Adjoint Solution of Time-dependent Multigroup Diffusion Model with Generalized Temporal and Spatial Boundary Conditions

Tao Liu and Zeyun Wu

Department of Mechanical and Nuclear Engineering, Virginia Commonwealth University Richmond, VA 23219,
liut8@mymail.vcu.edu; zwu@vcu.edu

INTRODUCTION

The understanding of the time-dependent behavior of the neutron population in a nuclear reactor due to either a planned change in the reactor conditions or abnormal operation conditions is of critical importance to the safe operation of nuclear reactors [1]. The description of the neutron flux distribution and the reactor power level and distribution could be obtained by solving the time-dependent neutron forward diffusion equation.

The neutron adjoint diffusion equation could be used for kinetics parameter calculations or in perturbation theory for the sensitivity analysis. For example, the adjoint solution can be used to calculate the change of k_{eff} due to the minor changes in the cross section of reactor core materials. The use of adjoint functions in perturbation theory and the application on reactor problems were first introduced to neutron transport equation by Wigner in 1945 [2]. He demonstrated that the adjoint neutron flux (i.e., the solution to the adjoint transport equation) could be used to assess the effects of perturbing the material properties efficiently in a critical nuclear reactor. Furthermore, Wigner also showed that the adjoint neutron flux could be interpreted as the importance in contributing to the detector responses [3]. However, partly due to exaggerated computation cost, the time-dependent adjoint equation is rarely solved either for diffusion or transport models in practical reactor applications. The typical practice in neutronics calculations is to solve the steady state adjoint equation and use the fundamental mode adjoint function as an approximation for all other dynamic states. This approach basically assumes the adjoint function would have small variations during the state transitions. This assumption works well for most fast transient situations, but will render non-negligible errors for slow transient phenomena in reactor physics if the adjoint solutions heavily relied on sensitivity or kinetics parameter calculations.

In this paper, we will obtain the time-dependent forward and adjoint solution of the one-dimensional two-group neutron diffusion equation for the spatial reactor kinetics problems. The numerical approaches to solve these equations are based on the finite difference method for the spatial discretization and semi-implicit method for the time discretization. For general application purpose, the generalized temporal and spatial boundary conditions for both the forward and adjoint models are considered. To demonstrate the applicability and validation of the presented method, a rod-ejection accident in a one-dimensional reactor problem is considered as a test problem.

The rest of the summary is organized as follows. Section 2 describes the time-dependent one-dimensional two-group neutron forward diffusion equations and adjoint diffusion equations. Section 3 outlines the numerical scheme of the method with semi-implicit method for the time discretization and finite difference method for the spatial discretization. The computational framework of the flux solver and the strategy to improve the computation efficiency are also included in Section 3. The test problem solved by the proposed method and numerical results are presented in Section 4. Finally, the conclusions of the current study and future work are summarized in Section 5.

GOVERNING EQUATIONS

Forward Diffusion Equations

The time-dependent one-dimensional two-group neutron forward diffusion equations coupled with multi-group delayed neutron precursor (DNP) equations are described as follows

$$\begin{cases} \dfrac{1}{\mathrm{v}_1}\dfrac{\partial \phi_1}{\partial t} - \dfrac{\partial}{\partial x}\left[D_1(x,t)\dfrac{\partial \phi_1}{\partial x}\right] + \Sigma_{r,1}\phi_1(x,t) \\ \quad = (1-\beta)\left[\nu\Sigma_{f,1}\phi_1(x,t) + \nu\Sigma_{f,2}\phi_2(x,t)\right] + \sum_{k=1}^{K}\lambda_k C_k(x,t) \\ \dfrac{1}{\mathrm{v}_2}\dfrac{\partial \phi_2}{\partial t} - \dfrac{\partial}{\partial x}\left[D_2(x,t)\dfrac{\partial \phi_2}{\partial x}\right] + \Sigma_{a,2}\phi_2(x,t) \\ \quad = \Sigma_{s,1\to 2}(x,t)\phi_1(x,t) \\ \dfrac{\partial C_k(x,t)}{\partial t} = \beta_k\left[\nu\Sigma_{f1}\phi_1 + \nu\Sigma_{f2}\phi_2\right] - \lambda_k C_k(x,t) \quad k = 1,\cdots,K. \end{cases} \tag{1}$$

where $\phi_g(x,t)$, $D_g(x,t)$, $\Sigma_{a,g}(x,t)$, $\Sigma_{r,g}(x,t)$, $\Sigma_{f,g}(x,t)$ are the space and time dependent neutron flux, diffusion coefficient, absorption cross section, removal cross section, and fission cross section in energy group g, respectively; $\Sigma_{s,1\to 2}(x,t)$ is the scattering cross section from fast group to thermal group; β is the delayed neutron fraction; $C_k(x,t)$ is the group k DNP concentration, which is also a space-time dependent variable, and K is the total number DNP groups; and λ_k is the decay constant for the group k DNP.

Without the loss of generality, the space-time domain of the one-dimensional problem can be considered within the following domain

$$0 \le x \le L, \qquad 0 \le t < T \tag{2}$$

such that the reflective boundary conditions (B.C.) and a prescribed initial condition (I.C.) can be defined as

$$\begin{aligned} &\text{B.C.:} \quad \left.\frac{\partial \phi_g(x,t)}{\partial x}\right|_{x=0} = 0, \ \text{and} \ \left.\frac{\partial \phi_g(x,t)}{\partial x}\right|_{x=L} = 0 \\ &\text{I.C.:} \quad \phi_g(x,0) = \phi_{g0}(x) \end{aligned} \tag{3}$$

For the DNP equations, only initial conditions are needed. They can be defined similarly

$$C_k(x,0) = C_{k0}(x), \qquad k = 1, \cdots, K. \tag{4}$$

Adjoint Diffusion Equations

Based on the definition and properties of adjoint operation [4], the adjoint diffusion equations associated with the forward diffusion equations can be derived as

$$\begin{cases} -\frac{1}{\mathrm{v}_1}\frac{\partial \phi_1^*}{\partial t} - \frac{\partial}{\partial x}\left[D_1(x,t)\frac{\partial \phi_1^*}{\partial x}\right] + \Sigma_{r,1}\phi_1^*(x,t) \\ \quad = (1-\beta)\nu\Sigma_{f,1}\phi_1^*(x,t) + \Sigma_{s,1\to 2}\phi_2^*(x,t) + \beta_k \nu\Sigma_{f1}C_k^*(x,t) \\ -\frac{1}{\mathrm{v}_2}\frac{\partial \phi_2^*}{\partial t} - \frac{\partial}{\partial x}\left[D_2(x,t)\frac{\partial \phi_2^*}{\partial x}\right] + \Sigma_{a,2}(x,t)\phi_2^* \\ \quad = (1-\beta)\nu\Sigma_{f,2}\phi_1^*(x,t) + \beta_k \nu\Sigma_{f2}C_k^*(x,t) \\ -\frac{\partial C_k^*(x,t)}{\partial t} = \left(\sum_{k=1}^{K}\lambda_k\right)\phi_1^*(x,t) - \lambda_k C_k^*(x,t) \qquad k = 1, \cdots, K. \end{cases} \tag{5}$$

The adjoint equation will be solved reversely in the time domain. The boundary conditions and the final condition (F.C.) of the adjoint diffusion equation can also be obtained as

$$\begin{aligned} &\text{B.C.:} \quad \left.\frac{\partial \phi_g^*(x,t)}{\partial x}\right|_{x=0} = 0, \ \text{and} \ \left.\frac{\partial \phi_g^*(x,t)}{\partial x}\right|_{x=L} = 0 \\ &\text{F.C.:} \quad \phi_g^*(x,T) = \phi_{g0}^*(x)\frac{\phi_{g0}(x)}{\phi_g(x,T)} \end{aligned}, \tag{6}$$

where $\phi_{g0}(x)$ is the initial state or steady state forward flux, $\phi_{g0}^*(x)$ is the steady state adjoint flux, and $\phi_g(x,T)$ is the final state forward flux. Similarly, the final conditions for DNP equations are

$$C_k^*(x,T) = C_{k0}^*(x)\frac{C_{k0}(x)}{C_k(x,T)}, \quad k = 1, \cdots, K. \tag{7}$$

NUMERICAL METHOD

Standard numerical approaches are employed to solve the forward and adjoint diffusion equations. The spatial variable is handled by the finite difference method, and the time variable is treated with the semi-implicit method. The mesh size and time step used in the calculation are small enough to minimize the discretization errors. The generalized temporal and spatial boundary conditions for the adjoint equations shown in Eqs. (6) and (7) are considered for a general application.

Initially, an iteration framework is used to solve for the flux in the time-dependent diffusion equations. To reduce the computational cost and improve the computation efficiency, the flux iteration algorithm is replaced with a direct matrix inversion approach that solves the two group fluxes simultaneously.

TEST PROBLEM AND NUMERICAL RESULTS

A rod-ejection accident in a one-dimensional reactor problem is considered as a test problem to demonstrate the applicability and validation of the presented method [5]. As depicted in Fig. 1, the reactor consists of 12 cells with different materials, there are 15 uniform mesh intervals in each cell, so in total 300 meshes are considered. There are three regions in the axial direction: reflector, unrodded fuel, and rodded fuel. Reflective boundary conditions are considered in this problem. The length of the fuel rod is 300 cm. More description of this problem can be found in Ref. [5].

During the rod-ejection accident, the control rod is assumed to be withdrawn from beginning to 4.0 s with a speed of 25 cm/s, later the control rod is inserted from 4.0 s to 10.0 s with the same constant speed. A uniform time step which is 0.5 s is considered in this problem.

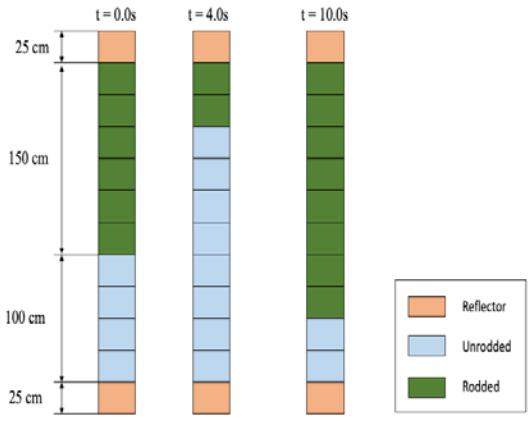

Fig. 1. Geometry of the one-dimensional reactor [5].

A computer code based on MATLAB was developed to solve the two-group forward and adjoint diffusion equations in the one-dimensional system. Before performing the time-

dependent analysis, the steady state condition before the rod-ejection was examined as a criticality calculation. The *k*-eigenvalue mode one-dimensional two-group forward diffusion equation was solved essentially. The k_{eff} obtained at the initial time of the reactor is 0.978821. This value agrees well with the result for the test problem in Ref. [5], which verifies the correct implementation of our code. For the dynamics behavior, the power changes of the slab reactor during the rod-ejection accident was calculated.

Fig. 2 shows the evolution of the normalized mean power into the accident. The normalized power increases with the control rod withdrawal from 0.0 s to 4.0 s, then starts to decrease due to the insertion of the control rod. This result showed in Fig. 3 is obtained from our own diffusion code. Compared to the best result from the moving mesh scheme proposed in Ref. [5], the maximum error is less than 1.13%. Moreover, compared with the results from the neutronic code PARCS which were taken as a reference in Ref. [5], the maximum error is less than 0.93%. All these results indicate our diffusion code has achieved solutions that have a good agreement with the ones in Ref. [5], which verifies the accuracy of our code to a certain degree.

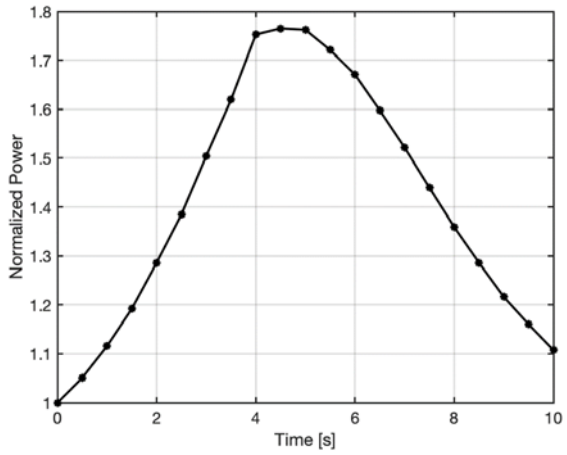

Fig. 2. Normalized power evolution for the 1D reactor.

In the course of calculations, we initially used an iterative approach to calculate the fluxes and noticed it requires more than 1000 iterations at each time step to reach a converged solution with a reasonable tolerance. To reduce the computational cost and improve the computation efficiency, we eliminated the flux iteration algorithm and replaced it with a direct matrix inversion approach that solves the two group fluxes simultaneously. After implementing this method, the total calculation time was reduced from 5.48 s to 0.19 s, which indicates that the direct matrix inversion method indeed improved much the computation efficiency in solving the time-dependent two-group forward diffusion equations.

The adjoint equations were solved similarly in our code using the numerical approaches discussed earlier. Fig. 3 illustrates the adjoint solutions obtained at the initial time by solving the adjoint equation backward in time starting from the final time back to the initial time. These adjoint solutions are compared with the adjoint solutions calculated at steady state. As indicated in Fig. 3, the initial time adjoint solutions agree well with the steady state adjoint solution, which confirms that the numerical method is correctly implemented in the code and obtained reasonably accurate solutions because the adjoint solutions at the initial time are assumed to converge to the steady state adjoint solutions.

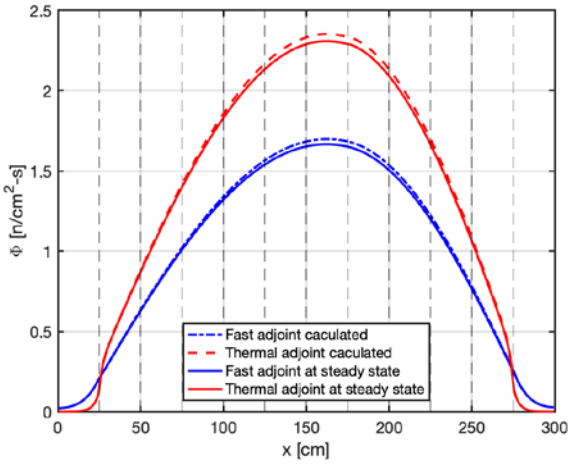

Fig. 3. Adjoint solutions at the initial time.

For transient behaviors, Fig. 4 shows the neutron flux distributions at various times into the rod-ejection accident. As shown in Fig. 4, the neutron flux increases overall with the control rod withdrawal from 0.0 s to 4.0 s, then starts to decrease overall after the insertion of the control rod.

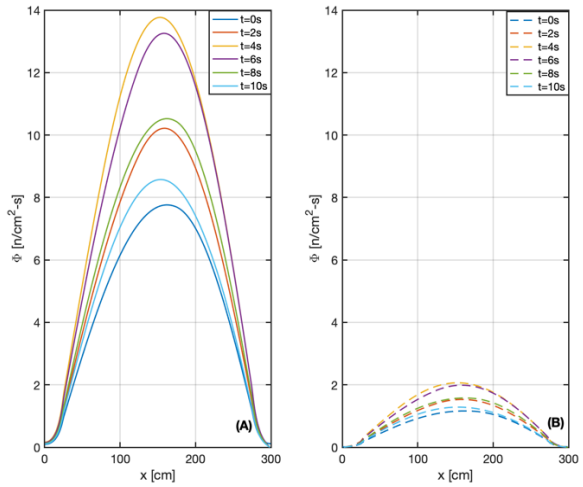

Fig. 4. Fast neutron flux (A) and thermal neutron flux (B) distribution during the rod-ejection accident.

The adjoint solutions indicates the importance of contributing to a specific quantity of interest. Fig. 5 illustrates the fast and thermal adjoint functions corresponding to the criticality status of the reactor at

different transient time steps. It clearly shows the shape and magnitude of the adjoint functions will involve substantial changes along with the rod-ejection procedure, which confirms the values of the time-dependent adjoint calculations.

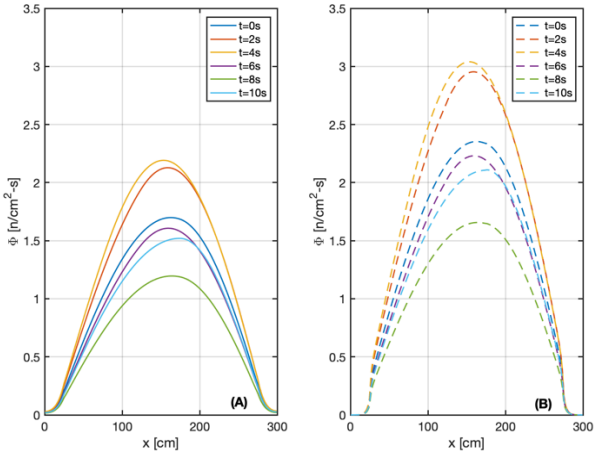

Fig. 5. Fast adjoint (A) and thermal adjoint solution (B) distribution during the rod-ejection accident.

Wigner stated that the adjoint neutron flux could be interpreted as the importance of contributing to the detector response [2]. To better understand the physical meaning of the adjoint neutron flux, we assume there is a neutron detector placed at the middle of the third cell of the reactor core.

The typical response R is chosen for the reading of detector [3], which could be given by the reaction rate

$$R = \sum_{g=1}^{G} \Sigma_{d,g} \phi_g(x,t), \qquad G = 2\,, \tag{8}$$

where $\Sigma_{d,g}$ represents the detector's equivalent reaction cross section at the location x_d for different groups of neutron flux. The $\Sigma_{d,1}$ and $\Sigma_{d,2}$ in this problem are assumed to be 0.01 and 0.1, respectively. The adjoint source is defined as a delta function in this problem. Then the adjoint equation with adjoint source become

$$-\frac{1}{\mathrm{v}}\frac{\partial \phi_g^*}{\partial t} - \frac{\partial}{\partial x}\left[D_g(x,t)\frac{\partial \phi_g^*}{\partial x}\right] + \Sigma_{r,g}\phi_g^*(x,t) = S + \Sigma_{d,g}\,. \tag{9}$$

Fig. 6 shows the fast adjoint solutions and thermal adjoint solution variation during the rod-ejection accident, which represents the neutron importance distribution for the designated detector response R.

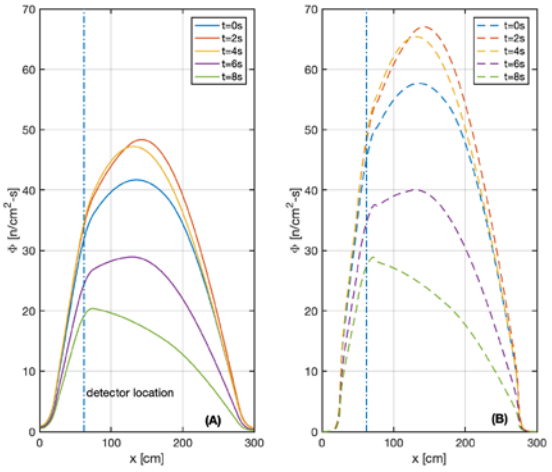

Fig. 6. Fast adjoint (A) and thermal adjoint (B) for R.

CONCLUSIONS

In this work, a MATLAB code based on the time-dependent one-dimensional two-group neutron forward and adjoint diffusion models are developed for spatial reactor kinetics problems. The numerical approaches are based on the finite difference method for the spatial discretization and semi-implicit method for the temporal discretization. A generalized temporal and spatial boundary conditions for both the forward and adjoint models are considered. To assess the capability and accuracy of the developed code, it is applied to the rod-ejection accident in a one-dimensional slab reactor. The computational results of the test problem demonstrate that the code is capable of outputting reasonably accurate solutions in both forward and adjoint modes. To reduce the computational cost, the flux solver in the code is improved by eliminating the iteration algorithm. In the future, the time-dependent adjoint solution will be used in the perturbation theory for dynamics sensitivity analysis. For instance, it can be used to calculate the time-dependent effect on sensitivity of k_{eff} due to the minor changes in cross section of reactor core materials.

REFERENCES

[1]. W. M. STACEY, *Nuclear Reactor Physics*, 3rd ed., Wiley-VCH, Weinheim (2018).

[2]. E. P. WIGNER, "Effects of Small Perturbations on Pile Period," CP-3048, Manhattan Project Report (1945).

[3]. D. G. CACUCI, *Sensitivity and Uncertainty Analysis*, Chapman and Hall/CRC, Washington, D.C. (2003).

[4]. G.I. BELL and S. GLASSTONE, *Nuclear Reactor Theory*, Van Nostrand Reinhold Company, New York (1970).

[5]. A. VIDAL-FERRÀNDIZ et al., "Moving meshes to solve the time-dependent neutron diffusion equation in hexagonal geometry," *Journal of Computational and Applied Mathematics*, **291**, 197–208 (2016).