

A Comparison of Coupling Algorithms for N/TH Transient Problems in HTR

ZHANG Han^{*a,b}, GUO jiong^a, LU jianan^a, NIU jinlin^a, LI fu^a

a Institute of Nuclear and New Energy Technology, Collaborative Innovation Center of Advanced Nuclear Energy Technology, Key Laboratory of Advanced Reactor Engineering and Safety of Ministry of Education, Tsinghua University, Beijing 100084, China

b Institute of Nuclear and Energy Technologies, Karlsruhe Institute of Technology P.O. Box 3640, 76021 Karlsruhe, Germany

**zhanghan06151510@126.com*

Abstract - *This paper evaluates the performance of neutronic/thermal-hydraulic coupling algorithms based on a practical HTR simulator TINTE. We compare the computational performance of OSSI method, Picard method and JFNK method over a range of transients. Numerical results indicate that the fully implicit coupling methods, Picard and JFNK, are more accurate and more stable which are suitable for the mild transient where longer time step could be utilized and higher computational performance could be achieved, while the time step of OSSI is always limited due to the accuracy and stability issue in this situation. However, the OSSI is recommended for the sharp transient where small time step must be utilized to resolve the dynamic process. In this situation, the advantage of its cheap computational cost per time step is dominant. Furthermore, the computational performance of fully implicit schemes, including linear preconditioning JFNK, nonlinear preconditioning JFNK and Picard, are assessed based on the simplified model and the last two methods are also compared on the practical HTR-PM model. The results show that the linear preconditioning JFNK is the best choice from the viewpoint of the computational efficiency, while extensive modifications are required to the existing physics codes. Nonlinear preconditioning JFNK is also attractive due to its “black box” coupling advantage. However, its computational performance is the worst in these three fully implicit schemes, because of its high computational cost at each Krylov iteration. The performance of Picard is worse than linear preconditioning JFNK due to its linear convergence rate. In short, we suggest that linear preconditioning JFNK should be a top option in the new developmental codes, while Picard method is suitable for coupling the existing codes. For an ideal practical code, both semi-implicit and fully implicit schemes should be developed to deal with different transient processes.*

Keywords: *Jacobian-free Newton Krylov method, Picard method, Semi-implicit operator-splitting method, Accuracy, Stability*

I. INTRODUCTION

In order to perform a complete transient calculation, the neutronics and thermal-hydraulics (N/TH) fields should be coupled with a stable, accurate and efficient method. Generally speaking, there are three algorithms to solve the N/TH coupling system, including Semi-Implicit Operator-Splitting (OSSI) method, Picard method and Jacobian-free Newton-Krylov (JFNK) method. Each coupling algorithm has its own advantages and disadvantages. OSSI method is a kind of well-developed and optimized coupling method and has been widely used in industry application, such as the PWR simulator PARCS/TRACE and PARCS/RELAP [1] and the HTR simulator TINTE [2-5]. For this method, the large coupled problems are divided into several smaller problems at first, and then each individual physical field is sequentially solved. It should be noted that there is no iteration between neutronics and thermal-hydraulics during the time step. As a result, some parameters are not converged

during the time step in OSSI method. Furthermore, accuracy and stability issues are caused due to the additional errors derived from the lagged parameters [6]. As a result, time step is always limited because of the lagged parameters, resulting in larger total CPU time cost. Picard method, also known as simple fixed point iteration method, is the extension and improvement of OSSI method [7]. The basic idea of Picard method is adding an outer iteration between neutronics and thermal-hydraulics to make all parameters converged in a fully implicit manner. Due to its simplicity and convenience, some efforts have been made to apply this method to N/TH coupled safety analysis code, such as the practical HTR simulator PARCS/AGREE [8]. Although this approach offers a convenient framework to couple different physics codes with the minimal code modifications, it also has several shortcomings. The main drawback is that the convergence rate of Picard method is only 1st order [9]. It means a number of outer iterations are required to achieve convergence. Additionally, the relaxation factors are always

required to achieve robust convergence which highly depends on the user experience. The third coupling approach is JFNK method, which is another fully implicit coupling scheme. Different from the former two coupling methods, in JFNK method, all the coupled equations are simultaneously solved in a tightly nonlinear coupled form. In theory, JFNK method is featured as its high convergence rate and robust convergence property without relaxation factors [9-10]. Due to these inherent advantages, JFNK method has been attracted wide attentions in the nuclear engineering field, including transient coupling problem [11-17] and steady-state k-eigenvalue coupling problem [7,18-22], and even pure thermal-hydraulic problem [23-25]. It should be noted that only transient coupling problem is investigated in this paper. Recently, many studies about solving the transient problem by JFNK method have been published. However, to date, most of the efforts focus on the simplified 1-dimensional transient problem [11-16], and only a few efforts have been made for the complex multidimensional transient application [6,17]. So the main disadvantage of JFNK method is that its computational performance has not been fully validated, especially for the complex practical engineering physical model. Furthermore, to the best of our knowledge, little efforts have been made for the comprehensive comparison among these three coupling algorithms. The question which algorithm is more recommended in a given situation has not been answered yet.

In this paper, we develop the fully implicit coupling schemes, Picard method and JFNK method, based on HTR simulator TINTE which has been widely used in the HTR transient behavior analysis for safety. And then we investigate the performance of Picard and JFNK solvers compared to OSSI on realistic complex coupling problems for modeling of modular pebble bed high-temperature gas-cooled reactors (HTRs). In order to make a relatively comprehensive comparison, the computational performance is presented and analyzed under different transient processes. And finally we give the recommended coupling algorithms under different situations.

This paper is organized as follows. The outline of the N/TH coupling methods is described in section 2. And the accuracy and stability of semi-implicit operator-splitting scheme and fully implicit schemes (Picard and JFNK) are also analyzed in theory. The numerical results are presented and discussed in section 3. In this section, both simplified model and complex HTR-PM model are utilized. And different transient processes are carried out to access the computational performance of these three coupling algorithms based on the HTR-PM model. The conclusions are presented in section 4.

II. COUPLING ALGORITHMS

For notational simplicity, in the remainder of this paper we will allow ϕ to refer to the neutron distribution. And we

will define T as not only the solution of the solid media thermal diffusion equation, but also the solution to the fluid media equations. So the N/TH coupling system could be expressed as following.

$$\begin{cases} f_{\phi}(x_{\phi}, x_T) = 0 \\ f_T(x_{\phi}, x_T) = 0 \end{cases}$$

$$\text{Where } x = [x_{\phi}, x_T]^T$$

In this section, the outlines of OSSI, Picard and JFNK algorithm are proposed as following.

1. Semi-Implicit Operator-Splitting

In the early development of the N/TH coupling code, the semi-implicit coupling algorithm is widely employed due to the limited computational capability. For example, the HTR simulator TINTE employs this scheme to couple the neutronics and thermal-hydraulics. Under many years of academic and industrial efforts, OSSI method has been fully developed and optimized, and extensively used in nuclear safety analysis. The outline of this coupling scheme is depicted in Table I. In this scheme, the coupled system is divided into several physical fields and each individual physical field is solved sequentially. In order to reduce the computational cost, there is no iteration between neutronics and thermal-hydraulics during the time step. As a result, some parameters are not converged during the time step. Ref.[13] is referred OSSI as to inconsistent coupling method.

Table 1. OSSI Coupling Scheme

Algorithm 1 OSSI coupling scheme.
Given x_{ϕ}^0, x_T^0
Solve $f_{\phi}(x_{\phi}^{new}, x_T^0) = 0$ for x_{ϕ}^{new}
Solve $f_T(x_{\phi}^{new}, x_T^{new}) = 0$ for x_T^{new}
The above calculation process can be rewritten as
$x^{new} = G(x^0)$
Prepare for the next time step calculation
Set $x_{\phi}^0 = x_{\phi}^{new}, x_T^0 = x_T^{new}$

2. Picard Method

With the enhancement of computational capability, fully implicit coupling methods have been proposed and developed. In order to eliminate the inconsistent terms in OSSI scheme, a straightforward path is Picard method, which is also known as simple fixed point iteration method. The basic idea is to add an outer iteration between neutronics and thermal-hydraulics to make all parameters

converged during the time step. The outline of this coupling scheme is presented in Table II. The lagged term x_T^m in the equation $f_\phi(x_\phi^{m+1}, x_T^m) = 0$ could be updated by the outer iteration. And when the outer iteration converges, all the physical quantities are consistent in time. However, it could be proved that the convergence rate of the outer iteration is only 1st order [9]. It always means a number of iterations are required to achieve convergence. Furthermore, in order to make the iteration process robust, the user-defined relaxation factors ω are always needed. And the determination of the relaxation factor ω highly depends on the user experience and several attempts are always needed.

Table 2. Picard Coupling Scheme

Algorithm 2 Picard coupling scheme.
Given x_ϕ^0, x_T^0
For $m = 0, 1, \dots$ until converged do
Solve $f_\phi(x_\phi^{m+1}, x_T^m) = 0$ for x_ϕ^{m+1}
Solve $f_T(x_\phi^{m+1}, x_T^{m+1}) = 0$ for x_T^{m+1}
$x^{m+1} = \omega \cdot x^{m+1} + (1 - \omega) \cdot x^m$
The above process can be rewritten as
$x^{m+1} = G(x^m)$
End for
Prepare for the next time step calculation
Set $x_\phi^0 = x_\phi^{m+1}, x_T^0 = x_T^{m+1}$

3. Jacobian-Free Newton-Krylov Method

The JFNK method significantly differs from the former two methods. Unlike decoupling process in OSSI and Picard method, all the coupled equations in JFNK method are solved in a tightly nonlinear coupled form and all unknowns are updated simultaneously. The outline of JFNK coupling scheme is shown in Table III. The basic idea of JFNK method is that Newton method is employed to solve the nonlinear coupling equation. And the Krylov subspace iteration method is utilized to solve the linear equation derived from the Newton linearization. An important feature of JFNK is that a finite difference approximation is utilized instead of the analytical Jacobian matrix, because only matrix-vector production, not the Jacobian matrix itself, is used in Krylov iteration process. Compared with the linear convergence rate in Picard method, JFNK has super-linear convergence rate. It means fewer iterations are required for JFNK to reach convergence. Another advantage is that no user-defined relaxation factors are required to make the iteration converge.

Table 3. JFNK Coupling Scheme

Algorithm 3 JFNK coupling scheme.
Given $x^0 = [x_\phi^0, x_T^0]^T$
For $n = 0, 1, \dots$ until converged do (Newton iteration)
Solve $J(x^n) \delta x^n = -F(x^n)$ for δx^n
by Krylov subspace iteration method
where the Jacobian matrix
$J(x^n) = \begin{bmatrix} \partial f_\phi / \partial \phi & \partial f_\phi / \partial T \\ \partial f_T / \partial \phi & \partial f_T / \partial T \end{bmatrix}$
For $m = 0, 1, \dots$ until converged do
(Krylov subspace iteration)
At each Krylov iteration, the matrix-vector production is approximated
$\partial f / \partial x \cdot v^m \approx \left[\begin{array}{c} f(\phi + \varepsilon v_\phi^m, T + \varepsilon v_T^m) \\ -f(\phi, T) \end{array} \right] / \varepsilon$
End for
$x^{n+1} = x^n + \delta x^n$
End for
Prepare for the next time step calculation
Set $x_\phi^0 = x_\phi^n, x_T^0 = x_T^n$

4. Accuracy and Stability Analysis

Both Picard and JFNK are the fully implicit coupling algorithms, because all parameters are converged during the time step. As motivation for fully coupling algorithms, the accuracy and stability of semi-implicit algorithms and fully implicit algorithms are analyzed and compared in this section. For simplicity, we also adopt a simplified point coupled reaction system used in Ref.[6], as shown in Eq.(1). The absorption cross section is a function of temperature to consider the negative temperature effect. The coolant equations are not considered explicitly. It should be noted that the following analysis could be easily extended to the multi-dimensional complete N/TH coupling problem.

$$\begin{cases} \frac{1}{v} \frac{\partial \phi}{\partial t} + \Sigma_a(T) \phi - \nu \Sigma_f \phi = 0 \\ \frac{\partial T}{\partial t} - e_f \Sigma_f \phi + \alpha(T - T_0) = 0 \end{cases} \quad (1)$$

Without loss of generality, we employ first-order backward Euler (BE) temporal discretization, as an example, for the neutronics equation. For the fully implicit coupling

algorithms, the absorption cross section is evaluated at t^{n+1} . And the discrete neutronics equation can be written as Eq.(2). While for the semi-implicit coupling algorithms, the absorption cross section is the lagged parameter which is evaluated at t^n . As a result, the neutronics equation can be expressed as Eq.(3).

$$\frac{1}{\nu} \frac{\phi^{n+1} - \phi^n}{\Delta t} + \Sigma_a(T^{n+1})\phi^{n+1} - \nu\Sigma_f\phi^{n+1} = 0 \quad (2)$$

$$\frac{1}{\nu} \frac{\phi^{n+1} - \phi^n}{\Delta t} + \Sigma_a(T^n)\phi^{n+1} - \nu\Sigma_f\phi^{n+1} = 0 \quad (3)$$

Numerical accuracy analysis is made based on the similar method in Ref.[6]. The numerical error for the fully implicit coupling algorithms is shown as Eq.(4). And only truncation error, which is from the first-order backward Euler temporal discretization, makes a contribution to the numerical error. However, for the semi-implicit coupling algorithms, not only the truncation error but also an additional error make contribution to the total numerical error. This additional error is from the unconverged absorption cross section. Furthermore, when higher order temporal schemes are utilized, such as the third order implicit Runge-Kutta scheme, the numerical error for the fully implicit coupling algorithms could be $O(\Delta t^3)$. While for the semi-implicit coupling algorithms, the numerical error is only $O(\Delta t)$ due to the first order additional error from the lagged cross section. We will show the numerical results about the accuracy issue caused by the additional error in section III.

$$\left[\frac{1}{\nu} \frac{\partial \phi}{\partial t} + \Sigma_a(T)\phi - \nu\Sigma_f\phi \right]^{n+1} = \frac{\Delta t}{2\nu} \frac{\partial^2 \phi}{\partial t^2} \quad (4)$$

$$\left[\frac{1}{\nu} \frac{\partial \phi}{\partial t} + \Sigma_a(T)\phi - \nu\Sigma_f\phi \right]^{n+1} = \underbrace{\frac{\Delta t}{2\nu} \frac{\partial^2 \phi}{\partial t^2}}_{\text{Truncation Error}} + \underbrace{\Delta t \frac{\partial \Sigma_a}{\partial T} \frac{\partial T}{\partial t} \phi}_{\text{Additional Error}} \quad (5)$$

Numerical stability analysis is also made in this section. For the fully implicit algorithms, the semi-discrete scheme could be expressed by Eq.(6). According to the stable requirement [26], all eigenvalues of the transition matrix should be non-positive when the transient process has been long enough. The eigenvalues of the fully implicit schemes are calculated and shown in Eq.(7). As the convective heat transfer coefficient is positive, the λ_1 is smaller than 0. According to the physical experience, a perturbed reactor system will return to the critical state after a long period of time due to the negative temperature effect. Therefore, the λ_2 equals to zero which means the critical state. As a result, the fully implicit algorithms are stable. However, for the semi-implicit algorithms, different from the fully implicit algorithms, there is an additional term in the effective absorption cross section in Eq.(8) and Eq.(9). And the semi-discrete scheme and the transition matrix could be expressed by Eq.(10) and Eq.(11). The corresponding eigenfunction of the transition matrix is Eq.(12). And the maximum eigenvalue is calculated in Eq.(13) which relies on the size of

time step. As a result, when the time step is large enough in Eq.(14), this eigenvalue is larger than 0 (is always larger than 0 due to the Doppler effect), which means the coupled system is unstable. In short, the fully implicit algorithms are unconditionally stable. While the semi-implicit algorithm is only conditionally stable. It means when the time step is large enough, unstable phenomenon will occur for semi-implicit scheme. We will show the numerical results about the stability issue in section III.

$$\frac{\partial}{\partial t} \begin{bmatrix} \phi \\ \tilde{T} \end{bmatrix}^{n+1} = \underbrace{\begin{bmatrix} \nu(\nu\Sigma_f - \Sigma_a) & \\ e_f\Sigma_f & -\alpha \end{bmatrix}}_{\text{transition matrix}} \begin{bmatrix} \phi^{n+1} \\ \tilde{T}^{n+1} \end{bmatrix} \quad (6)$$

Where $\tilde{T} = T - T_0$

$$\lim_{t \rightarrow \infty} \lambda_1 = -\alpha < 0, \quad \lim_{t \rightarrow \infty} \lambda_2 = \nu(\nu\Sigma_f - \Sigma_a) = 0 \quad (7)$$

$$\frac{1}{\nu} \frac{\phi^{n+1} - \phi^n}{\Delta t} + \underbrace{\left(\Sigma_a(T^{n+1}) - \Delta t \frac{\partial \Sigma_a}{\partial T} \right)}_{\text{effective absorption cross section}} \phi^{n+1} - \nu\Sigma_f\phi^{n+1} = 0 \quad (8)$$

$$\Sigma_{a,eff}^{n+1} = \Sigma_a(T^{n+1}) - \Delta t \underbrace{\frac{\partial \Sigma_a}{\partial T} \left(\frac{\partial T}{\partial t} \right)^{n+1}}_{\text{additional term}} \quad (9)$$

$$\begin{bmatrix} 1 & -\frac{\partial \Sigma_a}{\partial T} \phi \Delta t \\ 0 & 1 \end{bmatrix}^{n+1} \cdot \frac{\partial}{\partial t} \begin{bmatrix} \phi \\ \tilde{T} \end{bmatrix}^{n+1} = \begin{bmatrix} \nu(\nu\Sigma_f - \Sigma_a) & 0 \\ e_f\Sigma_f & -\alpha \end{bmatrix}^{n+1} \begin{bmatrix} \phi \\ \tilde{T} \end{bmatrix}^{n+1} \quad (10)$$

$$\frac{\partial}{\partial t} \begin{bmatrix} \phi \\ \tilde{T} \end{bmatrix}^{n+1} = \underbrace{\begin{bmatrix} \nu(\nu\Sigma_f - \Sigma_a) + e_f\Sigma_f\phi \frac{\partial \Sigma_a}{\partial T} \Delta t & -\alpha\phi \frac{\partial \Sigma_a}{\partial T} \Delta t \\ e_f\Sigma_f & -\alpha \end{bmatrix}}_{\text{transition matrix}} \begin{bmatrix} \phi \\ \tilde{T} \end{bmatrix}^{n+1} \quad (11)$$

$$\lambda^2 + \left(\alpha - \nu(\nu\Sigma_f - \Sigma_a) - e_f\Sigma_f\phi \frac{\partial \Sigma_a}{\partial T} \Delta t \right) \lambda \quad (12)$$

$$+ \alpha e_f\Sigma_f\phi \frac{\partial \Sigma_a}{\partial T} \Delta t - \alpha \left(\nu(\nu\Sigma_f - \Sigma_a) + e_f\Sigma_f\phi \frac{\partial \Sigma_a}{\partial T} \Delta t \right) = 0$$

$$\lambda_{\max} = \frac{e_f\Sigma_f\phi \frac{\partial \Sigma_a}{\partial T} \Delta t - \alpha + \nu(\nu\Sigma_f - \Sigma_a) + \sqrt{\Delta}}{2} \quad (13)$$

$$\Delta > \left(\alpha - \nu(\nu\Sigma_f - \Sigma_a) \right) \left(e_f\Sigma_f\phi \frac{\partial \Sigma_a}{\partial T} \right) \quad (14)$$

III. EXPERIMENTAL RESULTS

In this section, the performance of the three different algorithms is compared, first for a simplified two-dimensional model and then for a practical engineering coupled problem using HTR simulator TINTE.

1. Two-Dimensional Simplified Problem

A simplified two-dimensional homogeneous reactor model is adopted in this section. Six nonlinear equations are taken to describe the dynamic process in the nuclear reactor, including neutron diffusion equation, solid heat transfer

equation, and mass, momentum and energy equations for fluid. The coolant flows into the reactor from the bottom side and flow out from the top side. The Cartesian geometry is chosen with 3 meters in height and 2 meters in width. There are 100 meshes in both height direction and width direction. In order to simplify the examination of the numerical behavior of different algorithms, pilot code is utilized.

A. Convergence Rate Analysis

The fully implicit schemes are attractive due to its excellent accuracy and stability, which permits larger time step to pursue high efficiency. However, a nonlinear equation system should be solved at each time step. Therefore, an effective solution is crucial for the fully implicit schemes. The convergence and computational performance of Picard and JFNK are compared in this subsection.

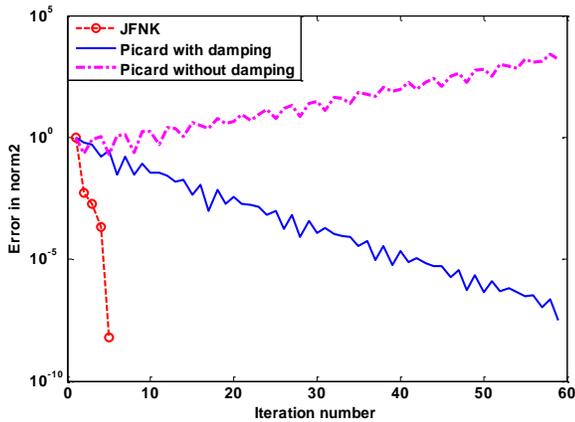


Fig. 1. Convergence process

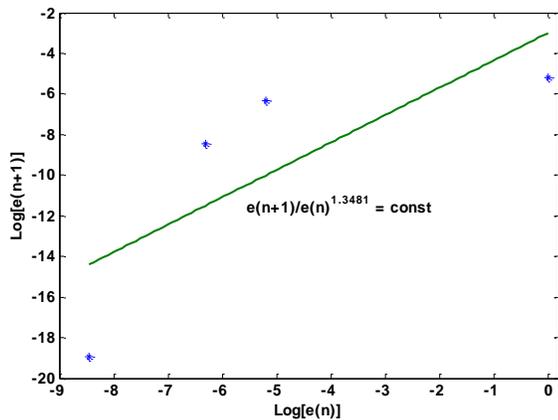


Fig. 2. Convergence rate of JFNK

Firstly, the convergence processes of this two algorithms are represented in Fig.1. For JFNK method, only a few iterations are needed to achieve convergence, due to its super-linear convergence rate, shown in Fig.2. The con-

vergence rate of JFNK in this case is 1.3481 which is consistent with the theory. For Picard method, about 60 iterations are utilized because of its only linear convergence rate. In this case, the convergence rate of Picard is 0.9703, as shown in Fig.3, which agrees well with the theory prediction. What's more, damping parameter is always required to ensure the convergence of the Picard method which is also shown in Fig.1. The Picard iteration will diverge if the damping parameter is not used.

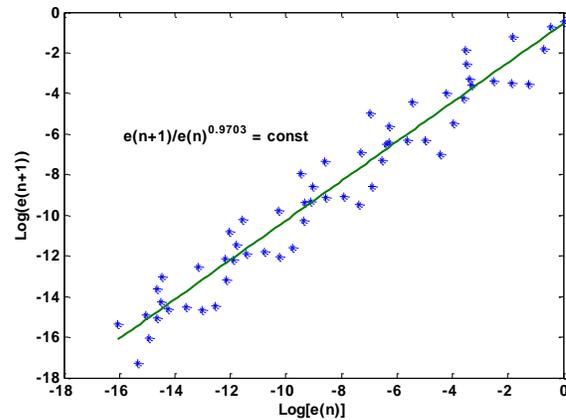


Fig. 3. Convergence rate of Picard

Secondly, the computational performance of these two algorithms is compared. The total computational efficiency not only relies on the iteration number, but also depends on the computational cost per iteration. Preconditioning method is a key issue for the JFNK method. In fact, there are two families of preconditioning method, linear preconditioning and nonlinear preconditioning, which are developed independently. Linear preconditioning JFNK employs the preconditioning technologies which have been widely used in linear equation problems. However, this preconditioning method likely requires extensive modifications to existing physics codes. Nonlinear preconditioning is another attractive choice because it could be implemented as “black box” coupling. Ref. [27] make a comparison between these preconditioning methods from the point of view of computational efficiency and convergence rate. And the main results could be summarized in Table 4. A numerical test is carried out to compare the computational performances among linear/nonlinear preconditioning JFNK and Picard. The results show that the convergence rate of these two preconditioning technologies is similar, because the only difference of the preconditioning system is that there is an additional term for nonlinear preconditioning, as shown in Table 4. However, the computational cost of nonlinear preconditioning is much larger than that of the linear preconditioning, as shown in Table 5. It is because the inverse of preconditioner should be calculated at each Krylov iteration in nonlinear preconditioning while the inverse in the linear one is just updated at each Newton

iteration. As a result, the computational performance of nonlinear preconditioning is even worse than Picard iteration. In short, for the fully implicit schemes, computational performance of linear preconditioning JFNK is the best due to its super-linear convergence rate. Picard method is worse than that of linear preconditioning JFNK because its convergence rate is only linear and many iterations are required to achieve convergence. Lastly, the nonlinear preconditioning JFNK is the worst of all, because of its high computational cost at each Krylov iteration.

Table 5. Computational Performance Comparison

Methods	Newton_It	Krylov_It	CPU time (s)
JFNK_NP*	4.00	17.19	1.8963
JFNK_LP*	4.00	17.34	0.3428
Picard	---	---	0.7521

NP: Nonlinear Preconditioning
LP: Linear Preconditioning

B. Accuracy Analysis

The accuracy and stability of semi-implicit and fully implicit schemes are discussed theoretically in section 2.4. In this subsection, several numerical tests are implemented to analyze the accuracy. And the stability analysis is made in the next subsection.

A cold inlet transient is calculated to compare the accuracy among three different coupling algorithms. The inlet coolant temperature is reduced by 20 K during 0.5s. The neutron flux is represented in Fig.4. The algorithms OSSI, Picard and JFNK are carried out to compare the accuracy. The size of time step is set to be a fixed value 0.01s. The reference solution is computed by JFNK with the fixed time-step size of 0.001s. All results of fully implicit scheme are overlapped, so only one is plotted in Fig.5. For this case, the positive reactivity is inserted due to the cold inlet temperature, so the neutron flux increases at first, and then reduces because of the too large introduced negative reaction. And finally the neutron flux reaches new balanced value. The local neutron profile near the first peak is enlarged as shown in Fig.5. It is clear that the OSSI results are larger than those of fully implicit schemes (JFNK and Picard). The computational performance and accuracy are shown in Table 6. The maximum relative error of OSSI is 1.02%, which is twice larger than those of fully implicit schemes. It is because that there is an additional truncation error term $\Delta t \phi (\partial \Sigma_a / \partial T) (\partial T / \partial t)$ in OSSI compared with the Picard and JFNK, as shown in section 2.4. It should be noted that the computational cost of OSSI per time step is minimal, even smaller than the best fully implicit scheme linear preconditioning JFNK method. It is because in the OSSI the coupled is divided into several subfields by operator-splitting and each individual physical field is only

sequentially solved once per time step. It is a possible situation that the smaller time step is employed for OSSI to ensure the accuracy, but its total computational cost is still better than that of Picard and JFNK due to its cheap cost per time step. In the next subsection, we will make a detailed efficiency comparison under the same accuracy based on the practical physical model.

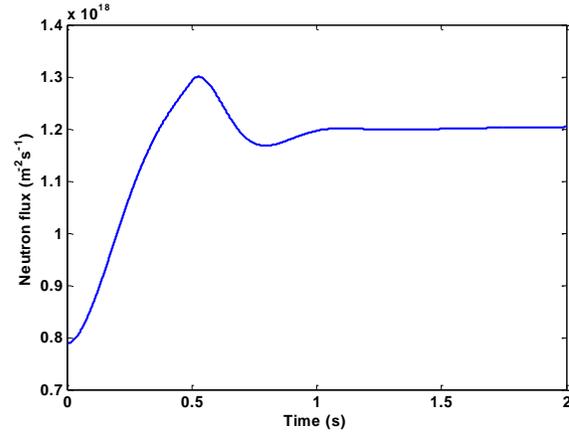


Fig. 4. Neutron flux profile for cold inlet case

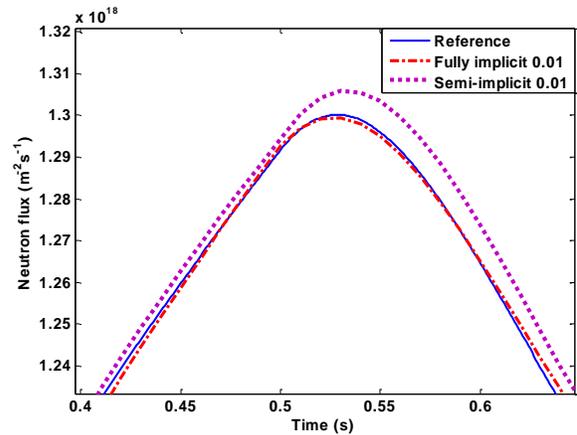


Fig. 5. Local profile for cold inlet case

Table 6. Numerical Accuracy Comparison

Methods	Iteration	Error	CPU time (s)
JFNK_NP	4.1	0.51%	1.9734
JFNK_LP	4.1	0.51%	0.3937
Picard	134.2	0.51%	0.8859
OSSI	---	1.02%	0.2203

C. Stability Analysis

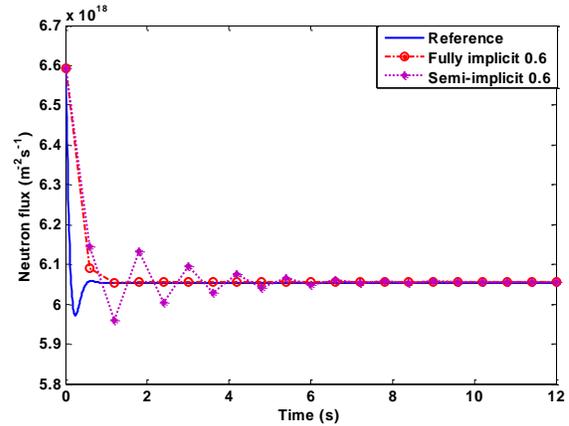
The stability of semi-implicit and fully implicit is verified by the numerical tests in this subsection. Here, the phrase “stable” is used to describe the phenomenon that if an equilibrium system is perturbed, the system will return to

the original state after a long time. This system is called stable system. Otherwise, this system is unstable [26].

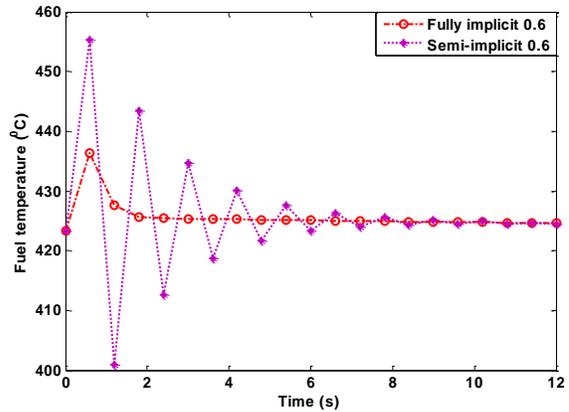
In this subsection, the neutron flux is disturbed by about +10% perturbation which makes the reactor system depart from the equilibrium state. According to the physical experience, the reactor system will return to the original state due to the negative temperature effect. The algorithms OSSI, Picard and JFNK are carried out to evaluate the stability under two different time step sizes. The medium time step size is set to be a fixed value 0.6s and the large time step size is 2.5s. The reference solution is computed by JFNK with the fixed time-step size of 0.01s. It should be noted that we focus on the stability evaluation in this subsection, and the accuracy is not considered here. All results of fully implicit schemes are overlapped, so only one is plotted.

The numerical results at the medium step are presented in Fig.6. The semi-implicit coupling solution exhibits unphysical oscillations, because of the additional term in the effective absorption cross section. It could be interpreted as following. At the first time step (0.6s), the previous temperature is employed to calculate the cross section in the semi-implicit method. The negative feedback doesn't work at the first time step (The semi-implicit neutron flux decreases because the delay neutron precursor is from equilibrium state which couldn't support enough delay neutron). Therefore, the neutron flux of semi-implicit method at 0.6s is higher than that of fully implicit method, as shown in Fig.6a. As a result, the solid fuel temperature at 0.6s is much larger than that of the fully implicit method, as shown in Fig.6b. This too large fuel temperature at 0.6s has two effects on the next time step, as shown in Eq.(9), which consists of two parts, physical component $\Sigma_a(T)$ and the additional component $-\Delta t(\partial \Sigma_a / \partial T)(\partial T / \partial t)$. One effect is that the physical absorption cross section $\Sigma_a(T^{t=1.2s})$ is larger than the fully implicit results which introduce a large negative reactivity. And this reactivity makes the solid temperature decreases at 1.2s. The other effect is that the additional component makes negative reactivity further larger at 1.2s because the additional term is larger than 0 ($(\partial T / \partial t)^{t=1.2} < 0$). Therefore, the neutron flux and solid temperature at 1.2s are much lower than the equilibrium state shown in Fig.6. Again, this too low temperature will introduce large positive reactivity at 1.8s. Furthermore, the additional term could make this positive reactivity much larger because of $(\partial T / \partial t)^{t=1.8} < 0$. As a result, the neutron flux and solid temperature at 1.8s are larger than the equilibrium state shown in Fig.6. This is why the unphysical oscillations occur. We could say that the additional term always makes a "positive feedback" contribution which makes a small quantity smaller, and a large quantity larger. Furthermore, this contribution is proportional to time step. When the time step is not too large, the physical absorption cross section component is still dominant. Even unphysical oscillations occur, the system still returns to the original equilibrium state due to the negative temperature effect. It

means this system is still stable. However, when the time step is further enlarged, the system is far away from the equilibrium state over the time steps, as shown in Fig.7. It means that the semi-implicit scheme is unstable. However, the full implicit schemes are still stable when large time step is utilized.

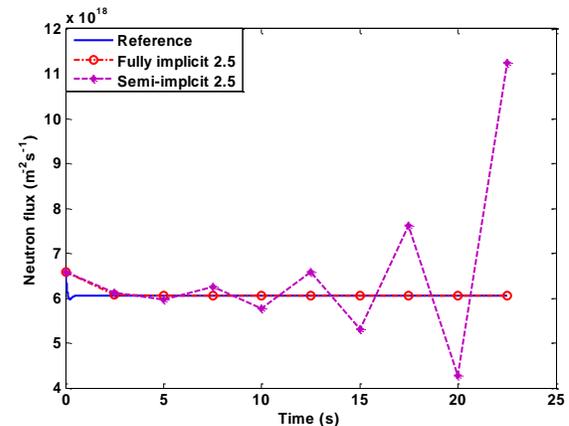


(a) Neutron flux profile

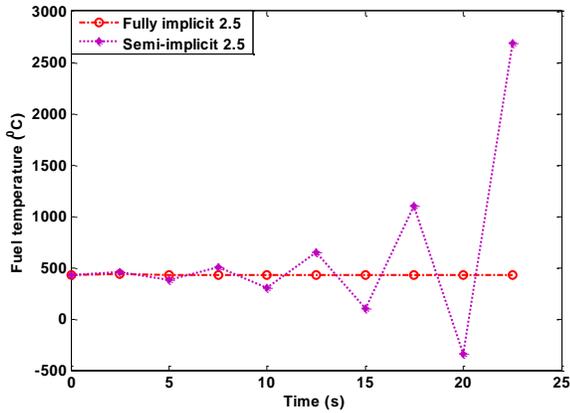


(b) Solid fuel temperature profile

Fig. 6. Neutron flux and fuel temperature at medium step



(a) Neutron flux profile



(b) Solid fuel temperature profile
Fig. 7. Neutron flux and fuel temperature at large step

In short, OSSSI is conditionally stable which is limited by the size of the time step. While the fully implicit method is unconditionally stable. The numerical test results are consistent with the theory prediction in section 2.4. It should be noted that in the practical nuclear safety analysis, the unphysical oscillations should be removed because we couldn't obtain the real physical information. It means we should further refine the time step in the semi-implicit scheme to obtain an accurate result.

2. HTR-PM Simulation by HTR Simulator TINTE

A. Simulator TINTE and Physical Model

TINTE is a well-established code for pebble bed HTR [2-5]. Complicated physical models, including the two energy groups neutronic module, thermal-hydraulic module, chemical reaction module and simplified automatic control module are employed in TINTE to calculate time-dependent neutron fluxes, solid pebble bed temperature and helium temperature in R-Z geometry. Furthermore, the TINTE code had been validated by experimental data from SANA experiment [28] and AVR reactor [29], and is widely used in the transient behavior and safety analysis of HTR, such as HTR-PM [30,31] in China and PBMR in South Africa [32].

However, only OSSSI coupling scheme is utilized in TINTE to couple the neutronic and thermal-hydraulic in which the system's neutronic equations are first solved and then the thermal-hydraulics equations are solved. In this paper, the fully implicit coupling schemes, Picard method (TINTE-Picard) and JFNK method (TINTE-JFNK), are developed based on the original HTR simulator TINTE. It should be noted that only nonlinear preconditioning JFNK is developed at the current stage due to its "black box" coupling advantages. In this paper, we only focus on the multi-physics coupling between neutronic kinetics and thermal-hydraulics. The chemical reaction module in TINTE is not considered here.

A practical High Temperature Reactor-Pebble-bed Modules (HTR-PM) model, rather than a simple model, is utilized to assess the performance of different coupling algorithms in this section. The HTR-PM is a demonstration plant and intends to be a series of commercial plants. The technical research for the HTR-PM began in 2001, and now the power plant is currently progressing well toward connecting to the grid at the end of 2017 [33]. And the cross-section of a reactor module is shown in Fig.8.

According to the structure layout of the HTR-PM reactor, a TINTE calculation model in R-Z geometry has been established based on some reasonable approximations presented in Fig.9a. Not only the reactor, but also the out loop is considered in this model. The out loop component includes a simplified stream generator, a helium circulator and a coaxial gas duct, as shown in Fig.9b. In this paper, only the equilibrium core is taken into account. The initial conditions of these transient processes select the normal operational state with a 100% rated thermal power. The steady-state distributions of neutron fluxes, solid pebble bed temperature and helium temperature are shown in Fig.10, Fig.11 and Fig.12, respectively.

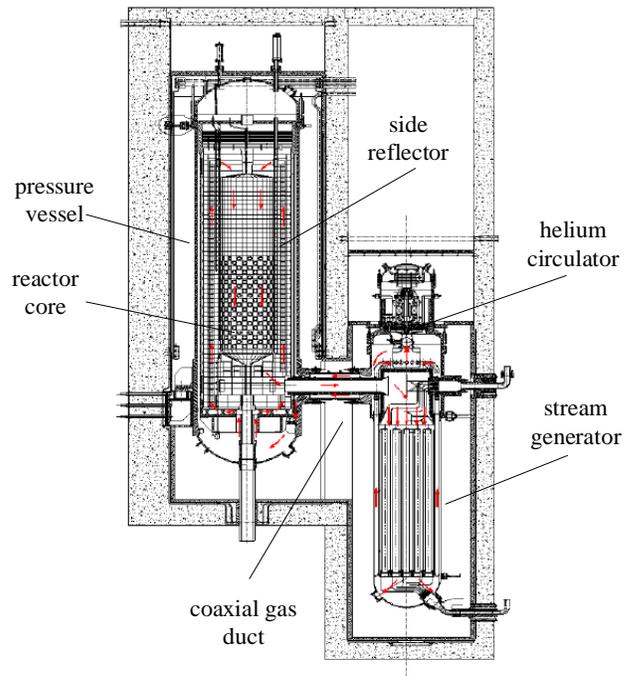
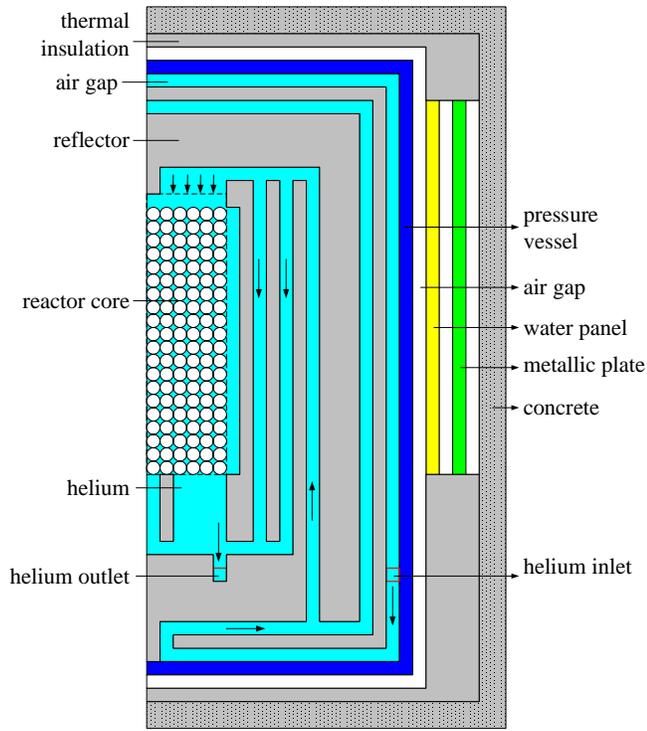
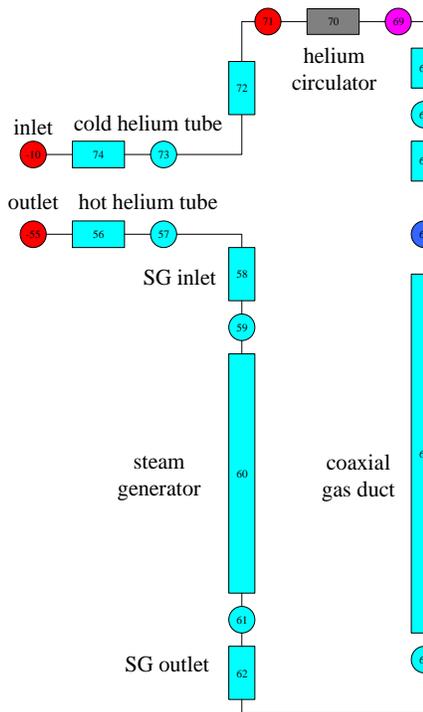


Fig. 8. TINTE calculation model



(a) Reactor model



(b) Outer loop model
Fig. 9. TINTE calculation model

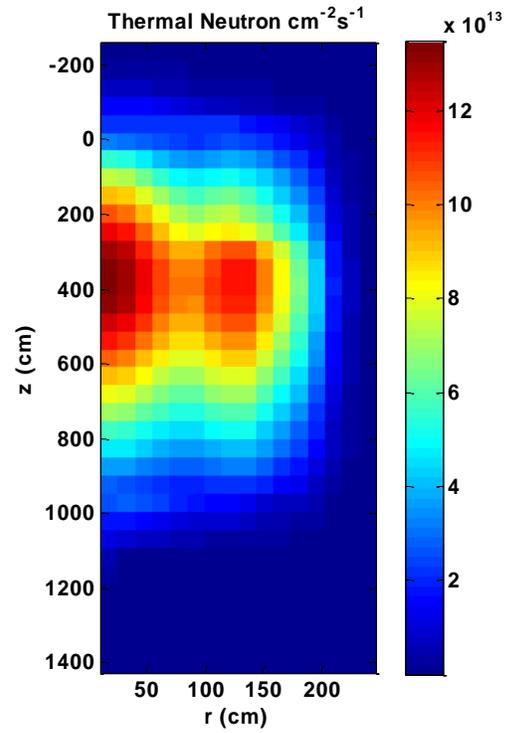


Fig. 10. Thermal neutron distribution

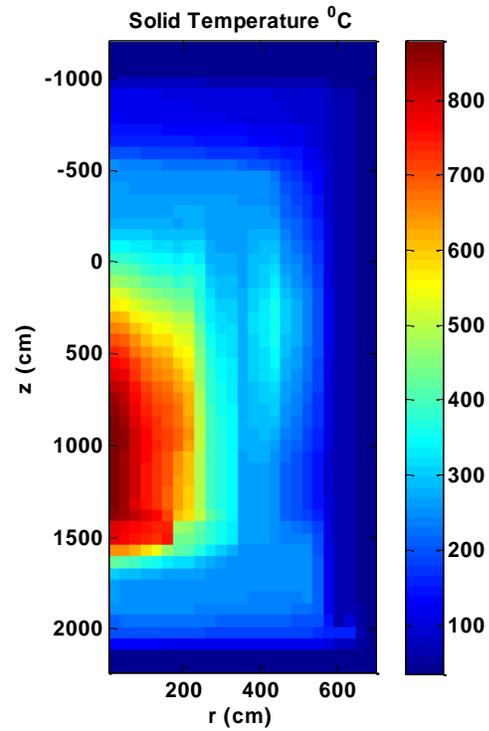


Fig. 11. Solid temperature distribution

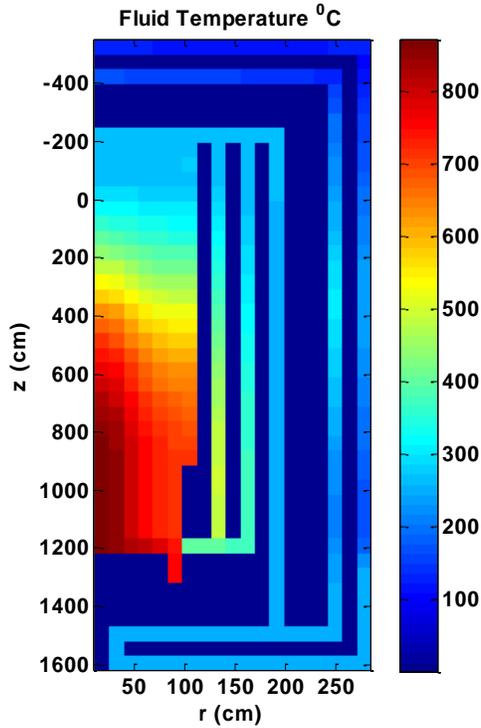


Fig. 12. Helium temperature distribution

B. Code-to-code Verification

In order to verify the computer codes developed in this paper, a code to code comparison is implemented in this subsection. A reduced power transient is utilized to verify the new developed codes TINTE-Picard and TINTE-JFNK. The transient processes are set as following. The total power is linearly decreased from full power to half power during the period between 300s and 600s. In order to keep the total energy balance, the mass flow through helium fan and heat load in the steam generator are also reduced to the half of the original normal value during the same time. The size of time step is set to be a fixed value 1s.

Fig.13 shows the variations of the mean values of the thermal neutron flux, helium temperature and solid pebble bed temperature for this transient. From 0s to 300s, it is a null transient and the physical quantity distributions are the steady-state results. Over the next 300s, the mean value of thermal neutron flux drops from $3.2 \times 10^{13} \text{ m}^{-2}\text{s}^{-1}$ to $1.6 \times 10^{13} \text{ m}^{-2}\text{s}^{-1}$ in a manner specified by the preset operation and then the neutron flux reaches a new balance state. During this transient process, the power generated by the neutron fission and the heat transfer by the thermal-hydraulic are always matched by adjusting the mass flow through helium circulator and the heat load in steam generator. Therefore, the solid pebble bed temperature and helium temperature are almost maintained as a constant throughout the entire transient.

The mean relative errors of neutron flux for TINTE-Picard and TINTE-JFNK are shown in Fig.14. During the period of rapid power change between 300s and 600s, the mean relative error grows up to the maximum. And then the mean relative error decreases as the rate of change in neutron flux decreases. However, the maximum relative error is only about 0.006% which is acceptable in a practical problem.

The numerical experiments show that accurate results can be obtained by the new developed code TINTE-Picard and TINTE-JFNK, which provides the basis for the algorithm comparison.

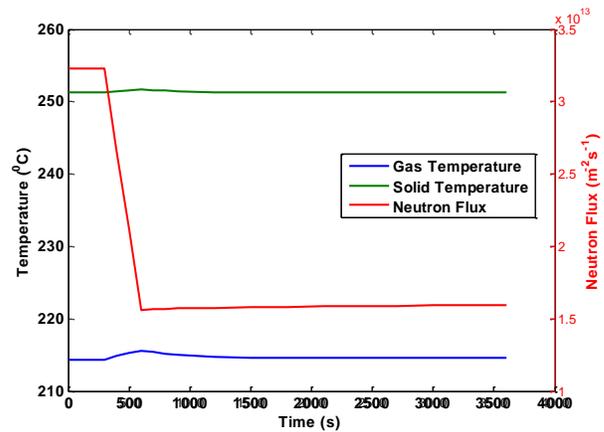


Fig. 13. Variation of the mean value of physical quantities

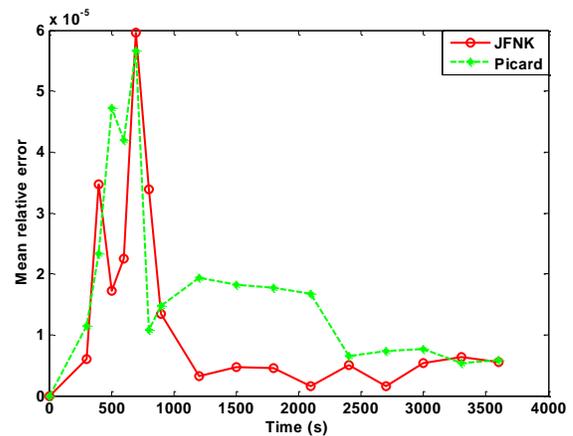


Fig. 14. Mean relative error in reduced power transient case

C. Mild Positive Reactivity Transient

In this subsection, an inserted positive reactivity transient is employed to compare the numerical accuracy between semi-implicit coupling scheme and fully implicit coupling schemes. For this case, it is a null transient in the first 30s. And then 0.05% positive reactivity is linearly introduced within the next 40s. The thermal neutron flux is increased due to the inserted positive reactivity, as shown in Fig.15. And within the next fifty seconds, the power reaches a peak.

In this case, the reactivity is linear introduced over a relatively long period (40s), therefore, there is enough time for thermal-hydraulic to counteract this external reactivity by negative temperature effect. So the peak value of neutron flux is only about 115% of the steady-state value.

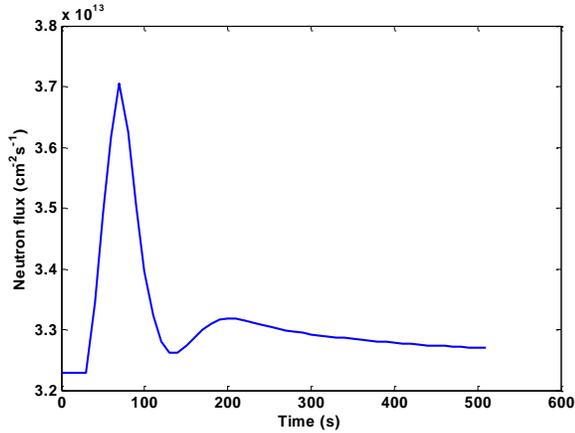


Fig. 15. Neutron flux profile with mild inserted reactivity

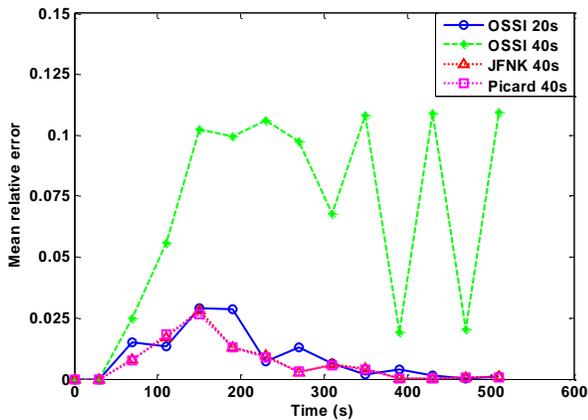


Fig. 16. Mean relative error in mild reactivity case

Table. 7 Accuracy Comparison in Slow Transient

Method	Time step	Max relative error	Total CPU time	Speedup ratio
OSSI (reference)	2s	—	—	—
OSSI	40s	10.9%	—	—
OSSI	20s	2.90%	148.25 s	1
Picard	40s	2.65%	38.52 s	3.85
JFNK	40s	2.82%	100.50 s	1.48

The large time step is utilized to assess the numerical accuracy of three different coupling algorithms. The size of time step is set to be a fixed value 40s. The reference solution is computed by the original TINTE with a very

small time step whose size is 2s. For the fully implicit coupling schemes (TINTE-Picard and TINTE-JFNK), the numerical error is only from the neutron truncation error which relies on the neutron flux change, as shown in Eq.(4). In this case, the dynamic process of neutron flux is mild, therefore, the neutron flux behavior could be resolved well even using large time step. The results show that the maximum values of mean relative error of TINTE-Picard and TINTE-JFNK are 2.65% and 2.82%, respectively, which is acceptable for practical problems. However, for the semi-implicit coupling scheme (original TINTE), the numerical error consists of neutron truncation error, which is from the neutron equation discretization, and additional error derived from the lagged temperature feed-back in the cross sections, as shown in Eq.(5). Furthermore, the size of the additional error is proportional to the size of time step. As a result, this additional error makes a great contribution to the total numerical error because of the large time step. The numerical results show that the maximum value of mean relative error of TINTE is 10.9%, which is more than 3.8 times as much as that of TINTE-Picard and TINTE-JFNK within the same time step, as shown in Fig.16 and Table 7. It means the contribution of this additional error term is almost three times larger than that of the truncation error. In order to obtain the same accuracy of TINTE-Picard and TINTE-JFNK, the smaller time step has to be utilized which is just a half of the original one, as shown in Fig.16. Smaller time step always means more computational cost because of the increased number of time steps. Therefore, the computational efficiency of TINTE-Picard and TINTE-JFNK is 3.85 times and 1.48 times higher than that of TINTE, respectively. In this case, the performance of TINTE-Picard is superior to that of TINTE-JFNK, because the nonlinear preconditioning method is employed in the current TINTE-JFNK. According to the numerical results in section 3.1 by a simplified model, the performance of linear preconditioning TINTE-JFNK has the potential to be superior to that of TINTE-Picard.

D. Mild Periodical Positive Reactivity Transient

In order to evaluate the numerical stability of different coupling algorithms, a periodic reactivity disturbance transient is utilized. This transient is designed by repeating the processes of mild inserted reactivity in subsection 3.2.C. In this case, the periodic change of external reactivity is shown in Fig.17 where the inserted reactivity is periodically changed and the maximum value of the external reactivity is 0.05%. Fig.18 illustrates the neutron flux profile under this periodic varied reactivity. The periodic variation of the neutron flux is consistent with the inserted external reactivity.

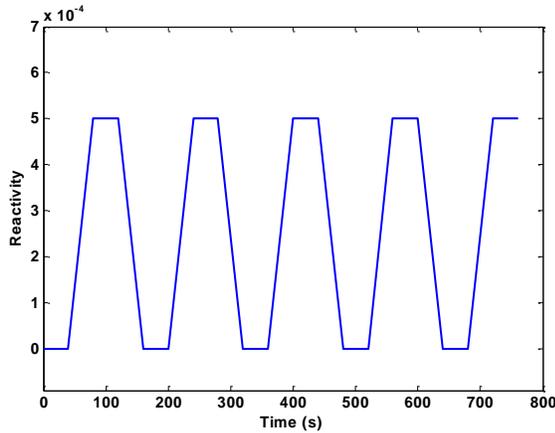


Fig. 17. External reactivity periodical change

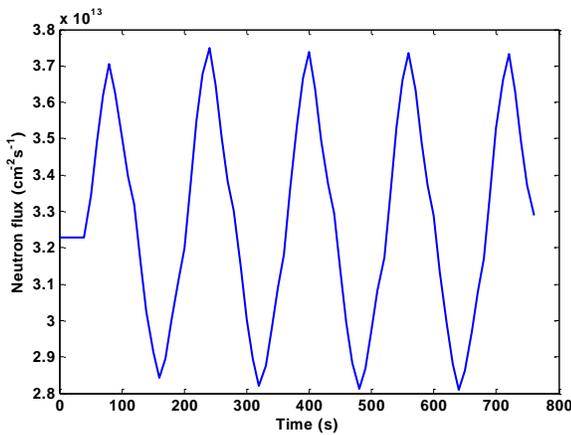


Fig. 18. Neutron flux profile with periodical reactivity

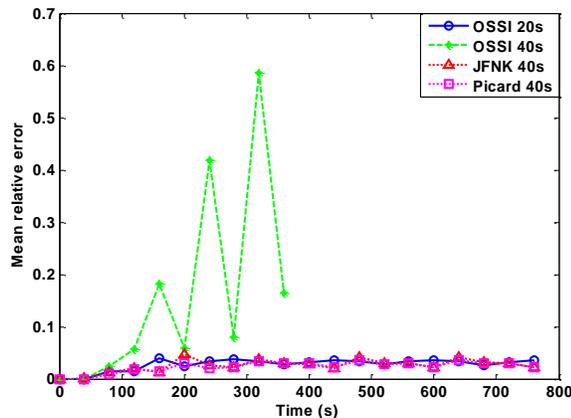


Fig. 19. Mean relative error in reactivity periodical case

In this case, the reference solution is computed by original TINTE with the fixed time-step size of 2s and the large time step (40s) is used to assess the numerical stability of the coupling algorithms. Similar to the previous test, the neutron flux changes gently in this case, therefore, the fullyimplicit coupling schemes (TINTE-Picard and TINTE-JFNK) could employ large time step to achieve an accept-

able result, as shown in Fig.19 and Table 8. The maximum relative errors of TINTE-Picard and TINTE-JFNK are 3.64% and 4.75%, respectively. However, for the semi-implicit coupling scheme, as mentioned in subsection 3.1C, the additional term $-\Delta t(\partial \Sigma_a / \partial T)(\partial T / \partial t)$ always makes a “positive feedback” contribution which makes a small quantity smaller, and a large quantity larger, especially when large time is employed. The same unstable phenomenon occurs in the practical simulator TINTE. The error of TINTE grows up rapidly in the form of oscillations and the maximum relative error of TINTE is 12 larger than those of TINTE-JFNK and TINTE-Picard, as shown in Table 8. Furthermore, the unbounded error leads the physical variables to exceed the reasonable range and cause the simulation to bad stop. Therefore, a smaller time step has to be taken to satisfy the stability requirement for semi-implicit coupling scheme. In this case, the time step size is just a half of the original one. As a result, the computational efficiency of TINTE-Picard and TINTE-JFNK is 3.04 and 1.20 times higher than that of TINTE, respectively. As shown in the simplified case, a similar conclusion could be made here. The OSSI algorithm (TINTE) is only “conditionally stable” and the size of time step is limited due to the stability issue. However, the fully implicit coupling algorithms, such as Picard and JFNK, are “unconditionally stable” and the limitation of time step is cancelled from the perspective of the stability.

Table. 8 Stability Comparison in Slow Transient

Method	Time step	Max relative error	Total CPU time	Speedup ratio
OSSI (reference)	2s	—	—	—
OSSI	40s	58.6%	—	—
OSSI	20s	3.86%	264.25 s	1
Picard	40s	3.64%	87.00 s	3.04
JFNK	40s	4.75%	220.03 s	1.20

E. Sharp Positive Reactivity Transient

In this subsection, a similar inserted positive reactivity transient is calculated. The only difference between this case and case in subsection 3.2.C is that the positive reactivity is linearly inserted within the 1s, not within the 40s. As a result, the dynamic process of thermal neutron flux is much more intense than that of the case in subsection 3.2.C, as shown in Fig.20. And the thermal neutron flux reaches the peak only within 4s, while it is 50s in the mild dynamic case. Furthermore, the peak value of neutron flux is also much higher than that of the mild dynamic case.

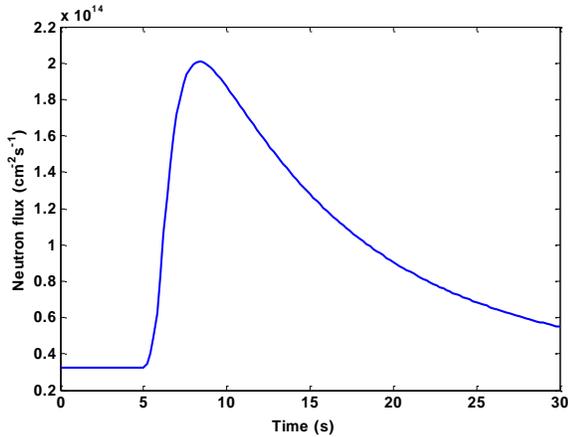


Fig. 20. Neutron flux profile with sharp inserted reactivity

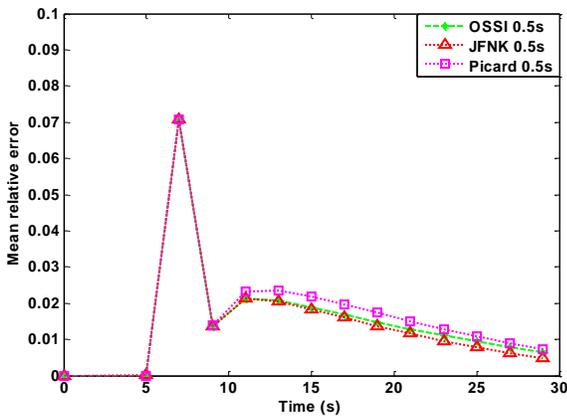


Fig. 21. Mean relative error in sharp reactivity case

Table. 9 Accuracy Comparison in Fast Transient

Method	Time step	Max relative error	Total CPU time
OSSI (reference)	0.01s	—	—
OSSI	0.5s	7.04%	45.56 s
Picard	0.5s	7.07%	56.18 s
JFNK	0.5s	7.09%	155.90 s

Different from the two former cases, only small time step could be utilized to obtain an accurate result in this case. The size of the time step is set to a fixed value 0.5s. And the reference solution is computed by original TINTE with the fixed time-step size of 0.01s. The numerical results are shown in Table 9. In this case, OSSI (TINTE) achieves the same accuracy under the same time step. It is because the effect of the additional error $\Delta t \cdot (\partial \Sigma_a / \partial T) \cdot (\partial T / \partial t) \cdot \phi$ could be ignored when the time step is small. What's more, since the equations in semi-implicit scheme are easier to solve, the total computational cost of OSSI is cheaper than

that of fully implicit schemes under the same time step, as shown in Table 9. Therefore, the OSSI coupling scheme is recommended for the sharp dynamic processes.

IV. CONCLUSIONS

In this study, the comparisons of operator-splitting semi-implicit, Picard iteration and Jacobian-free Newton-Krylov are performed. The accuracy and stability of different coupling schemes are analyzed in theory and verified by the numerical tests. Different from most existing studies that focus on the simplified model under a single transient process, our study considers not only simplified model based on the pilot code, but also the complex engineering model based on the practical HTR reactor simulator TINTE. Furthermore, several transient processes are designed and implemented to determine the suitable range of each coupling scheme.

Numerical results indicate that the fully implicit coupling methods, Picard and JFNK, are more accurate and more stable. And they are suitable for the mild transient where longer time steps could be utilized and higher computational performance could be achieved. In this situation, the time step of OSSI is always limited due to the accuracy and stability issue. While the OSSI is recommended for the sharp transient where small time step must be utilized to resolve the dynamic process. In this situation, the advantage of its cheap computational cost per time step is dominant. Therefore, for an ideal practical code, both semi-implicit and fully implicit schemes should be developed. And the user could choose the coupling method based on different situations.

The computational performances of different fully implicit schemes, including linear preconditioning JFNK, nonlinear preconditioning JFNK and Picard, are assessed based on the simplified model and the latter two methods are also compared on the practical HTR-PM model. The results show that the super-linear convergence rate is achieved in JFNK while Picard has only linear convergence rate. The linear preconditioning JFNK is the best fully implicit scheme from the viewpoint of the computational efficiency. However, it may require extensive modifications to existing physics codes. Nonlinear preconditioning JFNK is another attractive choice because it could be implemented as “black box” coupling. However, its computational performance is the worst in these three fully implicit schemes, because of its high computational cost at each Krylov iteration. The performance of Picard is better than nonlinear preconditioning. In short, linear preconditioning JFNK should be a top option in the new developmental codes. While Picard method is suitable for the existing codes coupling.

ACKNOWLEDGMENTS

This research is supported by Chinese National S&T Major Project (No. ZX06901) and Chinese National Natural Science Foundation Project (No.11375099 and No.115051-02).

REFERENCES

1. D. Barber, R.M. Miller, H. Joo, T. Downar, "Coupled 3D reactor kinetics and thermal-hydraulic code development activities at the U.S. nuclear regulatory commission," In: *Mathematics and Computation, Reactor Physics and Environmental Analysis in Nuclear Applications*, September, Madrid, Spain (1999).
2. H. Gerwin, "Das zweidimensionale Reaktordynamikprogramm TINTE," Teil1: Grundlagen und Lösungsverfahren, Jül-2167. Kernforschungsanlage Jülich GmbH (1987).
3. H. Gerwin, W. Scherer, E. Teuchert, "The TINTE modular code system for computational simulation of transient process in the primary circuit of a pebble-bed high-temperature gas-cooled reactor," *Nuclear Science and Engineering* 103, 302-312 (1989).
4. H. Gerwin, W. Scherer, "The two-dimensional reactor dynamics programme TINTE, part 3: program architecture and user's manual," Kernforschungsanlage Jülich GmbH (2001).
5. H. Gerwin, W. Scherer, A. Lauer, I. Clifford, "TINTE-nuclear calculation theory description report," Jül-4317. Institute for Energy Research, Forschungszentrum Jülich (2010).
6. H. Park, D.A. Knoll, D.R. Gaston, et al, "Tightly coupled multiphysics algorithms for pebble bed reactor," *Nuclear Science and Engineering*, 166, 118-133 (2010).
7. S. Hamilton, M. Berrill, K. Clarno, "An assessment of coupling algorithms for nuclear reactor core physics simulations," *Journal of Computational Physics*, 311, 241-257 (2016).
8. S. Volkan, "High Temperature Gas Cooled Reactor Analysis with PARCS/AGREE," (2009). <https://www.nrc.gov/docs/ML0910/ML091060017.pdf>
9. C.T. Kelley, "Iterative Methods for Linear and Nonlinear Equations," North Carolina State University, SIAM (1995).
10. D. Knoll, D. Keyes, "Jacobian-free Newton-Krylov methods: a survey of approaches and applications," *J. Comput. Phys.* 193, 357-397 (2004).
11. V.A. Mousseau, "A fully implicit, second order in time, simulation of a nuclear reactor core," *Proceeding of ICONE 14*, Florida, USA (2006).
12. M.A. Pope, V.A. Mousseau, "Accuracy and efficiency of a coupled neutronics and thermal hydraulics model," *Nuclear Engineering and Technology*, 41:885-892 (2008).
13. J.C. Ragusa, V.S. Mahadevan, "Consistent and accurate schemes for coupled neutronics thermal-hydraulics reactor analysis," *Nuclear Engineering and Design* 239: 566-579 (2009).
14. J.K. Watson, K.N. Ivanov, "Demonstration of implicit coupling of TRACE/PARCS using simplified one-dimensional problems," *Nuclear Technology*, 180: 174-190 (2012).
15. V.S. Mahadevan, J.C. Ragusa, V.A. Mousseau, "A verification exercise in multiphysics simulations for coupled reactor physics calculations," *Progress in Nuclear Energy*, 55: 12-32 (2012).
16. H. Zhang, F. Li, "Full Implicit Integrate Solution to the Coupled Neutronics/Thermal-Hydraulics Problems," 21st ICONE, Chengdu, China, July 29-August 2 (2013) (CD-ROM).
17. J. Gan, Y. Xu, T.J. Downar, "A matrix-free Newton method for coupled neutronics thermal-hydraulics reactor analysis," *Nuclear Mathematical and Computational Sciences* (2003) (CD-ROM).
18. J.A. Roberts, B. Forget, "Solving Eigenvalue Response Equations with Jacobian-Free Newton-Krylov Methods," *M&C 2011*, Rio de Janeiro, RJ, Brazil, May 8-12 (2011) (CD-ROM).
19. D.F. Gill, "Newton-Krylov Methods for the Solution of the k-Eigenvalue Problem in Multigroup Neutronics Calculations," Penn State University, Ph.D thesis (2009).
20. D.Kastanya, "Implementation of a Newton-Krylov Iterative Method to Address Strong Non-linear Feedback Effects in FORMOSA-B BWR Core Simulator," North Carolina State University, Ph.D thesis (2002).
21. J.A. Willert, "Hybrid Deterministic/Monte Carlo Methods for Solving the Neutron Transport Equation and k-Eigenvalue Problem," North Carolina State University, Ph.D thesis (2013).
22. A.M. Ward, "A newton-krylov solution to the coupled neutronics-porous medium equations," University of Michigan, Ph.D Thesis (2013).
23. C. Frepoli, J.H. Mahaffy, K. Ohkawa, "Notes on the implementation of a fully-implicit numerical scheme for a two-phase three-field flow model," *Nuclear Engineering and Design*, 225, 191-217 (2003).
24. L. Zou, H. Zhao, H. Zhang, "Applications of high-resolution spatial discretization scheme and Jacobian-free Newton-Krylov method in two-phase flow problems," *Annals of Nuclear Energy*, 83, 101-107 (2015).
25. L. Zou, H. Zhao, H. Zhang, "Numerical implementation, verification and validation of two-phase flow four-equation drift flux model with Jacobian-free Newton-Krylov method," *Annals of Nuclear Energy*, 87, 707-719 (2016).
26. M.W. Hirsch, S. Smale and R.L. Devaney, "Differential Equations, Dynamical Systems, and An Introduction to Chaos," Elsevier (2004).
27. H. Zhang, J. Guo, J. Lu, et al, "The comparison between nonlinear and linear preconditioning in JFNK method

- for solving the N/TH coupling problem,” Progress in Nuclear Energy (submitted) (2017).
28. H. Niessen, B. Stocker, “Data sets of SANA experiment: 1994–1996”, JUEL3409, Forschungszentrum Jülich (1997).
 29. W. Scherer, H. Gerwin, “Progress and problems in modelling HTR core dynamics”, IAEA IWGGCR/24 (1990).
 30. Y. Zheng, L. Shi, “Reactivity accident in a high temperature gas-cooled reactor due to inadvertent withdrawal of control rod”. Journal of Engineering for Gas Turbines and Power, 133, 052902(6) (2011).
 31. F. Li, D. Wang, C. Hao and et al, “Solution of multiple circuits of steam cycle HTR system,” Nuclear Engineering and Design, 271, 411–416 (2014).
 32. Ugur Emre Sikik, “TINTE analysis of PBMR reactivity insertion transients,” Proceedings HTR2006: Third International Topical Meeting on High Temperature Reactor Technology, Johannesburg, South Africa, Oct. 1–4 (2006) (CD-ROM).
 33. Z. Zhang, Y. Dong, F. Li, et al, “The Shandong Shidao Bay 200 MWe High-Temperature Gas-Cooled Reactor Pebble-Bed Module (HTR-PM) Demonstration Power Plant: An Engineering and Technological Innovation,” Engineering, 2: 112-118 (2016).

Table 4 Comparison Between Linear and Nonlinear Preconditioning

	Nonlinear preconditioning	Linear preconditioning	relationship
Nonlinear equations	$\tilde{F}(x) = x + M^{-1}(x) \cdot (N(x) \cdot x - q) = 0$	$F(x) = M(x) \cdot x + N(x) \cdot x - q = 0$	same solution
Linear equations	$\tilde{J} \cdot \delta x = -\tilde{F}(x)$	$M^{-1}(x_0) \cdot J \cdot \delta x = -M^{-1}(x_0) \cdot F(x)$	—
Preconditioned matrix of linear equations	$\tilde{J} = I + \frac{\partial}{\partial x} [M^{-1}(x) \cdot (N(x) \cdot x - q)]$ $= I + M^{-1}(x) \cdot \frac{\partial}{\partial x} [N(x) \cdot x]$ $+ \underbrace{\frac{\partial}{\partial x} [M^{-1}(x)] \cdot [N(x) \cdot x] - \frac{\partial}{\partial x} [M^{-1}(x) \cdot q]}_{\text{additional term}}$	$M^{-1}(x_0) \cdot J = M^{-1}(x_0) \cdot \frac{\partial}{\partial x} [M(x) + N(x) \cdot x - q]$ $= I + M^{-1}(x_0) \cdot \frac{\partial}{\partial x} [N(x) \cdot x]$	Additional term
Computational formula	$\tilde{J}_v = \frac{M^{-1}(x + \varepsilon v) F(x + \varepsilon v) - M^{-1}(x) F(x)}{\varepsilon}$	$M^{-1}(x_0) J_v = \frac{M^{-1}(x_0) F(x + \varepsilon v) - M^{-1}(x_0) F(x)}{\varepsilon}$	
Computational cost	Calculate the inverse of $M(x + \varepsilon v)$ at each Krylov step	Calculate the inverse of $M(x_0)$ at each time step or Newton step	