P processors. This quantity represents how much faster the problem is solved by using P processors.

2. Waveform Relaxation Method

The basic idea behind the dynamic iterative method or waveform relaxation method is analogous to the standard Jacobi and Gauss-Seidel iterations used to solve linear systems of equations[5]. First we decompose the system into several lower-order subsystems. As an inevitable consequence, these subsystems have coupled variables one after another. The results of the WR method rely on a splitting or decoupling of a given system of differential equations into a set of weakly coupled subsystems. It is very important to know which set of variables will be solved by each subsystem. Then the global iteration process starts with an initial guess of the solution of the original system and is carried out repeatedly until satisfactory convergence is achieved. However, in each subsystem, waveform relaxation operates on groups of function approximations (waveforms) rather than on groups of real values. During an iteration, each decomposed subsystem is solved for its assigned unknown variables in the given time interval by using the approximate waveforms of its neighbor subsystems.

Therefore the general structure of a WR algorithm for analyzing the system in a given time interval $[t_0, t_f]$ consists of two major processes, namely the *assignment-partitioning* process and the *relaxation* process. In the assignment-partitioning process, each unknown variable is assigned to a subsystem in which it is involved. Then the system is partitioned into P subsystems of equations, each of which may have only differential equations or only algebraic equations or both. The relaxation process is an iterative process. For simplicity we shall consider two most commonly used types of relaxation, namely the *Jacobi* and the *Gauss-Seidel* relaxations. The relaxation process starts with an initial guess of the waveform solution of the original dynamical equations in order to initialize the approximate waveforms of the decoupling vectors.

Let us consider a dynamical system which can be described by a system of differential equations of the form,

$$F(t, y, \dot{y}, u) = 0, \qquad y(t_0) = y_0, \tag{2.1}$$

where $F : R \times R^N \times R^N \times R^r \rightarrow R^N$ is a nonlinear function, $y(t) \in R^N$ is the vector of unknown variables, $\dot{y} \in R^N$ is the vector of the time derivatives of the unknown variables, and $u(t) \in R^r$ is the input vector.

Without loss of generality, we can rewrite Eq.(2.1) as follows :

$$F_i(t, y_i, y_i, d_i, u) = 0, \qquad y_i(t_0) = y_{i0}, \tag{2.2}$$

where, for $i = 1, 2, \cdots, P$, $y_i \in R^{n_i}$ is the subvector of the unknown variables assigned to the ith partitioned subsystem, $F : R \times R^{n_i} \times R^{2N-2n_i} \times R^r \rightarrow R^{n_i}$ and d_i is the decoupling vector which contains all the unknown

components of $\dot{y}$ which are not in $\dot{y}_i$ and all time derivative components of $\dot{y}$ not in $\dot{y}_i$. Now, if we consider the decoupling vector of each subsystem in Eq.(2.2) as an input vector, we can solve Eq.(2.1) for a given time interval by solving iteratively P independent subsystems.

Let the superscript k denote the iteration count. Then the general structure of a WR algorithm for a given subinterval $\tau_q=[t_q, t_{q+1}]$ is formally described as follows.

Step 0 : (Assignment-partitioning process)
Assign the unknown variables to Eq.(2.1) and partition Eq.(2.1) into P subsystems of equations as given by Eq.(2.2).

Step 1 : *(Initialization of the relaxation process)*
Set $k=1$ and guess an initial waveform and neighbor waveforms.

Step 2 : *(Solving the decomposed system at the kth iteration)*
Solve Eq.(2.2). For iterations like Gauss-Seidel, the waveform solution obtained by solving one decomposed subsystem is immediately used to update its approximate waveform used by other subsystems. For Jacobi iterations, all approximate waveforms are updated at the beginning of the next iteration.

Step 3 : *(Iteration Process)*
Set $k=k+1$ and go to Step 2.

3. Applications and Results

3.1. Test Problem 1

Suppose we want to solve the following equations:

$$\dot{x}_1 = x_2 , \tag{3.1a}$$

$$\dot{x}_2 = -x_1 , \tag{3.1b}$$

where

$$x_1^0 = 0 ,$$

$$x_2^0 = 1 . \tag{3.2}$$

Any ODE integrator can be used to solve Eq.(2.2). To solve Eq.(3.1), we use the well-known Crank-Nicolson method.

This problem is a typical example solved by parallel waveform relaxation. When we solve Eqs.(3.1a) and (3.1b) concurrently, we can use two transputers. Since the problem is simple but has highly coupled variables, serial WR can solve the above equations faster than parallel WR.

The results are shown in Fig. 1 and Fig. 2. Both figures show that as global iteration goes on, approximate waveforms converge to the exact solution.

3.2. Test Problem 2 (Reactor Point Kinetics Equations)

The second test problem is reactor point kinetics

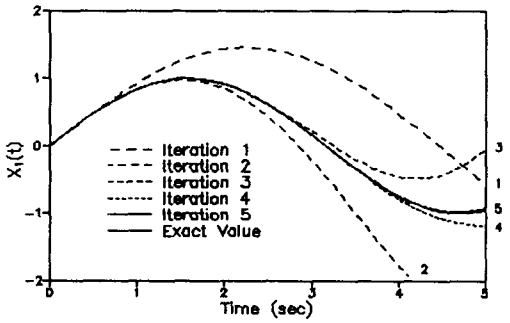

Fig. 1a. The Waveform of X_1 of Test Problem 1 : Serial Calculation

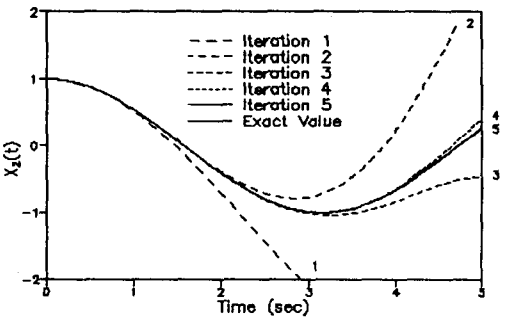

Fig. 1b. The Waveform of X_2 of Test Problem 1 : Serial Calculation

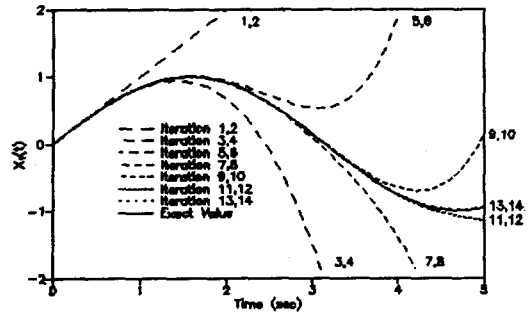

Fig. 2a. The Waveform of X_1 of Test Problem 1 : Parallel Calculation

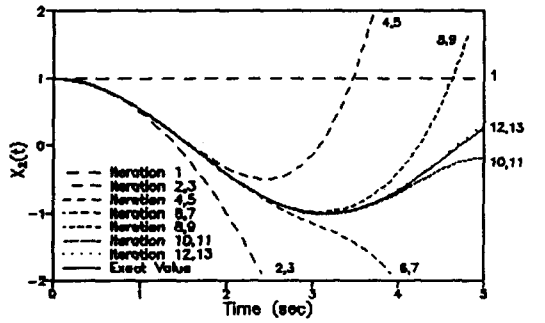

Fig. 2b. The Waveform of X_2 of Test Problem 1 : Parallel Calculation

equations with six groups of delayed neutron precursors[6].

$$\frac{dn}{dt} = \frac{\rho-\beta}{l} n + \sum_{i=1}^{6} \lambda_i C_i , \qquad (3.3)$$

$$\frac{dC_i}{dt} = -\lambda_i C_i + \frac{\beta_i}{l} n, \quad i=1,2,\cdots,6 . \qquad (3.4)$$

Reactor parameters are shown in Table 1.

Table 1. Reactor Parameters Used for Numerical Tests

i	β_i	λ_i
1	0.000266	0.0127
2	0.001491	0.0317
3	0.001316	0.115
4	0.002849	0.311
5	0.000896	1.4
6	0.000182	3.87
l		2×10^{-5} sec
β		0.007

For solving Eqs.(3.3) and (3.4), as an example, we construct integrator as follows. We rewrite Eq.(3.3) as

$$\frac{dx_i}{dt} = a_i x_i(t) + b_i(t) \quad i=1,2,\cdots,7. \qquad (3.5)$$

For a given time interval Δt, we assume $b_i(t)$ in Eq. (3.5) as follows :

$$b_1(t) = b_{10} + b_{11}t + b_{12}t^2 + b_{13}t^3 , \qquad (3.6a)$$

$$b_i(t) = b_{i0} + b_{i1}t + b_{i2}t^2 + b_{i3}t^3 + b_{i4}\exp(a_1 t) , \qquad (3.6b)$$

$$i=2,\cdots,7, \quad t\in[t_0, t_0+\Delta t]$$

Integrating Eq.(3.5) analytically,

$$x_i(t) = \exp(a_i t)\left[\int \exp(-a_i t) b_i(t) dt + D_i \right] . \qquad (3.7)$$

So using Eqs.(3.5), (3.6), and (3.7), we obtain the following solution for the neutron density equation and delayed neutron precursors equations,

$$x_1(t) = -\left(\frac{b_{10}}{a_1} + \frac{b_{11}}{a_1^2} + \frac{2b_{12}}{a_1^3} + \frac{6b_{13}}{a_1^4} \right) - \left(\frac{b_{11}}{a_1} + \frac{2b_{12}}{a_1^2} + \frac{6b_{13}}{a_1^3} \right) t - \left(\frac{b_{12}}{a_1} + \frac{3b_{13}}{a_1^2} \right) t^2 - \left(\frac{b_{13}}{a_1} \right) t^3 + D_1 \exp(a_1 t) . \qquad (3.8)$$

$$x_i(t) = -\left(\frac{b_{i0}}{a_i} + \frac{b_{i1}}{a_i^2} + \frac{2b_{i2}}{a_i^3} + \frac{6b_{i3}}{a_i^4} \right) - \left(\frac{b_{i1}}{a_i} + \frac{2b_{i2}}{a_i^2} + \frac{6b_{i3}}{a_i^3} \right) t - \left(\frac{b_{i2}}{a_i} + \frac{3b_{i3}}{a_i^2} \right) t^2 - \left(\frac{b_{i3}}{a_i} \right) t^3 + \frac{b_{i4}}{a_1 - a_i} \exp(a_1 t) + D_i \exp(a_i t), \quad i=2,\ldots,7. \qquad (3.9)$$

where D_i can be calculated from initial conditions.

We use four transputers to solve Eqs.(3.3) and (3.4) concurrently. Transputer No. 1 calculates Eq.(3.3), and No. 2, 3, and 4 calculate Eq.(3.4) (one transputer calculates two delayed neutron precursors equations, respectively).

Three reactivity insertions are considered : two prompt subcritical cases with $\rho=0.003$ and $\rho=0.0055$, and one prompt supercritical case with $\rho=0.008$ and $\varepsilon=10^{-6}$. The values of $n(t)$ obtained with WRM are compared to those obtained with

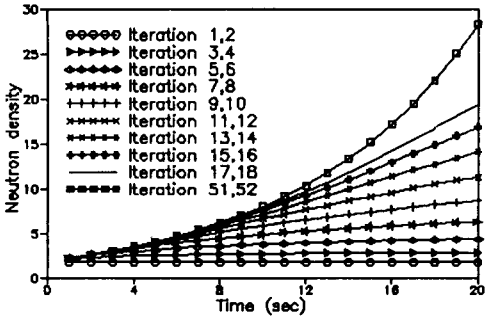

Fig. 3a. The Waveform of n of Test Problem 2 : ρ=0.003, Δt=1.0sec

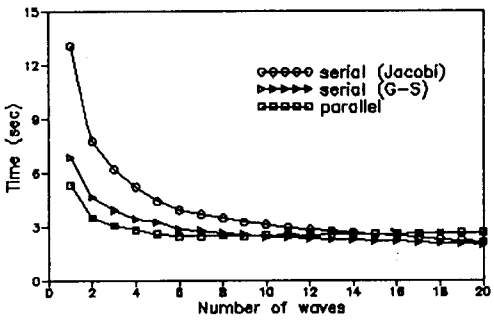

Fig. 4a. Comparison of Time of Test Problem 2 : ρ=0.003, Δt=1.0sec

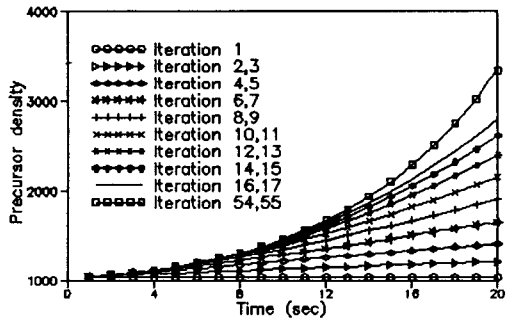

Fig. 3b. The Waveform of C_1 of Test Problem 2 : ρ=0.003, Δt=1.0sec

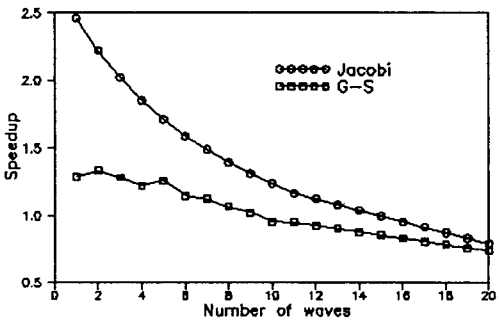

Fig. 4b. Comparison of Time of Test Problem 2 : ρ=0.003, Δt=1.0sec

Table 2. Comparison of WRM with ADM with $\rho=\rho_0$

ρ_0		Method	Δt	n(t)	
			t = 1sec	t = 10sec	t = 20sec
	Exact		2.2098	8.0192	28.297
0.003	ADM	1.0	2.2098	8.0190	28.296
	WRM	0.01	2.2098	8.0192	28.296
	WRM	0.1	2.2099	8.0192	28.297
	WRM	1.0	2.2099	8.0194	28.298
			t = 0.1sec	t = 2sec	t = 10sec
	Exact		5.2100	43.025	1.3886E5
0.0055	ADM	0.1	5.2100	43.025	1.3886E5
	WRM	0.01	5.2100	43.025	1.3886E5
	WRM	0.1	5.2127	43.028	1.3887E5
			t = 0.01sec	t = 0.1sec	t = 1sec
	Exact		6.2029	1.4104E3	6.1634E23
0.008	ADM	0.01	6.2029	1.4104E3	6.1633E23
	WRM	0.01	6.2029	1.4104E3	6.1636E23
	WRM	0.05		1.4148E3	6.3571E23

Adomian's decomposition method (ADM)[6] in Table 2. In Table 2, WRM with a large time step gives less accurate solutions compared to ADM. This is due to the characteristics of ADM that in ADM the solution is represented in a higher order polynomial within a time step. Figure 3 shows the waveforms of n and C_i for $\rho = 0.003$. At a given reactivity, computing time and speedup in various numbers of waves are presented in Fig. 4.

3.3. Test Problem 3 (KMRR Dynamics)

The waveform relaxation method is applied to the dynamics of Korea Multipurpose Research Reactor (KMRR, or Hanaro), and Runge-Kutta method is used as integrator. KMRR is an open-tank-in-pool type research reactor of 30 MWth, operating at Korea Atomic Energy Research Institute. The plant model used in this paper consists of 39 first-order nonlinear dynamics equations developed as a simulator called KMRRSIM[7]. The plant model includes 2-point kinetics model for core and reflector regions with 6-group delayed neutrons and 9-group photo-neutrons. The thermal-hydraulic model includes flow loop and pump characteristics.

We use three transputers to solve the KMRR model, that consists of 39 nonlinear differintial equations. Transputer No. 1 solves 13 equations, that consist of 2-point reactor kinetics model, fission product and fuel power, fuel temperature and coolant temperature through the reactor channel model. Transputer No. 2 solves 14 equations, composed of iodine and xenon model and primary coolant heat exchanger model. Transputer No. 3 solves 12 equations, composed of temperatures in the intermediate pipings from the reactor to the heat exchanger, reflector cooling system and reflector heat exchanger. The schematic diagram of variable communication is shown in Fig. 5.

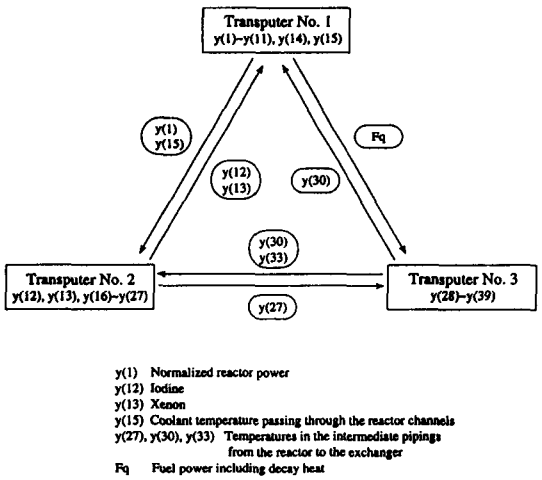

Fig. 5. Schematic Diagram of Variable Communication in Parallel Computation

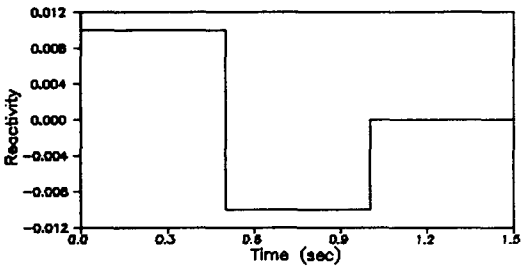

Fig. 6a. Reactivity $\rho_1(t)$ of Test Problem 3 (KMRR)

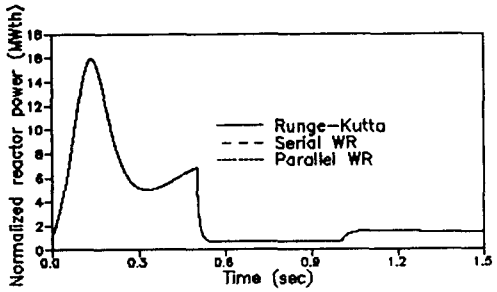

Fig. 6b. Comparison of Reactor Power of Test Problem 3 (KMRR)

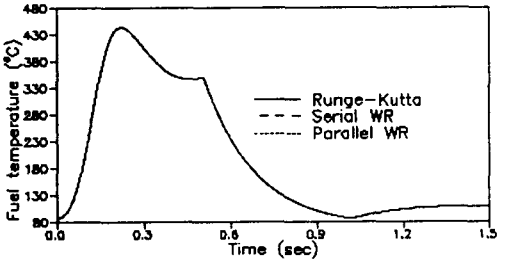

Fig. 6c. Comparison of of Fuel Temperature of Test Problem 3 (KMRR)

Three reactivity insertions are considered : $\rho(t) = \rho_1(t)$ (a series of step functions), $\rho_2(t)$ (sine function), and $\rho_3(t)$ (constant, i.e., $\rho(t) = \rho_0$). Since the heat transfer model is not stiff in comparison to the neutron kinetics model, time steps of Transputer No. 2 and 3 are larger than that of Transputer No 1. The time step and error criterion used are $\Delta t = 0.001$sec (Transputer No. 1), $\Delta t = 0.01$ (No. 2 and 3), and $\varepsilon = 10^{-7}$, respectively. The results are presented in Figs. 6, 7, and 8. The three methods give practically the identical results.

4. Conclusions

In this work, the waveform relaxation method was implemented on a transputer network and applied to three test problems. Our numerical results show that computing time strongly depends on the number of decoupling vectors, because if decoupling vectors increase then more global iterations are needed to converge to the error criteria.

In Test Problem 2 of reactor point kinetics, since the system is partitioned into four subsystems but

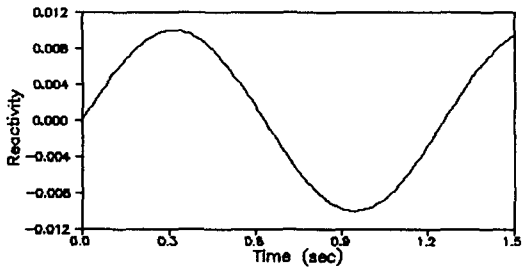

Fig. 7a. Reactivity $\rho_2(t)$ of Test Problem 3 (KMRR)

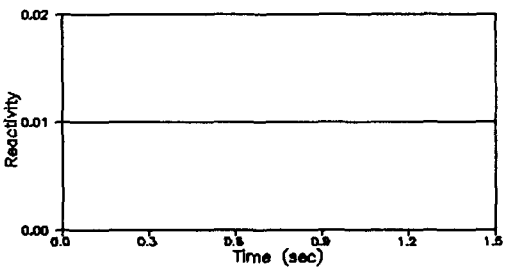

Fig. 8a. Reactivity $\rho_3(t)$ of Test Problem 3 (KMRR)

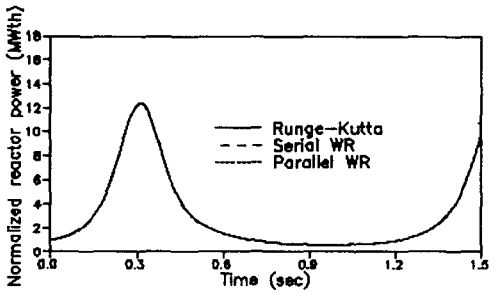

Fig. 7b. Comparison of Reactor Power of Test Problem 3 (KMRR)

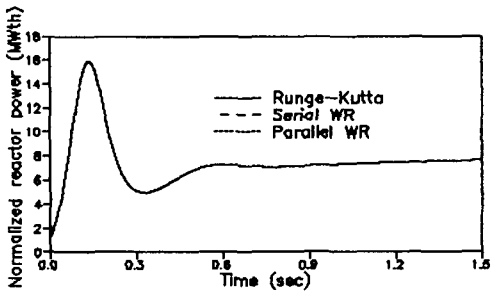

Fig. 8b. Comparison of Reactor Power of Test Problem 3 (KMRR)

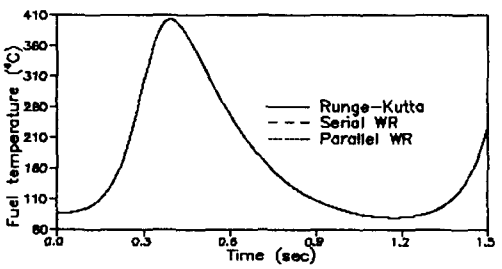

Fig. 7c. Comparison of of Fuel Temperature of Test Problem 3 (KMRR)

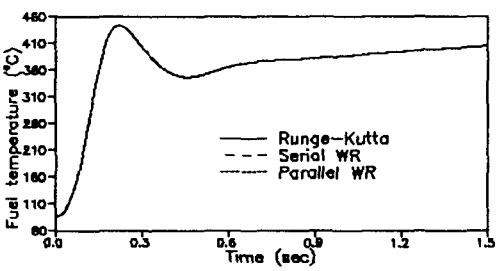

Fig. 8c. Comparison of of Fuel Temperature of Test Problem 3 (KMRR)

Table 3. Comparison of Computing Time for Test Problem 3 (KMRR)

$\rho(t)$	Method	time(sec)	speedup
$\rho_1(t)$[a]	Runge-Kutta	4.869	
	Serial WR	7.817	
	Parallel WR	4.526	1.076[b], 1.727[c]
$\rho_2(t)$[a]	Runge-Kutta	5.262	
	Serial WR	8.603	
	Parallel WR	5.312	0.991[b], 1.620[c]
$\rho_3(t)$[a]	Runge-Kutta	4.838	
	Serial WR	7.756	
	Parallel WR	4.464	1.084[b], 1.737[c]

[a] see Figs. 6, 7, and 8.
[b] with respect to Runge-Kutta
[c] with respect to Serial WR

these variables are highly coupled, many global iterations are needed. If we solve the equations using serial WR algorithm, we do not need times that are consumed in communicating updated decoupling vectors from one subsystem to other subsystems. But in using a parallel WR algorithm, communication times are consumed in order to update decoupling vectors and solve independent subsystems concurrently. So in solving the equations, if we communicate decoupling vectors too frequently, the parallel WR solver requires a little longer computing time than the serial WR solver. The results of the test problems show that there is an optimal number of waves, so to speak, optimal number of communication frequency.

If we decompose a system into several subsystems, we can allocate different time steps in each subsystem. In Test Problem 3, since the heat transfer model is not stiff in comparison to the neutron kinetics model, a large time step is used and contributed to speedup. This is a special feature of the WR method, which cannot be applied to conventional Runge-Kutta method. Thus, the WR method is a general method that can be flexible and implemented on a parallel computer system. However, the example results on the KMRR dynamics indicate that its parallel performance is only modest.

We should note the following if we solve large dynamical systems using WR algorithm. First, we should consider the kind of integrators, although WR algorithm allows any integrators. Second, the number of transputers, which is the same as that of subsystems. Using more transputers does not necessarily mean higher speedup. Third, the method which divides a large system into subsystems. If we divide the system inappropriately, then the variables related between subsystems are highly coupled, so we cannot get good speedup.

References

1. A.R. Secchi, M. Morari and E.C. Biscaia JR, "The Waveform Relaxation Method in the Concurrent Dynamic Process Simulation," Computers and Chem. Eng., 17, 683 (1993).
2. E. Lelarasmee, A.E. Ruehli and A.L.S. Vincentelli, "The Waveform Relaxation Method for Time-Domain Analysis of Large Scale Integrated Circuits," IEEE Trans. on CAD Integrated Circuit Sys., CAD-1, 131 (1982).
3. A.R. Newton, "Techniques for the Simulation of Large-Scale Integrated Circuits," IEEE Trans. Circuits Syst., CAS-26, 741 (1979).
4. "Express," Parasoft Corp., Pasadena, CA (1990).
5. U. Miekkala, "Dynamic Iteration Methods Applied to Linear DAE Systems," J. Comput. Appl. Math., 25, 133 (1989).
6. Y.C. Cho, "Adomian's Decomposition Method Applied to the Reactor Kinetics," M.S. Thesis, KAIST (1992).
7. T.W. Noh et al, "Korea Multipurpose Research Reactor," Technical Report, KM-031-400-02, Korea Atomic Energy Research Institute (1989).