Journal of the Korean Nuclear Society
Volume 28, Number 5, pp. 452~457, October 1996

Adomian Decomposition Method for Point Reactor Kinetics Problems

Young Chul Cho* and Nam Zin Cho
Korea Advanced Institute of Science and Technology
Department of Nuclear Engineering
373-1 Kusong-dong, Yusong-gu, Taejon 305-701, Korea

(Received January 29, 1996)

Abstract

A system, such as a reactor point kinetics equation, can be solved with *Adomian Decomposition Method* (ADM) which uses the notion that all solutions and operators can be expressed as an infinite sum of those basis states, like Adomian polynomials. In this work, ADM is applied to point reactor kinetics equations for step reactivity insertion, ramp input of reactivity, and nonlinear feedback cases without linearization approximation. The results of ADM are more accurate and faster than those of other existing methods, even though we use comparatively large time step sizes.

1. Adomian Decomposition Method

A well-known, long-standing problem in reactor kinetics is the *stiffness* arising from the order-of-magnitude difference between the prompt and delayed neutron lifetimes, which results in the restriction of very small time step increments in numerical solutions to the kinetics equations. There have been a number of methods, e.g. Stiffness Confinement Method[1] (SCM), θ Weighting Method[2], suggested to avoid the difficulty, but they generally involve some approximations or apply effectively only to certain types of problems. But with Adomian Decomposition Method[3,4], we can use large time step increments and get an accurate solution very fast.

Consider the general form

$$\hat{H}u(t) = h(t), \tag{1}$$

where $\hat{H}$ represents a general nonlinear ordinary differential operator involving both linear and nonlinear terms, including d/dt, u(t) is the solution that we want to know, like $n(t)$ and $C_i(t)$'s, and $h(t)$ is the source term.

Supposing that $\hat{H}$ may be decomposed into $\hat{H}_0$ and $\hat{H}_1$, and introducing the separation parameter λ gives

$$(\hat{H}_0 + \lambda \hat{H}_1)u(t) = h(t), \tag{2}$$

where $\hat{H}_0$, is a linear and invertible part of $\hat{H}$, and $\hat{H}_1$ is the remaining part of $\hat{H}$, i.e., $\hat{H} - \hat{H}_0$ Denoting

$$\hat{H}_1 u = A(u),$$

and supposing that

$$A(u) = \sum_{i=0}^{\infty} \lambda^i A_i, \tag{4}$$

$$u = \sum_{i=0}^{\infty} \lambda^i u_i, \tag{5}$$

gives

$$\hat{H}_0 \sum_{i=0}^{\infty} \lambda^i u_i + \sum_{i=0}^{\infty} \lambda^{i+1} A_i = h(t). \tag{6}$$

By arranging Eq. (6) according to the power of λ and from Eq. (3) we get

$$\hat{H}_0 u_0 = h(t), \tag{7}$$

$$\hat{H}_0 u_j = -A_{j-1} = -\hat{H}_1 u_{j-1}, \quad j = 1, 2, \cdots \tag{8}$$

*Present address : Korea Atomic Energy Research Institute.

From Eqs. (7) and (8) we have u_0 and u_j. We can evaluate A_j from Eqs. (3) and (4),

$$A_j = \frac{1}{j!}\frac{d^j A(u)}{d\lambda^j}\bigg|_{\lambda=0} = \frac{1}{j!}\frac{d^j \hat{H}_1 u}{d\lambda^j}\bigg|_{\lambda=0} \qquad (9)$$

A_j polynomials are expressed as

$$A_0 = A(u_0),$$
$$A_1 = u_1(d/du_0)A(u_0),$$
$$A_2 = u_2(d/du_0)A(u_0) + (u_1^2/2!)(d^2/du_0^2)A(u_0),$$
$$A_3 = u_3(d/du_0)A(u_0) + u_1u_2(d^2/du_0^2)A(u_0) + (u_1^3/3!)(d^3/du_0^3)A(u_0),$$
$$\vdots$$

By letting $\lambda = 1$ and with $|u_{n-1}/\sum_i^{n-1} u_i| \le \varepsilon$, the n-term partial sum

$$\phi_n = \sum_i^{n-1} u_i, \qquad (10)$$

will be the approximate solution. Here, ε is the small number chosen reasonably.

1.1 Example

Let us consider the following nonlinear example.

$$du/dt = u^2, u(0) = 1 \quad .$$

The exact solution of the above equation is

$$u(t) = \frac{1}{1-t}, u(1/2) = 2 \quad .$$

Here, we may define $\hat{H}_0 u$ and $\hat{H}_1 u$

$$\hat{H}_0 u = \frac{du}{dt}, \quad \hat{H}_1 u = -u^2 \quad .$$

In this case, the formulation is used for the evaluation of A_j. We evaluate the form of A_j as

$$A_0 = \hat{H}_1 u|_{\lambda=0} = -u_0^2,$$
$$A_1 = \frac{d\hat{H}_1 u}{d\lambda}\bigg|_{\lambda=0} = \frac{d\hat{H}_1 u}{du}\frac{du}{d\lambda}\bigg|_{\lambda=0} = -2u(\lambda)\frac{du}{d\lambda}\bigg|_{\lambda=0} = -2u_0u_1,$$
$$A_2 = -u_1^2 - 2u_0u_2.$$
$$\vdots$$
$$A_j = -\sum_{i=0}^{j} u_i u_{j-i},$$

and evaluate u_j from $\hat{H}_0 u_j = -A_{j-1}$ as

$$u_0 = u(0), u_1 = u(0)^2 t, u_2 = u(0)^3 t^2, u_3 = u(0)^4 t^3$$

At t = 1/2 we have

$$u_0 = 1, u_1 = 0.5, u_2 = 0.25, u_3 = 0.125$$

and the n-term approximation

$$\phi_1 = 1.0, \phi_2 = 1.5, \phi_3 = 1.75, \phi_4 = 1.875$$

that approaches the exact solution 2.

1.2 Calculational Procedure

The procedure to use ADM as a solution scheme to solve some reactor kinetics problems is as follows :

(i) Reformulate the original problem if necessary.
(ii) Determine $\hat{H}_0$ and $\hat{H}_1$
(iii) Evaluate the form of $\hat{H}_1 u_j(t)$
(iv) Evaluate $u_0(t)$, with $u(t_0)$, from $\hat{H}_0 u_0(t) = h(t)$
(v) Evaluate $u_j(t)$ for $j>0$ from the relation $\hat{H}_0 u_j(t) = -\hat{H}_1 u_{j-1}(t)$.
(vi) Add the generated terms to some order $u_j(t)$ to some order n, an acceptable order for the solution thought to be accurate at time $t = t_0 + \Delta t$, to get the solution ϕ_n
That is, $\phi_n(t_0+\Delta t) = \sum_{i=0}^{n-1} u_i(t_0+\Delta t)$.
(vii) If $\phi'(t_0)\,\phi'(t_0+\Delta t) < 0$, set $t_0 = t_0 - \Delta t$, and $\Delta t = 0.1\Delta t$, and repeat the processes (iv) through (vi) twenty times (at each iteration, substitute $u(t_0+\Delta t)$ into $u(t_0)$). Then set $\Delta t = 10\Delta t$ and go to process (viii).
(viii) Substitute $u(t_0+\Delta t)$ into $u(t_0)$ and repeat the processes (iv) through (vii) for the next time step until t reaches the final time t_f.

2. Application of ADM to the Point Kinetics

After some mathematical handling, we can apply the ADM to the following point kinetics with six groups of delayed neutron precursors :

$$\frac{dn}{dt} = \frac{\rho-\beta}{l}n + \sum_{i=1}^{6}\lambda_i C_i, \qquad (11)$$
$$\frac{dC_i}{dt} = \frac{\beta_i}{l}n - \lambda_i C_i, \quad i = 1,2,\cdots,6 \quad . \qquad (12)$$

Here, n and $C_i(t)$'s are neutron density and precursors respectively. And generally, in Eq. (11), ρ is a function of time, or of solution we want to know in the case of feedback.

Tests are done with the following physical parameters :

i	1	2	3	4	5	6	
β_i	0.000266	0.001491	0.001316	0.002849	0.000896	0.000182	$\beta=0.007$
λ_i	0.0127	0.0317	0.115	0.311	1.4	3.87	$l=2\times10^{-5}$(sec)

and with the following initial conditions :

$$n(0) = 1,$$
$$C_i(0) = \frac{\beta_i}{l\lambda_i}, \quad i=1,2,\cdots,6 .$$

2.1 Step Reactivity Insertion and Ramp Input of Reactivity

From Eqs. (7), (8), (11), and (12), by substituting ρ_0+r_t for ρ and denoting that

$$(\hat{H}_0)_{qr} = \delta\frac{d}{dt}, \quad q,r=1,2,\cdots,7 \tag{13}$$

$$\hat{H}_1 u = \begin{pmatrix} \left(\frac{r\tau-\beta'}{l}+a+b\tau\right)\hat{n}+\sum_{i=1}^{6}\lambda_i C_i \\ \frac{\beta_q}{l}\hat{n}+(a+b\tau-\lambda_q)\hat{C}_q \end{pmatrix}, \quad q=1,2,\cdots,6 \tag{14}$$

$$h(t) = 0, \tag{15}$$

$$u = \begin{pmatrix} \hat{n} \\ \hat{C}_i \end{pmatrix} = \begin{pmatrix} ne^{(a\tau+b\tau^2/2)} \\ C_i e^{(a\tau+b\tau^2/2)} \end{pmatrix}, \quad i=1,2,\cdots,6 \tag{16}$$

$$\tau = t-t_0, \tag{17}$$

$$\beta' = \beta-\rho_0-rt_0 , \tag{18}$$

we get

$$u_0 = u(0) = (\hat{n}(0),\ \hat{C}_1(0),\ \hat{C}_2(0),\cdots,\ \hat{C}_6(0))^T , \tag{19}$$

$$u_j = \begin{pmatrix} \hat{n}_j \\ \hat{C}_{1,j} \\ \hat{C}_{2,j} \\ \vdots \\ \hat{C}_{6,j} \end{pmatrix} = \tag{20}$$

$$\int_0^\tau ds \begin{pmatrix} \left(\frac{rs-\beta'}{l}+a\right)\hat{n}_{j-1}+\sum_{i=1}^{6}\lambda_i\hat{C}_{i,j-1} \\ \frac{\beta_1}{l}\hat{n}_{j-1}+(a+bs-\lambda_1)\hat{C}_{1,j-1} \\ \frac{\beta_2}{l}\hat{n}_{j-1}+(a+bs-\lambda_2)\hat{C}_{2,j-1} \\ \vdots \\ \frac{\beta_6}{l}\hat{n}_{j-1}+(a+bs-\lambda_6)\hat{C}_{6,j-1} \end{pmatrix}, \quad j\geq 1 .$$

In Eqs. (17) and (18), t_0 is arbitrary.

The stiffness arises from the difference of the magnitude difference between $(rs-\beta)/l$ and λ_q's. So, we can get $n(t)$ and $C_i(t)$'s easily by letting

$$a=\beta'/2l, \tag{21}$$

$$b=-\gamma/2l. \tag{22}$$

The values of a and b are derivd from the idea that the shapes of $n(t)$ and $C_i(t)$'s are similar.

If $r=0$ we get the solution for step reactivity insertion, else if $\rho_0=0$ we get the solution for ramp input of reactivity.

Three step insertions are considered in step reactivity insertion : prompt subcritical case with $\rho=0.003$ prompt critical case with $\rho=0.007$, and one prompt supercritical case with $\rho=0.008$. The values of $n(t)$ obtained with ADM are compared to those obtained with SCM in Table 1.

Table 1. Comparison of ADM with SCM with $\rho=\rho^0$

ρ	Method	Δt (sec)	$n(t)$†			T_{CPU}*
			$t=1$sec	$t=10$sec	$t=20$sec	
0.003	SCM	0.01	0.706	0.165	0.191	
	ADM	1.0	0.0	−0.003	−0.004	0.2
	ADM	2.0		−0.001	0.0	0.1
			$t=0.01$sec	$t=0.5$sec	$t=2$sec	
0.007	SCM	0.01	−0.193	0.133	0.170	0.3
	ADM	0.01	0.002	0.0	0.005	0.0
	ADM	0.05		0.0	0.005	
			$t=0.01$sec	$t=0.1$sec	$t=1$sec	
0.008	SCM	0.01	0.027	−0.106	−0.096	
	ADM	0.01	0.0	0.0	−0.002	0.1
	ADM	0.1		0.0	−0.002	0.0

† Relative % Error to Exact Solution

* CPU Time on SUN4/75 SPARC station

Next we consider a moderately fast ramp input of reactivity. This example is a 0.1$/s ramp input to a reactor core. The values obtained with ADM are compared to those obtained with SCM in Table 2.

Finally, we consider a saw-tooth input shown in Fig 1. In this case we must choose Δt adaptively since we do not know the peak value of neutron density. The results compared with reference (fine time step Runge-Kutta) are shown in Figs. 2 and 3. The comparison of computing times for various time steps is shown in Table 3.

Table 2. Comparison of ADM versus SCM for Moderately Fast Ramp

time (sec)	θ Weighting Δt=0.0001sec	SCM Δt=0.1sec	ADM Δt=0.2sec
2.0	1.3382E00	1.3382E00 (0.000%)*	1.3382E00 (0.000%)*
4.0	2.2283E00	2.2284E00 (0.005%)	2.2284E00 (0.005%)
8.0	4.2781E01	4.2788E01 (0.016%)	4.2785E01 (0.009%)
11.0	1.7919E16	1.9593E16 (9.342%)	1.7922E16 (0.017%)

* Relative % Error to θ Weighting

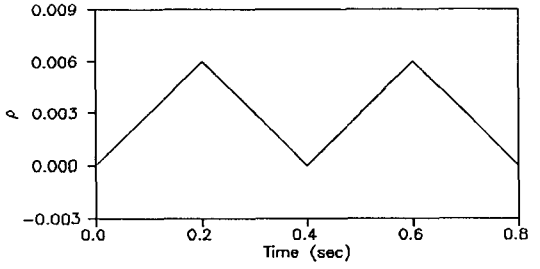

Fig. 1. Varying Input Values of ρ from t=0.0s to t=0.8sec

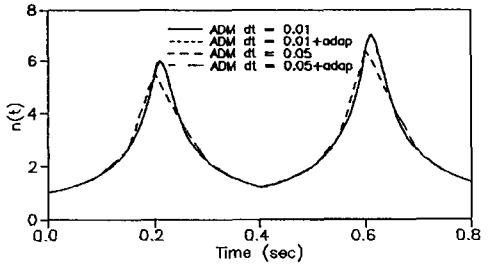

Fig. 2. Neutron Density Obtained for Input in Fig 1. by ADM with various Δt

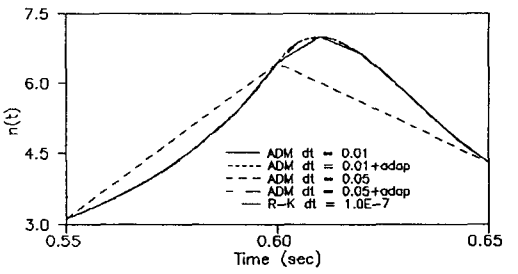

Fig. 3. Neutron Density from t=0.55 to t=0.65 by ADM with various Δt.

Table 3. Comparison of computing times for various time steps

Method	Δt(sec)	T_{CPU}*
ADM	0.05	0.8
	0.005	1.0
	0.05 + adaptive(0.005)	1.1
ADM	0.01	0.8
	0.001	2.1
	0.01 + adaptive(0.001)	1.0
Runge-Kutta	0.001	0.3

* CPU Time on SUN4/75 SPARC station

2.2 Reactivity Feedback

We consider a power feedback case, i.e., $\rho = \rho_0 + \gamma n$. By defining $E = n^2$, we get

$$u_0 = u(0) = (n(0), C_1(0), C_2(0), \cdots, C_6(0))^T \quad (23)$$

$$u_j = \begin{pmatrix} n_j \\ C_{q,j} \end{pmatrix} = \int_0^t ds \begin{Bmatrix} \dfrac{\gamma E_{j-1} + (\rho_0 - \beta) n_{j-1}}{l} + \sum_{i=1}^{6} \lambda_i C_{i,j-1} \\ \dfrac{\beta_1}{l} n_{j-1} + (-\lambda_1) C_{1,j-1} \end{Bmatrix}, \quad j \geq 1, \ q = 1, 2, \cdots, 6. \quad (24)$$

according to the similar process of the previous section.

Two cases are considered with power feedback: one with $\rho_0 = 0$ and $r = 0.003$, and the other with various ρ_0's and r_0's shown in Figures 4 and 5. The results obtained and CPU time taken with ADM and with Runge-Kutta method[5] are presented in Tables 4 and 5. The results of the first case are in Table 4 and those of the second case obtained with ADM are in Figure 6. Comparison of the results with ADM and with Runge-Kutta method is shown in Table 5.

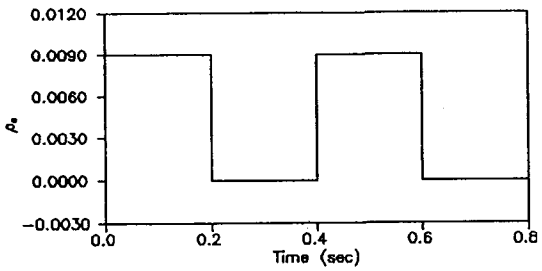

Fig. 4. Varying Input Values of ρ_0 from t=0.0sec to t=0.8sec

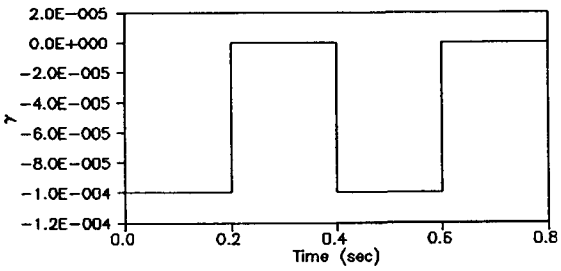

Fig. 5. Varying Input Values of γ from t=0.0sec to t=0.8sec

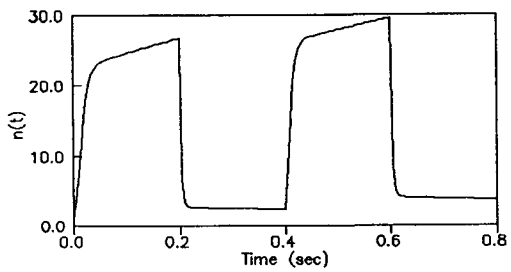

Fig. 6. Neutron Density Obtained for Input in Fig 4. and 5 by ADM with Δt=0.005sec

Table 4. Comparison of ADM with Runge-Kutta Method for $\rho = rn$

Method	Δt(sec)	$n(t)$		T_{CPU}*(sec)
		t=0.01sec	t=0.0117sec	
ADM	1.0E-4	4.9072	1.5948E2	0.1
Runge-Kutta	1.0E-4	4.9072	1.4868E2	0.1
Runge-Kutta	1.0E-5	4.9072	1.5948E2	0.9

* CPU Time on SUN4/75 SPARC station

Table 5. Comparison of ADM with Runge-Kutta Method for $\rho = \rho_0 + rn$

Method	Δt(sec)	$n(t)$		T_{CPU}*(sec)
		t=0.01sec	t=0.8sec	
ADM	1.0E-2	7.4852	3.7608	0.03
ADM	5.0E-3	7.4852	3.7608	0.1
Runge-Kutta	5.0E-3	7.4854	3.7622	0.1
Runge-Kutta	5.0E-4	7.4853	3.7609	1.2

* CPU Time on SUN4/75 SPARC station

3. Conclusions and Discussion

From Tables 1~5 we may say that ADM is more accurate and faster than other methods at least in the cases implemented in this work, even though it uses comparatively large time steps. In reactivity feedback case, even though we did not transform the original problem and did not use any approximations, ADM provides more accurate and quite faster solutions than Runge-Kutta method. The step reactivity insertion case and the ramp input of reactivity case, reformulated according to the analytical solution approach, provide better results than SCM. In all cases studied in this work, ε was chosen to be 10^{-6}.

ADM is proven to be much better than SCM in all cases studied in this work. It can not only use much larger time steps due to the decomposition of the operator and solution, but also compute rapidly for a given time step due to its complete analytic formulation. To solve the power feedback case using SCM, we must make an approximation that within each time step the feedback should be linearized. But

ADM solves the problem without this approximation.

To generalize the method to space-time kinetics, the operator $\hat{H}$ must be extended to $\hat{H}_x+\hat{H}_l$ to include the space kinetics part. The details of the formulation would depend on the additional problem of the type used to solve the spatial diffusion equation.

Acknowledgment

The authors would like to express thanks to Keon Woo Park for his help in performing the numerical calculations.

References

1. Y. A. Chao and A. Attard, "A Resolution of the Stiffness Problem of Reactor Kinetics," Nucl. Sci. Eng. , 90 40-46 (1985)
2. T. A. Porshing, "The Numerical Solution of the Reactor Kinetics Equations by Difference Analogs: A Comparison Methods," WADP-TM-564, U. S. National Bureau of Standards, U. S. Department of Commerce (1966)
3. G. Adomian, Nonlinear Stochastic Operator Equations, Academic Press, Orlando, FL (1986)
4. G. Adomian, "The Decomposition Method for Nonlinear Dynamical Systems," J. Math. Anal. Appl. 120, 370-383 (1986)
5. W. H. Press, B. P. Flannery, S. A. Teukolsky and W. T. Vetterling, Numerical Recipes in C (The Art of Scientific Computing), Cambridge University Press (1988)
6. Young Chul Cho, "Adomian's Decomposition Method Applied to the Reactor Kinetics," MS thesis, Department of Nuclear Engineering, Korea Advanced Institute of Science and Technology, Taejon, February (1992)