frontiers | Frontiers in Energy Research

TYPE Original Research
PUBLISHED 10 October 2023
DOI 10.3389/fenrg.2023.1284230

OPEN ACCESS

EDITED BY
Yugao Ma,
Nuclear Power Institute of China (NPIC), China

REVIEWED BY
Shanfang Huang,
Tsinghua University, China
Zeyun Wu,
Virginia Commonwealth University, United States
Luteng Zhang,
Chongqing University, China

*CORRESPONDENCE
Tengfei Zhang,
zhangtengfei@sjtu.edu.cn

RECEIVED 28 August 2023
ACCEPTED 15 September 2023
PUBLISHED 10 October 2023

CITATION
Wang D, Xiao W, Du Z and Zhang T (2023), An adaptive time stepping stiffness confinement method for solving reactor dynamics equations.
Front. Energy Res. 11:1284230.
doi: 10.3389/fenrg.2023.1284230

An adaptive time stepping stiffness confinement method for solving reactor dynamics equations

Dan Wang, Wei Xiao, Zhaohui Du and Tengfei Zhang*

School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai, China

The stiffness confinement method (SCM) is frequently employed to solve the reactor dynamics equations because it confines the stiffness of the problem by frequency transformation. However, the balance between the error and efficiency of the SCM has not been well studied. This paper reports the error analysis of the SCM. The error by SCM is derived mathematically and written as integral error, driven by the integral of the frequency interpolation function. An easy-to-implement adaptive time-stepping (ATS) algorithm is proposed based on the error analysis by controlling the neutron flux amplitude error. First, a fine-step PKE is leveraged to estimate the second-order derivative of the flux amplitude-frequency, which is used to predict the error of the neutron flux amplitude. The low cost of solving the PKE incurs a negligible effect on the algorithm's efficiency. Second, based on the error analysis, an error estimator proposed to determine an optimal time-step size for the neutron temporal-spatial equation. With a pre-set error tolerance, the ATS algorithm is exempted from the empirical selection of the time-step size in transient simulations. Numerical tests with TWIGL and modified 2D LMW benchmark problems show that the optimal time-step size effectively confines the local truncation error of the flux amplitude within the pre-set tolerance. The ATS algorithm yields a higher accuracy at a commensurate computational cost than calculations with fixed time-steps.

KEYWORDS

stiffness confinement method, reactor dynamics equation, error analysis, adaptive time stepping algorithm, nuclear reactor

1 Introduction

Reactor dynamics equations (NDEs), such as the neutron temporal-spatial equation (NTSE) and the point kinetics equation (PKE), are employed to predict the transient behaviors of the neutron flux and precursors in a nuclear reactor. In the NDEs, the generation time of prompt and delayed neutrons involves multiple time scales. As these different time scales bring stiffness in the NDE, the time-step size must be sufficiently small to yield accurate results. The fine time-step causes a significant computational burden to the reactor dynamics calculations. Therefore, the stiffness confinement method (SCM) was proposed. In doing so, the SCM introduces the flux frequency and the precursor frequency to decouple the precursor equation from the NDE, thus confining the stiffness to the prompt neutron equation (Chao and Attard, 1985). Next, an exponential solution assumption decomposes the flux amplitude-frequency into the amplitude and shape frequencies (Park and Joo, 2015). By these means, the problem is transformed into a dynamic eigenvalue problem (EVP) that exhibits a similar equation form as that for the steady-state eigenvalue

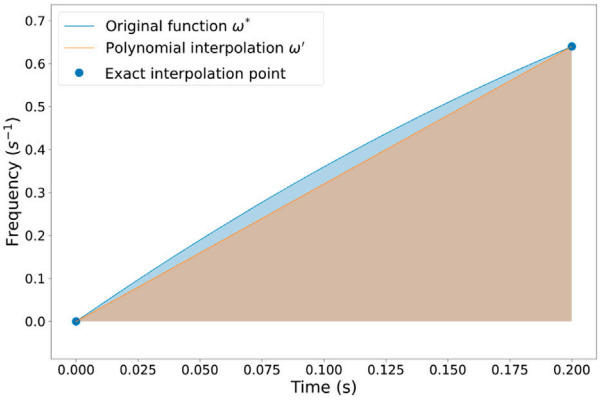

FIGURE 1
Illustration of integral error and zero-point bias with linear approximation.

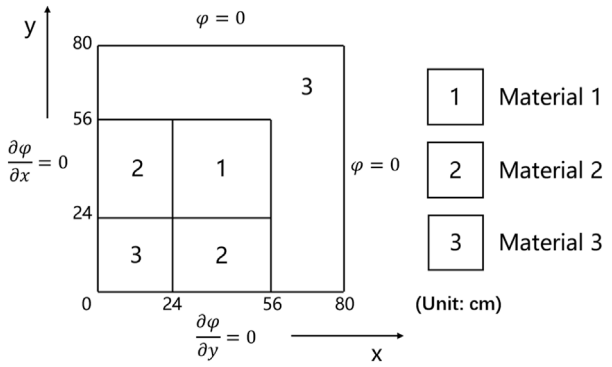

FIGURE 2
Layout of TWIGL benchmark problem.

problem. Therefore, the dynamic EVP can be solved efficiently by power iteration and the non-linear iteration algorithm.

The SCM has already been applied in the PKE and the diffusion and transport NTSEs (Park and Joo, 2015; Tang et al., 2019). It has been demonstrated that the SCM can provide accurate and stable solutions. However, although the SCM enables solving the NTSE with a large time-step size, the computational cost of the NTSE at each time step is still high, especially for transport calculations. The adaptive time-stepping (ATS) algorithm has received much attention for further reducing computational costs. It allows transient solvers to optimally determine the time-step sizes according to the current transient state and the pre-set error tolerance. The algorithm improves the computational efficiency by allocating more time steps in time intervals where necessary. In addition, the ATS algorithm saves the trouble of empirically selecting the time-step size in transient simulations. Hence, ATS algorithms have been applied in solution methods to NTSE, such as the implicit Euler method, the quasi-static method, and the PKE-based SCM (Caron et al., 2017). However, an effective ATS algorithm for the three-dimensional SCM has yet to be developed because of the lack of comprehensive error analysis.

ATS algorithms are generally based on estimating and controlling of the local truncation error. Classically, the local truncation error ε of a numerical method of order q can be expressed as (Tang et al., 2019):

$$\varepsilon(t) = \Theta(t)h^{q+1} + O\left(h^{q+2}\right) \tag{1}$$

where t denotes the time, Θ is the norm of the principal error function, and h is the time-step size. Generally, Θ is estimated with an error bound. The error bound is appropriated via the high-order derivative or even the Jacobian matrix of the solution. For lower complexity, Θ can be assumed as constant between successive time steps and calculated by numerical differentiation. The numerical differentiation is computed using solutions from previous time steps (Caron et al., 2017). This approach does not require extra computation, but the predicted error may deviate significantly from the actual error. The deviation leads to a non-optimal time-step size and affects the efficiency of the ATS algorithm. Another practical ATS algorithm estimates the error by comparing the solution with a reference solution. With gradually reduced time-step size, the solution is rejected until the error between the computed solution and the reference solution reaches the pre-set tolerance. The adaptive time-step size is found by successively comparing solutions with two different time-step sizes and choosing the finer step solution as the reference one (Boffie and Pounders, 2018). Another approach is the embedded pair, which selects the time-step size by embedding low-order methods in the high-order method. For instance, a generalized Runge-Kutta method of fourth-order accuracy embeds a third-order solution to estimate the adaptive time-step size (Zimin and Ninokata, 1998). Although these ATS algorithms are easy to implement, the rejection-acceptance procedure calls for extra computation. Therefore, it is beneficial to develop an ATS algorithm that can efficiently and accurately control the time-step size for the SCM.

In this work, a theoretical error analysis of the SCM is performed. The theoretical analysis produces a mathematical error expression of the SCM. Further, an ATS algorithm is proposed by controlling the error of the neutron flux amplitude. In doing so, a fine-step PKE solver is used to evaluate high order derivatives of the neutron flux amplitude in the error expression. In addition, an error estimator based on the error expression is proposed and examined using benchmark problems.

TABLE 1 Transient cases in TWIGL benchmark problem.

Case	Perturbation
Composite	$\Sigma_{a,2}(\vec{r},t) = \begin{cases} \Sigma_{a,2}(\vec{r},0) \times (1 - 0.2905t) & t \le 0.2s \\ \Sigma_{a,2}(\vec{r},0) \times [1.01167 + 0.05833(t - 0.2)] & 0.4s \ge t > 0.2s \\ \Sigma_{a,2}(\vec{r},0) & 0.5s \ge t > 0.4s \end{cases}$

Note: $\vec{r} \in$ regionofMaterial1.

10.3389/fenrg.2023.1284230

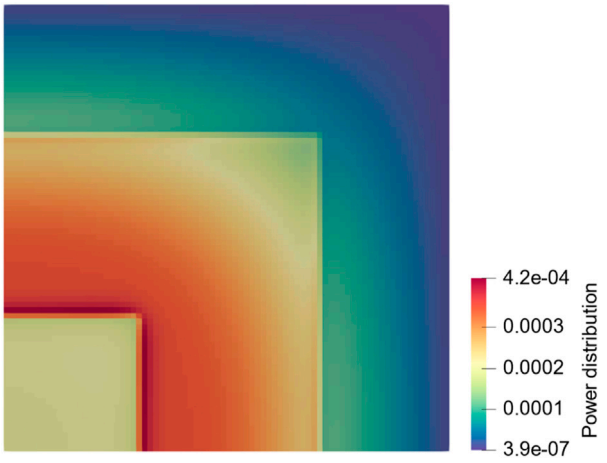

FIGURE 3
Initial power distribution for TWIGL.

It is demonstrated that the ATS algorithm yields a higher accuracy at a commensurate computational cost than calculations with fixed time-steps.

The remainder of the paper proceeds as follows. Section 2 briefly describes the frequency-transformed NTSE and PKE. Solution methods to these equations are introduced in Section 3. In Section 4, the SCM's theoretical error analysis is performed, and an ATS algorithm based on the error analysis is proposed. Section 5 elaborates the coupling between the NTSE solver with the PKE solver. Section 6 illustrates the performance of error estimators and offers comparisons of the efficiency and accuracy between ATS and fixed time-stepping (FTS). Section 7 concludes the paper and points to directions for future research.

2 Frequency-transformation of dynamics models

The derivation of the SCM starts with the definition of the dynamic frequency. The dynamic frequency $\alpha(r,t)$ of a physical quantity $f(r,t)$ is defined as (Chao and Attard, 1985):

$$\alpha(r,t) = \frac{1}{f(r,t)} \frac{\partial f(r,t)}{\partial t} \quad r \in \mathbb{R}^3, t \in \mathbb{R} \tag{2}$$

where r is the spatial variable and t is the time variable. By introducing the dynamic frequency, a composite exponential function is employed to describe $f(r,t)$. Thus, the following exponential form of the solution is obtained:

$$f(r,t) = f(r,t_0) e^{\int_{t_0}^{t} dt' \alpha(r,t')} \tag{3}$$

Transient equations imposed with the dynamic frequency are called frequency-transformed equations. The frequency-transformed equations are solved by discretizing the time variable and searching $\alpha(r,t)$ iteratively.

For simplicity, we investigate the application of the SCM to the NTSE with diffusion approximation. Extending the methodology to neutron transport problems is not arduous (Park and Joo, 2015). Transient multi-group neutron diffusion equations with delayed neutron precursors are given as:

$$\begin{aligned} &\frac{1}{v_g} \frac{d\varphi_g(r,t)}{dt} + \left[-\nabla \cdot D_g(r,t)\nabla + \Sigma_{t,g}(r,t)\right]\varphi_g(r,t) \\ &= \left[\chi_g(r)(1-\beta)Q(r,t) + \sum_{g'=1}^{G} \Sigma_{g'-g}(r,t)\varphi_{g'}(r,t)\right] \\ &\quad + \sum_{i=1}^{I} \chi_{ig}\lambda_i C_i(r,t) \quad g = 1, \cdots, G \end{aligned} \tag{4}$$

$$\frac{dC_i(r,t)}{dt} = \beta_i Q(r,t) - \lambda_i C_i(r,t) \quad i = 1, \cdots, I \tag{5}$$

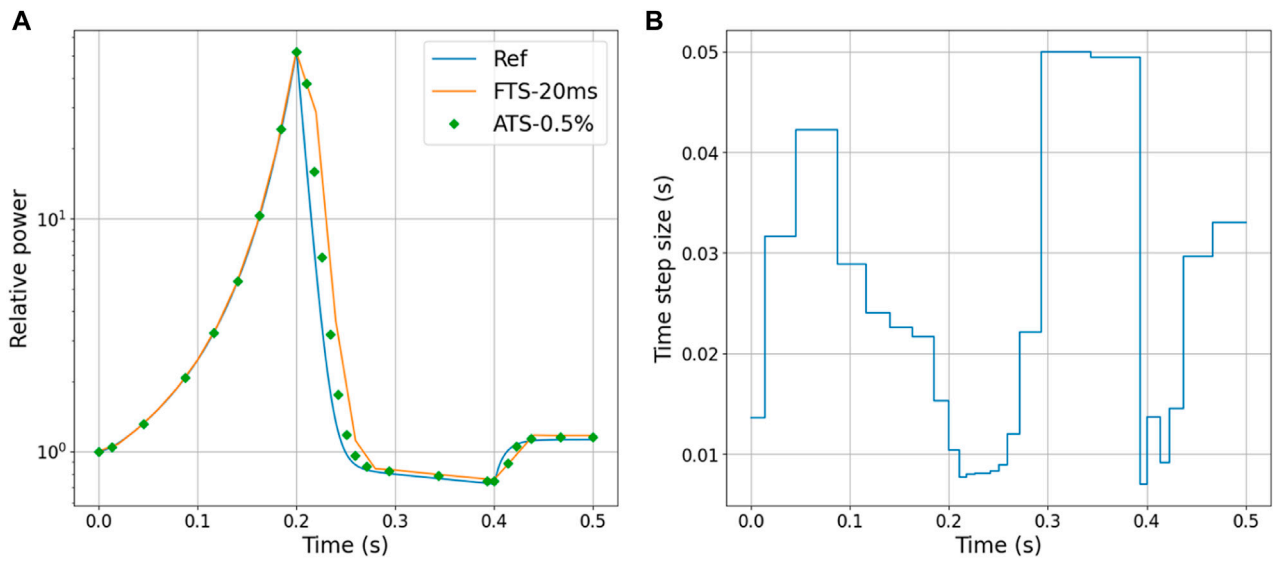

FIGURE 4
Comparison of ATS-0.5% and FTS-20 ms in TWIGL **(A)** Core power **(B)** Time-variant adaptive time-step size.

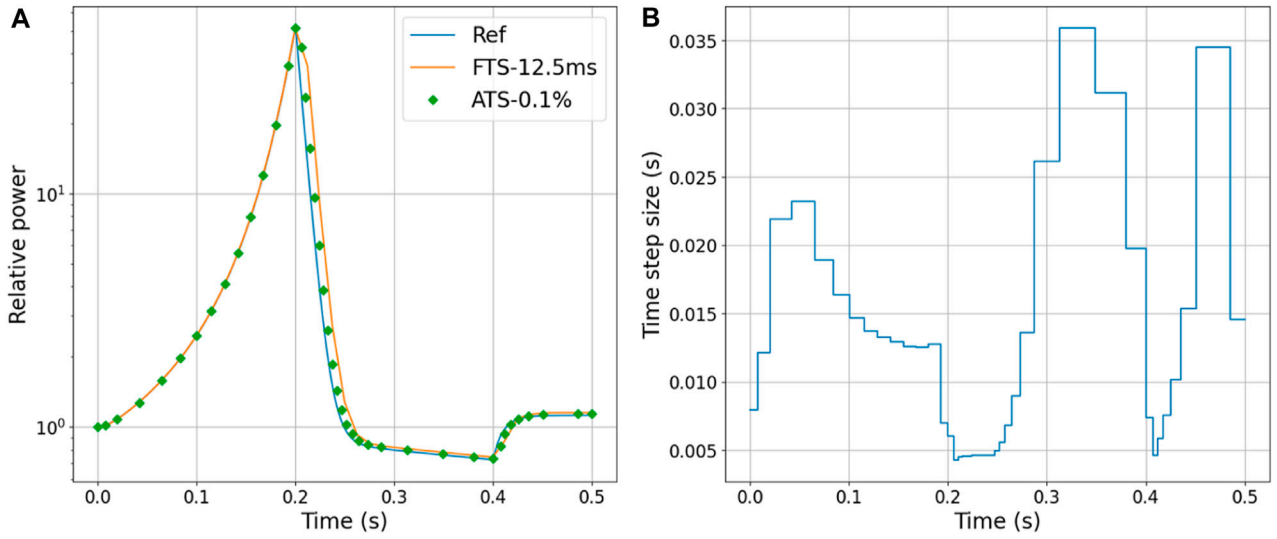

FIGURE 5
Comparison of ATS-0.1% and FTS-12.5 ms in TWIGL **(A)** Core power **(B)** Time-variant adaptive time-step size.

TABLE 2 Comparison of global amplitude error between ATS and FTS in TWIGL.

Time	Reference power	FTS-20 ms error	ATS-0.5% error	FTS-12.5 ms error	ATS-0.1% error
0.2 s	51.222	1.354%	1.504%	0.542%	0.400%
0.4 s	0.725	4.092%	2.514%	2.634%	1.403%
0.5 s	1.123	4.072%	2.412%	2.623%	1.351%
Number of steps	500	25	25	40	41

where $\varphi_g(r,t)$ represents the neutron flux of group g, $C_i(r,t)$ is the delayed neutron precursor concentration of precursor family i, and other notations are conventional. The fission source Q is defined as:

$$Q(r,t) = \sum_{g'=1}^{G} \nu\Sigma_{f,g'}(r,t)\varphi_{g'}(r,t) \tag{6}$$

The neutron flux frequency is introduced to derive the frequency-transformed dynamics model:

$$\omega_g(r,t) \equiv \frac{1}{\varphi_g(r,t)} \frac{\partial}{\partial t}\varphi_g(r,t) \tag{7}$$

where $\omega_g(r,t)$ represents the neutron flux frequency which can be further split as $\omega_g(r,t) = \omega_{S,g}(r,t) + \omega_T(t)$. The flux amplitude-frequency $\omega_T(t)$ represents a global quantity and is dependent only on time; the flux shape-frequency $\omega_{S,g}(r,t)$ is dependent on space, time, and energy. For normalization, a physics-based constraint on the shape-frequency is introduced as (Chao and Attard, 1985):

$$\sum_{g=1}^{G} \int_V dr\kappa\Sigma_{f,g}(r,t)\varphi_g(r,t_0) e^{\int_{t_0}^{t} dt' \omega_{S,g}(r,t')} = P(t_0) \tag{8}$$

where, κ is the heat release per fission reaction, and $P(t_0)$ is the initial total power of the nuclear reactor. The constraint guarantees that the shape-frequency affects only the flux shape, not the total power. The power distribution $q(r,t)$ is defined as:

$$q(r,t) = \sum_{g=1}^{G} \kappa\Sigma_{f,g}(r,t)\varphi_g(r,t_0) \tag{9}$$

The precursor concentration frequency is defined as:

$$\mu_i(r,t) \equiv \frac{1}{C_i(r,t)} \frac{\partial}{\partial t} C_i(r,t) \tag{10}$$

Introducing the frequency-transformation, the transient multi-group neutron diffusion equations and associated delayed neutron precursor equations are rewritten as:

$$\begin{aligned} &\left[-\nabla \cdot D_g(r,t)\nabla + \Sigma_{t,g}(r,t)\right]\varphi_g(r,t) + \frac{\omega_{S,g}(r,t) + \omega_T(t)}{v_g}\varphi_g(r,t) \\ &= \left[\chi_g(r)(1-\beta) + \sum_{i=1}^{I} \frac{\chi_{ig}\beta_i\lambda_i}{\mu_i(r,t) + \lambda_i}\right] Q(r,t) \\ &\quad + \sum_{g'=1}^{G} \Sigma_{g'-g}(r,t)\varphi_{g'}(r,t) \\ &g = 1, \ldots, G \end{aligned} \tag{11}$$

$$\mu_i(r,t)C_i(r,t) = \beta_i Q(r,t) - \lambda_i C_i(r,t) \quad i = 1, \ldots, I \tag{12}$$

Equation 11 is the frequency-transformed temporal-spatial equation. Taking all frequencies in Eq. 11 to be zeros yields the static neutron diffusion equation:

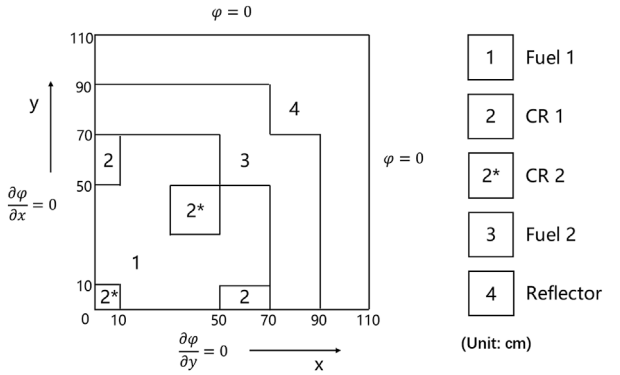

FIGURE 6
Layout of modified LMW-2D benchmark problem.

$$\left[-\nabla\cdot D_g(r,t_0)\nabla+\Sigma_{t,g}(r,t_0)\right]\varphi_g(r,t_0)=\frac{\chi_g(r)}{k_{\text{eff}}}\sum_{g'=1}^{G}\nu\Sigma_{f,g'}(r,t_0)\varphi_{g'}(r,t_0)+\sum_{g'=1}^{G}\Sigma_{g'-g}(r,t_0)\varphi_{g'}(r,t_0)\quad g=1,\cdots,G \tag{13}$$

The adjoint equations corresponding to Eq. 13 are:

$$\left[-\nabla\cdot D_g(r,t_0)\nabla+\Sigma_{t,g}(r,t_0)\right]\varphi_g^{\dagger}(r,t_0)=\frac{\nu\Sigma_{f,g}(r,t_0)}{k_{\text{eff}}}\sum_{g'=1}^{G}\chi_{g'}(r)\varphi_{g'}^{\dagger}(r,t_0)+\sum_{g'=1}^{G}\sum_{g-g'}(r,t_0)\varphi_{g'}^{\dagger}(r,t_0)\quad g=1,\cdots,G \tag{14}$$

The left-hand side of Eq. 14 is the adjoint diffusion-absorption operator, which is self-adjoint, and the right-hand side is the adjoint fission-scattering operator. Eqs 13, 14 are both EVPs, which can be solved by power iteration.

The PKE is a lumped-parameter model used to analyze the dynamic behaviors of the flux amplitude while neglecting the flux shape. As will be shown, this model is beneficial in the error analysis of the SCM. The PKE results by integrating Eqs 4, 5 weighted with the initial adjoint flux. The resulting equations are:

$$\frac{dn(t)}{dt}=\frac{\rho(t)-\beta}{\Lambda}n(t)+\sum_{i=1}^{I}\lambda_i C_i(t) \tag{15}$$

$$\frac{dC_i(t)}{dt}=\frac{\beta_i}{\Lambda}n(t)-\lambda_i C_i(t)\quad i=1,2,...,I \tag{16}$$

in which the notations are conventional thus are neglected for brevity. Readers can find detailed definitions of the dynamic parameters in Appendix-A. Correspondingly, the neutron density frequency for the PKE is be given by:

$$\omega(t)\equiv\frac{1}{n(t)}\frac{dn(t)}{dt} \tag{17}$$

which can also be called the amplitude-frequency. The frequency-transformed point kinetics model is expressed as:

$$\omega(t)=\frac{\rho(t)-\beta}{\Lambda}+\frac{1}{n(t)}\sum_{i=1}^{I}\lambda_i C_i(t) \tag{18}$$

The precursor concentration equations are identical to Eq. 16.

3 Solution method for frequency-transformed equations

Introducing the dynamic eigenvalue k_D in Eq. 11, the equation can be transformed into an EVP:

$$\left[-\nabla\cdot D_g(r,t)\nabla+\Sigma'_{t,g}(r,t)\right]\varphi_g(r,t)=\frac{\chi'_g(r)}{k_D}Q(r,t)+\sum_{g'=1}^{G}\Sigma_{g'-g}(r,t)\varphi_{g'}(r,t)\quad g=1,...,G \tag{19}$$

where, $\Sigma'_{t,g}(r,t)$ is the dynamic total cross-section, and $\chi'_g(r)$ is the dynamic fission spectrum, which are respectively defined as:

$$\sum\nolimits'_{t,g}(r,t)\equiv\Sigma_{t,g}(r,t)+\frac{\omega_{S,g}(r,t)+\omega_T(t)}{v_g} \tag{20}$$

$$\chi'_g(r)\equiv\chi_g(r)(1-\beta)+\sum\nolimits_{i=1}^{I}\frac{\chi_{ig}\beta_i\lambda_i}{\mu_i(r,t)+\lambda_i} \tag{21}$$

The dynamic frequencies rendering the maximum eigenvalue k_D equal to 1are the solutions to Eq. 19. It is noted that the dynamic eigenvalue k_D can be expressed as a non-linear function of the flux amplitude-frequency ω_T. Thus, the dynamic frequencies are solved iteratively using power

TABLE 3 Perturbations in modified LMW-2D benchmark problem.

Region	Perturbation
CR 1	$\Sigma_{a,2}(\bar{r}_2,t)=\begin{cases}\Sigma_{a,2}(\bar{r}_2,0)\times(1-10t)+\Sigma_{a,2}(\bar{r}_1,0)\times 10t & t\le 0.1s\\ \Sigma_{a,2}(\bar{r}_1,0) & 0.1s<t\le 0.4s\\ \Sigma_{a,2}(\bar{r}_2,0)\times\left(\frac{5t}{3}-\frac{2}{3}\right)+\Sigma_{a,2}(\bar{r}_1,0)\times\left(\frac{5}{3}-\frac{5}{3}t\right) & 0.4s<t\le 1.0s\end{cases}$
CR 2	$\Sigma_{a,2}(\bar{r}_{2^*},t)=\begin{cases}\Sigma_{a,2}(\bar{r}_{2^*},0)\times(1-10t)+\Sigma_{a,2}(\bar{r}_1,0)\times 10t & t\le 0.1s\\ \Sigma_{a,2}(\bar{r}_1,0) & 0.1s<t\le 0.2s\\ \Sigma_{a,2}(\bar{r}_{2^*},0)\times\left(\frac{5t}{3}-\frac{1}{3}\right)+\Sigma_{a,2}(\bar{r}_1,0)\times\left(\frac{4}{3}-\frac{5}{3}t\right) & 0.2s<t\le 0.8s\\ \Sigma_{a,2}(\bar{r}_{2^*},0) & 0.8s<t\le 1.0s\end{cases}$

Note: $\bar{r}_1\in$ regionofFuel1, $\bar{r}_2\in$ regionofCR1, $\bar{r}_{2^*}\in$ regionofCR2

iteration and the non-linear iteration algorithms. The non-linear iteration algorithm employed in this paper is the $k-\omega$ iteration, as shown in Algorithm 1:

function SpatialSolverSCM(t_N, h, XS)
  $\{tn\} \leftarrow \{0 : h : t_N\}$ Time steps
  $\tilde{\varphi}_g(r,t_0), k_{eff} \leftarrow$ EVPSolver($XS(t_0)$)
  $\tilde{\varphi}_g(r,t_0) \leftarrow \frac{P_0}{\int_V dr\tilde{q}(r,t_0)}\tilde{\varphi}_g(r,t_0)$ Normalize initial flux
  $C_i(r,t_0) \leftarrow \frac{\beta_i}{\lambda_i}Q(r,t_0)$
  $\omega_T(t_0), \omega_{S,g}(r,t_0), \mu_i(r,t_0) \leftarrow 0$
  while $n \le N$ **do**
    $\omega_T^{(0)}(t_n) \leftarrow \omega_T(t_{n-1})$
    $\omega_{S,g}^{(0)}(r,t_n) \leftarrow \omega_{S,g}(r,t_{n-1})$
    $C_i^{(0)}(r,t_n) \leftarrow C_i(r,t_{n-1})$
    KOmegaIteration(⋯)
    $P_n \leftarrow \int_V drq(r,t_n)$
  end while
  return $\{\varphi_g(r,t_n)\}, \{C_i(r,t_n)\}, \{\omega(r,t_n)\}$
end function
function EVPSolver(XS)
  % Any existing neutron transport or diffusion EVP solver
  return $\tilde{\varphi}_g(r), k_{eff}$
end function
function KOmegaIteration( )
  % Input and output variables are omitted
  while $m \le M$ do
    $\tilde{\varphi}_g^{(m)}(r,t_n), k_D^{(m+1)} \leftarrow$ EVPSolver($XS'(t_0)$)
    $\omega_T^{(m+1)}(t_n) \leftarrow$ Update $\omega_T^{(m+1)}(t_n)$ based on Eq. 22
    $\tilde{\varphi}_g^{(m+1)}(r,t_n) \leftarrow \frac{P_{n-1}}{\int_V dr\tilde{q}(r,t_n)}\tilde{\varphi}_g^{(m)}(r,t_n)$
    $\omega_{S,g}(t_n) \leftarrow \frac{1}{h}ln\left(\frac{\varphi_g^{(m+1)}(r,t_n)}{\varphi_g(r,t_{n-1})}\right)$
    $\varphi_g(r,t_n) \leftarrow \hat{\varphi}_g^{(m+1)}(r,t_n)e^{\frac{\omega_T^{(m+1)}(t_n)+\omega_T(t_{n-1})}{2}h}$
    $C_i^{(m+1)}(r,t_n) \leftarrow$ Update $C_i^{(m)}(r,t_n)$ based on Eq. 26
    $\mu_i^{(m+1)}(r,t_n) \leftarrow$ Update $\mu_i^{(m)}(r,t_n)$ based on Eq. 28
    if $|k_D^{(m+1)} - 1| < \varepsilon$ **then**
      break
    end if
    $XS'(tn) \leftarrow$ Update dynamics cross-sections based Eqs. 20 and 21
  end while
end function

Algorithm 1. SCM for time-spatial equations.

where XS denotes all coefficients in Eqs 11, 12, including cross-sections and dynamic parameters, h is the time step size, and t_N is the last time point. If not otherwise specified, n and m denote the time step index and iteration index, respectively. In this study, the finite difference method (FDM) is applied to solve the EVP of the neutron diffusion equation, and the solutions are $\tilde{\varphi}_g(r,t)$ and the associated eigenvalue k_{D}. Besides, other spatial discretization algorithms are also applicable, such as the nodal method (Abo et al., 2008) and the finite element method.

The amplitude-frequency is updated using the secant method (Chao and Attard, 1985):

$$\omega_T^{(m+1)}(t_n) = \omega_T^{(m)}(t_n) + \left[\omega_T^{(m-1)}(t_n) - \omega_T^{(m)}(t_n)\right]\frac{1-k_{\mathrm{D}}^{(m)}}{k_{\mathrm{D}}^{(m-1)} - k_{\mathrm{D}}^{(m)}} \quad (22)$$

The iteration continues until the dynamic eigenvalue converges to 1. According to Eq. 8, normalization is necessary to update the shape-frequency with the normalized neutron flux $\hat{\varphi}_g(r,t_n)$:

$$\hat{\varphi}_g(r,t_n) = \frac{P(t_{n-1})}{\sum_{g=1}^{G}\int_V dr\kappa\Sigma_{f,g}(r,t_n)\tilde{\varphi}_g(r,t_n)}\tilde{\varphi}_g(r,t_n) \quad (23)$$

Such a normalization enforces the total power contributed by the normalized flux to equal the power of the previous time step, which ensures that the shape-frequency is independent of the flux amplitude. Thus, the update formula of the shape-frequency is:

$$\bar{\omega}_{S,g}(r,t_n) = \frac{1}{\Delta t_n}\ln\left[\frac{\hat{\varphi}_g(r,t_n)}{\varphi_g(r,t_{n-1})}\right] \quad (24)$$

where $\Delta t_n = t_n - t_{n-1}$, and $\bar{\omega}_{S,g}(r,t_n)$ is the average shape-frequency in $[t_{n-1}, t_n]$. According to Eq. 3, the update formula for the actual neutron flux is:

$$\begin{aligned}\varphi_g(r,t_n) &= \varphi_g(r,t_{n-1})e^{\frac{\omega_T(t_n)+\omega_T(t_{n-1})}{2}\Delta t_n + \bar{\omega}_{S,g}(r,t_n)\Delta t_n} \\ &= \hat{\varphi}_g(r,t_n)e^{\frac{\omega_T(t_n)+\omega_T(t_{n-1})}{2}\Delta t_n}\end{aligned} \quad (25)$$

When the actual neutron flux is solved, the precursor concentration is calculated by:

$$C_i(r,t_n) = C_i(r,t_{n-1})e^{-\lambda_i\Delta t_n} + \beta_i e^{-\lambda_i\Delta t_n}\int_{t_{n-1}}^{t_n}\Delta Q(r,t)e^{\lambda_i t}dt \quad (26)$$

Suppose that the fission source changes linearly within $[t_{n-1}, t_n]$, the expression of the fission source is given by:

$$Q(r,t) = Q(r,t_{n-1}) + \frac{Q(r,t_n) - Q(r,t_{n-1})}{\Delta t_n}(t - t_{n-1}) \quad (27)$$

According to Eq. 12, the precursor frequency is calculated by:

$$\mu_i(r,t) = \begin{cases}\beta_i\dfrac{Q(r,t)}{C_i(r,t)} - \lambda_i & C_i(r,t) \neq 0 \\ 0 & C_i(r,t) = 0\end{cases} \quad (28)$$

The resulting flux and precursor frequencies are then used to update the dynamic cross-section. For the case of the PKE, the non-linear frequency-transformed equations can be solved using Algorithm 2:

function PKESolverSCM(t_N, h, IC, DP)
  $\{t_n\} \leftarrow \{0{:}\ h{:}\ t_N\}$ Time steps
  $n(t_0), C_i(t_0), \omega(t_0) \leftarrow IC$ Initial conditions
  while $n \le N$ **do**
    $\omega^{(0)}(t_n) \leftarrow \omega(t_n - 1)$
    $C_i^{(0)}(t_n) \leftarrow C_i(t_{n-1})$
    while $m \le M$ **do**
      $n^{(m+1)}(t_n) \leftarrow n(t_n)e^{\frac{\omega^{(m)}(tn)+\omega(t_{n-1})}{2}h}$
      $\omega^{(m+1)}(tn) \leftarrow$ Update $\omega^{(m)}(tn)$ based on Eq. 31
      $C(m+1)(tn) \leftarrow$ Update $C_i^{(m)}(t_n)$ based on Eq. 33
      if $|f| < \varepsilon$ then
        break
      end if
    end while
  end while
  return $\{n(tn)\}, \{Ci(tn)\}, \{\omega(tn)\}$
end function

Algorithm 2. SCM for PKE.

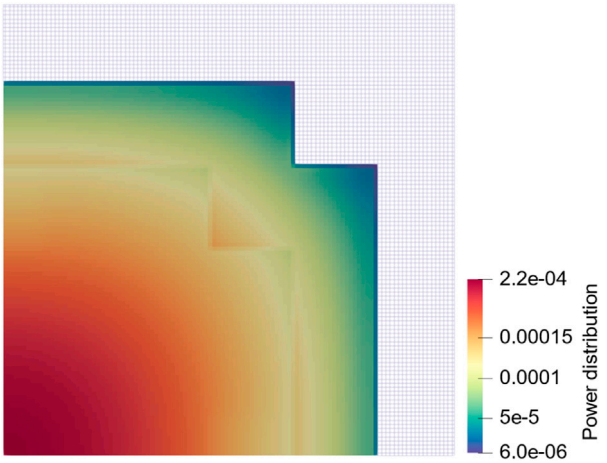

FIGURE 7
Initial power distribution for modified 2D-LMW (Only non-zero power displayed).

where IC represents the initial conditions, and DP is the dynamic parameters in Eqs 16, 18. With a linear approximation of the neutron density frequency, the neutron density can be updated by:

$$n(t_n) = n(t_{n-1})e^{\frac{\omega(t_n)+\omega(t_{n-1})}{2}h} \tag{29}$$

in which ω is solved iteratively. According to Eq. 18, function f is introduced:

$$f(t_n) = -\omega(t_n) + \frac{\rho(t_n)-\beta}{\Lambda} + \frac{1}{n(t_{n-1})e^{\frac{\omega(t_n)+\omega(t_{n-1})}{2}h}} \sum_{i=1}^{I} \lambda_i C_i(t_n) \tag{30}$$

In this case, ω is solved with the secant method until $f(t_n) = 0$:

$$\omega^{(m)}(t_n) = \omega^{(m-1)}(t_n) - \frac{\left[\omega^{(m-1)}(t_n) - \omega^{(m-2)}(t_n)\right]}{\left[f^{(m-1)}(t_n) - f^{(m-2)}(t_n)\right]} f^{(m-1)}(t_n) \tag{31}$$

The precursor concentration $C_i(t_n)$ can be determined using the analytical solution to Eq. 16:

$$C_i(t_n) = C_i(t_{n-1})e^{-\lambda_i \Delta t_n} + \frac{\beta_i}{\Lambda} e^{-\lambda_i \Delta t_n} \int_{t_{n-1}}^{t_n} n(t) e^{\lambda_i t} dt \tag{32}$$

Assuming that the neutron density changes linearly with time, Eq. 32 is transformed into:

$$C_i(t_{n+1}) = C_i(t_n)e^{-\lambda_i \Delta t_n} + \frac{\beta}{\Lambda \lambda_i^2 \Delta t_n} \left\{ \begin{array}{l} [-\Delta t_n \lambda_i n(t_n) - n(t_n) + n(t_{n+1})]e^{-\lambda_i \Delta t_n} \\ +\Delta t_n \lambda_i n(t_{n+1}) + n(t_n) - n(t_{n+1}) \end{array} \right\} \tag{33}$$

4 Error analysis and error estimator

In the SCM, polynomial functions are used to interpolate the exact frequency. The interpolation points are frequencies obtained by solving frequency-transformed equations at different time points. Figure 1 offers a schematic view of errors in the numerical integral. As shown in the figure, when a polynomial interpolation function ω' replaces the original function ω^*, error is introduced when integrating over the time interval. The error can be estimated by the difference in the covered area between the original function and the interpolation function in Figure 1.

With the foregoing considerations, we define the integral error contributing to truncation error in the SCM:

Definition 1. (Integral error): The integral error ε_{int} is defined as the difference between the integral using the original function ω^* and the integral using the polynomial interpolation ω' of ω^*:

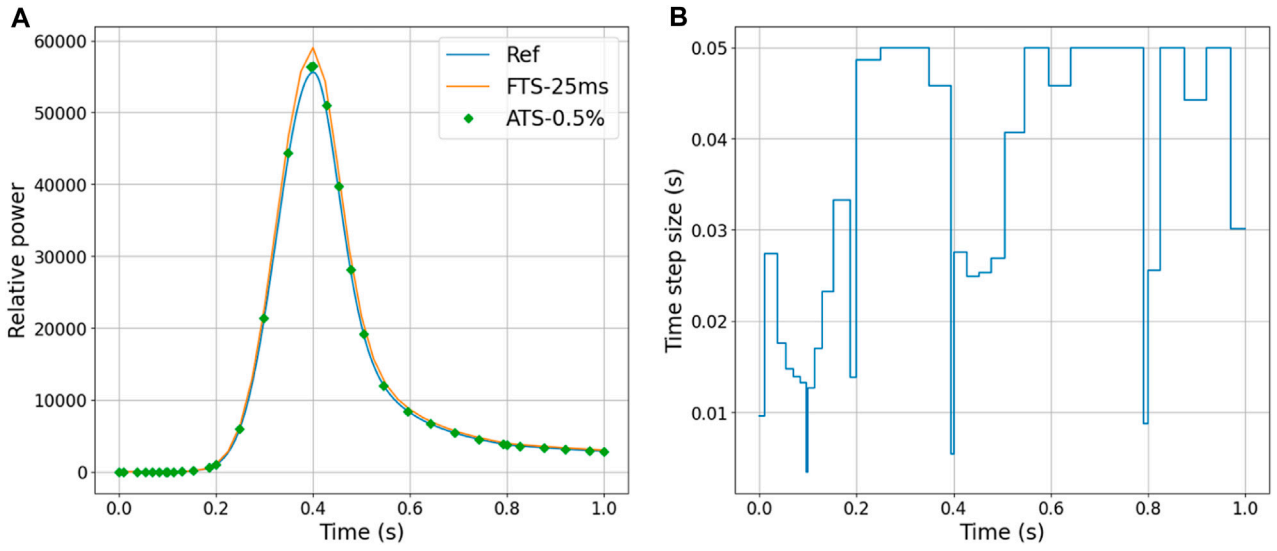

FIGURE 8
Comparison of ATS-0.5% and FTS-25 ms in modified 2D-LMW **(A)** Core power **(B)** Time-variant adaptive time-step size.

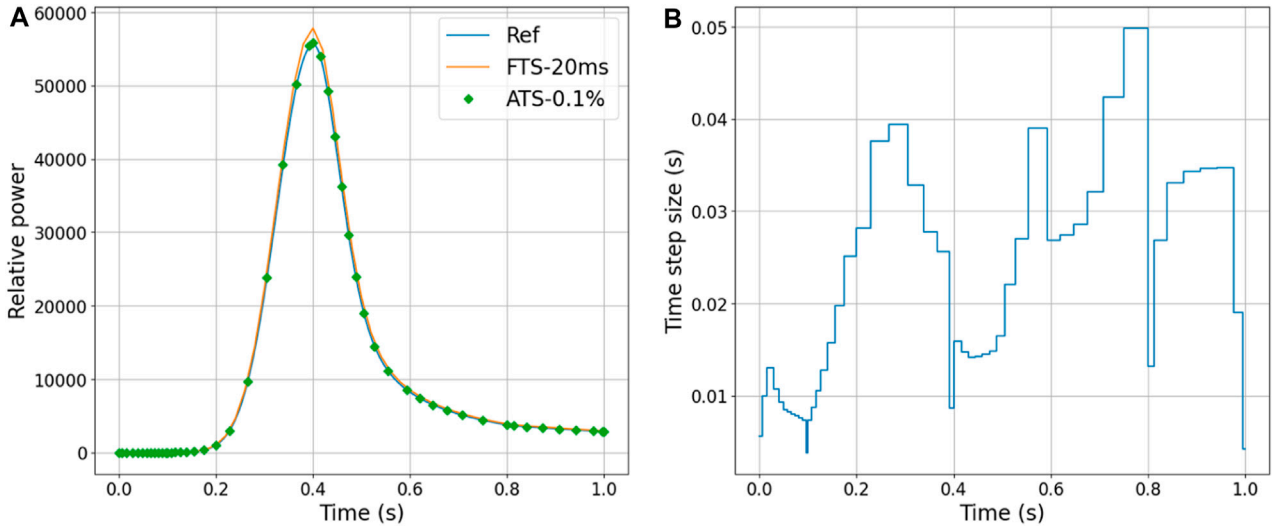

FIGURE 9
Comparison of the ATS-0.1% and FTS-20 ms in modified 2D-LMW **(A)** Core power **(B)** Time-variant adaptive time-step size.

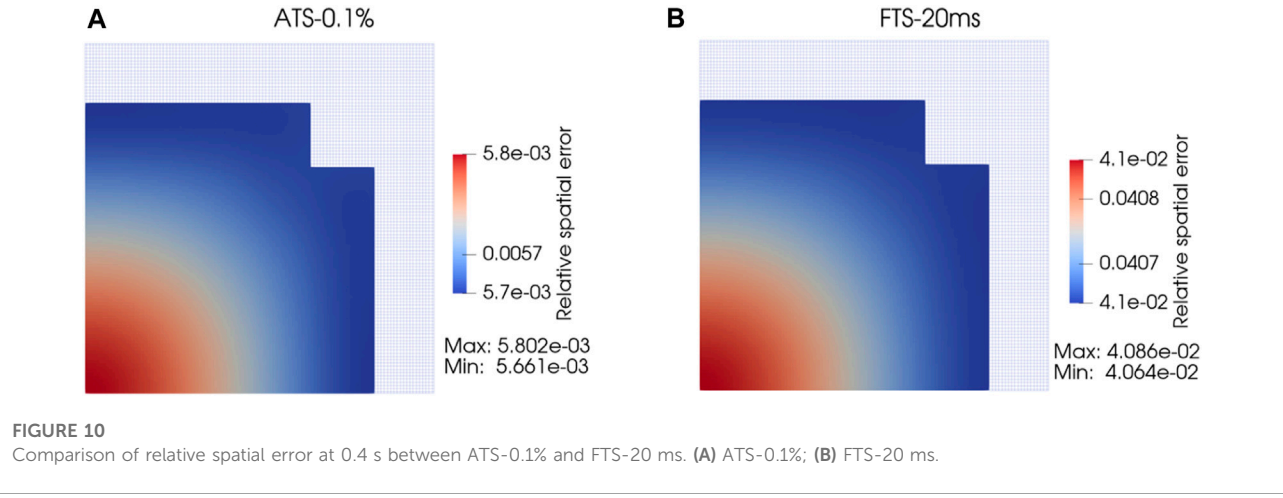

FIGURE 10
Comparison of relative spatial error at 0.4 s between ATS-0.1% and FTS-20 ms. **(A)** ATS-0.1%; **(B)** FTS-20 ms.

$$\varepsilon_{int} \equiv \int_{\Delta t} \omega^{\star}(t)dt - \int_{\Delta t} \omega'(t)dt \tag{34}$$

The interpolation error at t is $\omega^{\star}(t) - \omega'(t)$, which is the difference between the original function providing the interpolation points and the interpolating polynomial. Assume that $\omega'(t)$ is the (degree n or less) interpolating polynomial fitting the $n+1$ points $\{(t_0, \omega^{\star}(t_0)), \cdots, (t_n, \omega^{\star}(t_n))\}$. Without loss of generality, it is assumed that the frequency is $(n+1)$-order differentiable due to the smooth variation of reactivity ρ in Eq. 18. The interpolation error is then (Zhang et al., 2017):

$$\omega^{\star}(t) - \omega'(t) = \frac{1}{(n+1)!}\omega^{\star(n+1)}(\xi_t)\prod_{i=0}^{n}(t - t_i) \tag{35}$$

where ξ_t lies between $[t_0, t_n]$, and $\omega^{\star(n+1)}$ is the $(n+1)$-order derivative of the frequency. When the neutron density frequency is interpolated with a linear function within a time step, by applying the mean value theorem of integrals, the integral error in $[t_{n-1}, t_n]$ is obtained as:

$$\varepsilon_{int} = \int_{t_{n-1}}^{t_n} \omega^{\star}(t)dt - \int_{t_{n-1}}^{t_n} \omega'(t)dt = -\frac{1}{12}\omega^{\star(2)}(\xi)(t_n - t_{n-1})^3 \tag{36}$$
$$\xi \in [t_{n-1}, t_n]$$

Because the exact value of ξ is uncertain, Eq. 36 can be used to provide error bounds to the solutions. Hence, the optimal time step is given using the error estimator:

$$h' = \sqrt[3]{\frac{12\varepsilon_{tol}}{\omega^{(2)}(\xi)}}\, h' \in \mathbb{R}^{+} \tag{37}$$

5 Coupling of the NTSE solver with the PKE predictor

Based on the above discussions, one can select the optimal time steps for the SCM based on Algorithms 3, 4:

TABLE 4 Comparison of global amplitude error between ATS and FTS in modified 2D-LMW.

Time	Reference power	FTS-25 ms error	ATS-0.5% error	FTS-20 ms error	ATS-0.1% error
0.1 s	11.07	3.430%	0.620%	2.182%	0.101%
0.2 s	997.86	5.812%	1.690%	3.651%	0.359%
0.4 s	55591.11	6.017%	1.628%	3.992%	0.570%
0.8 s	3776.35	5.258%	0.402%	3.470%	0.257%
1.0 s	2871.30	5.161%	0.564%	3.295%	0.348%
Number of steps	1,000	40	33	50	50

function SpatialSolverAdaptiveSCM(t_N, h_{max}, h_{min}, ε_{tol}, XS)
  $t_0 \leftarrow 0$
  $\tilde{\varphi_g}(r, t_0), k_{eff} \leftarrow$ EVPSolver($XS(t_0)$)
  $\varphi_g(r, t_0) \leftarrow \frac{P_0}{\int_V dr\tilde{q}(r,t_0)} \tilde{\varphi}_g(r, t_0)$ Normalize initial flux
  $C_i(r, t_0) \leftarrow \frac{\beta_i}{\lambda_i} Q(r, t_0)$
  $\omega_T(t0), \omega_{S,g}(r, t0), \mu i(r, t0) \leftarrow 0$
  for $t_n \le t_N$ **do** n++
    $t'_n \leftarrow t_{n-1} + h_{max}$
    $ICn\text{-}1, DPn\text{-}1 \leftarrow$ Initialized IC and DP
    $h_n \leftarrow$ TimeStepSelection($\varepsilon_{tol}, h_{max}, h_{min}, IC_{n-1}, DP_{n-1}$)
    $tn \leftarrow tn\text{-}1 + hn$
    $\omega_T^{(0)}(t_n) \leftarrow \omega_T(t_{n-1})$
    $\omega_{S,g}^{(0)}(r, t_n) \leftarrow \omega_{S,g}(r, t_{n-1})$
    $C_i^{(0)}(r, t_n) \leftarrow C_i(r, t_{n-1})$
    KOMegaIteration(…)
    $P_n \leftarrow \int_V drq(r, t_n)$
  end for
  return $\{\phi g\ \{r, tn\}, \{Ci(r, tn)\}, \{\omega(r, tn)\}$
end function

Algorithm 3. Adaptive time stepping SCM based on PKE predictor.

function TimeStepSelection(ε_{tol}, h_{max}, h_{min}, IC, DP)
  $h_{sub} \leftarrow \frac{h_{max}}{K}$ Sub − step for PKE
  $\{\omega_k\} \leftarrow$ PKESolverSCM(h_{max}, h_{sub}, IC, DP)
  $\{\omega_k^{[2]}\} \leftarrow \frac{\omega_{k-1} 2\omega_k + \omega_{k+1}}{h^2_{sub}}$
  $\omega^{[2]} \leftarrow max\{|\omega_k^{[2]}|\}$
  $h' \leftarrow$ ErrorControl($\varepsilon_{tol}, \omega^{[2]}$)
  $h \leftarrow \max\{hmin, \min\{h', hmax\}\}$
  return h
end function
function ErrorControl(ε_{tol}, $\omega^{[2]}$)
  $h \sqrt[3]{\frac{12\varepsilon_{tol}}{\omega^{[2]}}}$
  return h'
end function

Algorithm 4. Adaptive time step selection based on PKE.

Algorithm 4 shows the time-step selection subroutine with the error estimator demonstrated in Eq. 37. An optimal time-step size is chosen using the error estimator to control the local error within the pre-set error tolerance in this subroutine. Algorithm 3 is obtained by embedding Algorithm 4 in Algorithm 1.

The key to the ATS algorithm is to solve the PKE in the time interval $[t_{n-1}, t_{n-1} + h_{max}]$ before solving the NTSE. In solving the PKE, dynamic parameters and initial conditions are generated using the solution of the NTSE and the initial adjoint flux. To make appropriate predictions, the following assumptions are used for solving the PKE:

- The flux shape is fixed in the prior interval and the solution $\varphi_g(r, t_{n-1})$ is used to evaluate all necessary parameters and initial conditions;
- All dynamic parameters are constant in the prior interval except the reactivity $\rho(t)$;
- $\rho(t)$ is a linear function in the prior interval, while $\rho(t_{n-1})$ and $\rho(t_{n-1} + h_{max})$ are evaluated based on Supplementary Eq. 4.

The SCM is also applied to solve the PKE because it directly provides the frequency for error estimation.

6 Numerical Results

In this section, the efficiency and accuracy of the ATS algorithm are tested by comparing it with FTS calculation. We examine the time stepping algorithm in the TWIGL problem and the modified LMW-2D problem. The TWIGL problem represents a transient case with step perturbation, and the LMW problem involves smooth reactivity insertions with more complicated geometrical layout that that of the TWIGL problem. The combination of the two problems can be utilized to examine the performance of the ATS algorithm under different transient scenarios.

The cross-sections and dynamic parameters of the two problems are presented in Appendix-B. The reference solutions are obtained by fine time-step size calculations. We apply the ATS algorithm to the problems with specified error tolerance. For comparisons, we also adopt FTS calculations with the same number of time steps as those used to produce the ATS results to evaluate the accuracy improvements of using the ATS algorithm. For ease of description, in these discussions we denote ATS with a tolerance of x% as ATS-x %, and denote FTS with the time step of y ms as FTS-y ms.

6.1 TWIGL

The TWIGL benchmark problem is one quarter of a 2D reactor core consisting of three kinds of fuel materials, as shown in Figure 2 (Kennedy and Riley, 2012).

The TWIGL benchmark problem includes two different reactivity insertion cases: a simple ramp case and a composite case. The composite case is adopted to validate the ATS algorithm, which is shown in Table 1. The transient durations for both cases are 0.5 s. Note that perturbations occur only in Material 1; the other materials remain in their initial conditions. The initial power distribution is shown in Figure 3.

In the TWIGL problem, the pre-set maximum and minimum time-step sizes are chosen as 50 and 1 ms, respectively. Figure 4A compares ATS-0.5% and FTS-20 ms. Figure 4B presents the adaptive time-step size. As shown in Figure 4A, the power curve of ATS agrees well with the reference solution, especially shortly after the step points. Step perturbations are introduced at 0.2 and 0.4 s. Accordingly, the ATS algorithm automatically refines step sizes shortly after step points but otherwise uses a coarse step size. Such a time step adjustment intelligently allocates time steps and spends more computational resources on the interval with rapid changes.

Figure 5A, B compare the results of ATS-0.1% and FTS-12.5 ms. The performance is similar to that in the previous error tolerance setting.

Table 2 summarizes the global core power error at different time step points. The ATS algorithm significantly improves the computational accuracy when the same number of time steps is used, with an error reduction of up to 40~50%. The comparison of ATS-0.5% and FTS-12.5 ms in Table 2 indicates that the ATS algorithm can reduce by 37% the computational time of the FTS calculation with similar numerical accuracy. More significant gains can be expected for large-scale neutron transport problems.

6.2 Modified 2D-LMW

The second 2D benchmark problem is modified from the 3D LMW (Langenbuch, Maurer, Werner) problem without thermal-hydraulic feedback to match the simulation code used in the paper (Kennedy and Riley, 2012). The reactor core is a simplified pressurized water reactor containing two kinds of fuel assemblies and two groups of control rods (CR), as shown in Figure 6.

The modified 2D LMW problem is designed to model a reactivity insertion case, as shown in Table 3. This case is the superposition of ramp perturbations in different regions, by perturbing materials in CR 1 and CR 2. The initial power distribution is illustrated in Figure 7.

For the modified 2D-LMW, Figures 8, 9 present a comparison between the results of the ATS and FTS calculations. In Figure 8A, the error of the power peak, which appears at 0.4 s, is 905.02 and 3344.92 for ATS and FTS, respectively. In Figure 9A, the error of the power peak is 316.87 and 2219.20. The noticeable improvement in the accuracy of the power peak demonstrates the efficacy of the ATS algorithm.

Figure 10 presents the relative spatial error at 0.4s for ATS and FTS. We can observe that the error distribution is fairly uniform in both cases, but the accuracy of ATS is higher than that of FTS.

Table 4 summarizes the global core power error. The results show that the numerical accuracy can be improved significantly. For example, comparing FTS-20 ms and ATS-0.1%, the error reduction can be up to 90%~95%. The core power increases by more than four orders of magnitudes from the initial power, whose variation is much wider than that in the TWIGL problem, but there is no step perturbation. Thus, continuity of frequency enables superior error control.

7 Conclusion

In this work, we perform theoretical and numerical error analyses of the SCM. By expressing the error term using integral error, the mathematical expression of the amplitude error is derived. The integral error is caused by using a polynomial function to represent the exact solution. Based on theoretical analysis, we develop an efficient and easy-to-implement ATS algorithm based on the PKE predictor. A fine-step PKE solver is used to rapidly evaluate high order derivatives of the neutron flux amplitude in the theoretical error expression. The high-order derivatives are used to determine the optimal time-step size for the NTSE. A time step error-estimator for this ATS algorithm is derived, and numerical validation indicates that the ATS is satisfactory in performance and easy to implement.

The accuracy and computational cost of ATS and FTS are examined using benchmark problems. Comparisons show that the ATS algorithm can achieve higher accuracy with the same number of time steps, and significantly improve computational efficiency in the NTSE. It was found that the ATS algorithm significantly improves the computational accuracy when the same number of time steps is used, with error reductions of up to 40%~50% for the TWIGL benchmark and up to 90%~95% for the LMW benchmark. With similar numerical accuracy, the ATS algorithm can reduce by 37% the computational time of the FTS calculation. Future work is to apply the adaptive SCM to 3D neutron transport problems.

Data availability statement

The raw data supporting the conclusion of this article will be made available by the authors, without undue reservation.

Author contributions

DW: Conceptualization, Formal analysis, Methodology, Validation, Writing–original draft. WX: Formal analysis, Methodology, Resources, Writing–review and editing. ZD: Supervision, Writing–review and editing. TZ: Conceptualization, Funding acquisition, Methodology, Writing–review and editing.

Funding

The authors declare that no financial support was received for the research, authorship, and/or publication of this article.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fenrg.2023.1284230/full#supplementary-material

References

Aboanber, A. E., and Hamada, Y. M. (2008). Generalized Runge–Kutta method for two-and three-dimensional space–time diffusion equations with a variable time step. *Ann. Nucl. Energy* 35 (6), 1024–1040. doi:10.1016/j.anucene.2007.10.008

Ban, Y., Endo, T., and Yamamoto, A. (2012). A unified approach for numerical calculation of space-dependent kinetic equation. *J. Nucl. Sci. Technol.* 49 (5), 496–515. doi:10.1080/00223131.2012.677126

Boffie, J., and Pounders, J. M. (2018). An adaptive time step control scheme for the transient diffusion equation. *Ann. Nucl. Energy* 116, 280–289. doi:10.1016/j.anucene.2018.02.044

Caron, D., Dulla, S., and Ravetto, P. (2017). Adaptive time step selection in the quasi-static methods of nuclear reactor dynamics. *Ann. Nucl. Energy* 105, 266–281. doi:10.1016/j.anucene.2017.03.009

Chao, Y. A., and Attard, A. (1985). A resolution of the stiffness problem of reactor kinetics. *Nucl. Sci. Eng.* 90 (1), 40–46. doi:10.13182/nse85-a17429

Kennedy, D., and Riley, D. (2012). *Romes Desert Frontiers*. Routledge.

Park, B. W., and Joo, H. G. (2015). Improved stiffness confinement method within the coarse mesh finite difference framework for efficient spatial kinetics calculation. *Ann. Nucl. Energy* 76, 200–208. doi:10.1016/j.anucene.2014.09.029

Tang, C., Bi, G., and Yang, B. (2019). Application of stiffness confinement method in transient neutron transportation calculation. *Atomic Energy Sci. Technol.* 53 (7), 1202. doi:10.7538/yzk.2018.youxian.0836

Wanner, G., and Hairer, E. (1996). *Solving ordinary differential equations II*, 375. New York: Springer Berlin Heidelberg.

Zhang, T., Wang, Y., Lewis, E. E., Smith, M. A., Yang, W. S., and Wu, H. (2017). A three-dimensional variational nodal method for pin-resolved neutron transport analysis of pressurized water reactors. *Nucl. Sci. Eng.* 188 (2), 160–174. doi:10.1080/00295639.2017.1350002

Zimin, V. G., and Ninokata, H. (1998). Nodal neutron kinetics model based on nonlinear iteration procedure for LWR analysis. *Ann. Nucl. Energy* 25 (8), 507–528. doi:10.1016/s0306-4549(97)00078-9