Nuclear Science and Techniques (2026) 37:75
https://doi.org/10.1007/s41365-026-01889-3

DAYU3D: a modern code for HTGR thermal-hydraulic design and accident analysis

Hao-Jie Zhang[1] · Yu-Tong Wen[1] · Ding She[1] · Yan-Hua Zheng[1] · Lei Shi[1]

Received: 25 March 2025 / Revised: 20 April 2025 / Accepted: 30 April 2025 / Published online: 9 February 2026

Abstract
DAYU3D is a modern three-dimensional (3D) computer code for thermal-hydraulic design and accident analysis in high-temperature gas-cooled reactors (HTGRs), developed by the Institute of Nuclear and New Energy Technology (INET) at Tsinghua University. Compared to the traditional codes like TINTE, the DAYU3D code has advantages due to its refined framework, improved models, and more efficient algorithms. It is able to simulate the continuous movement of control rods and is more rigorous in treating radiation heat transfer and the break mass flow. Advanced computational methods significantly improve the computational efficiency of DAYU3D, achieving a time reduction of over 60% compared to TINTE. Extensive verification and validation with more than 100 cases demonstrate that DAYU3D is promising for HTGR 3D thermal-hydraulic design and accident analyses.

Keywords High-temperature gas-cooled reactor · Thermal-hydraulic design and accident analysis code · Three-dimensional · DAYU3D

1 Introduction

Modular pebble-bed high-temperature gas-cooled reactor (HTGR) is one kind of advanced reactors because of its inherent safety. The core of a pebble-bed HTGR is composed of a large number of fuel pebbles, and it utilizes TRISO-coated fuel particles to enhance the reactor's safety. The single-phase helium is usually employed as coolant, and it enables a high core outlet temperature of 750 °C. Due to the distinct thermal-hydraulic and structural features of HTGRs compared to other kind of reactors, specialized computer codes are usually required for thermal-hydraulic design and accident analysis.

In the past few decades, some two-dimensional (2D) codes have been developed around the world. Among these codes, the THERMIX [1, 2] and TINTE [3] codes developed by Forschungszentrum Jülich GmbH (FZJ) in Germany are widely used in the thermal-hydraulic design and safety analyses of multiple pebble-bed HTGRs including AVR [4, 5], PBMR [6], HTR-10 [7, 8] and HTR-PM [9, 10].

With growing demands for higher-fidelity analysis of the three-dimensional (3D) phenomena in HTGRs, various institutions have developed some 3D thermal-hydraulic analysis codes. Some notable developments include the MGT-3D code by Jülich (Germany) [11], ATTICA3D code from the Stuttgart University (Germany) [12, 13], and the PRONGHORN code developed at Idaho National Laboratory (USA) [14].

In China, the Institute of Nuclear and New Energy Technology (INET) at Tsinghua University has conducted sustained research on pebble-bed HTGR. Based on the framework and models of THERMIX, the INET developed the two-dimensional DAYU code using the Fortran language. The DAYU code maintains the main functions of THERMIX while demonstrating enhanced performance in fuel temperature [15] and flow field calculations [16]. However, the framework and algorithms employed in DAYU code exhibit lower scalability and computational efficiency, which are not suitable for addressing complex 3D problems. Therefore, a modern C++ code named DAYU3D is developed and adopts

This work was supported by the CNNC Youth Research Project.

✉ Ding She
sheding@tsinghua.edu.cn

[1] Institute of Nuclear and New Energy Technology, Key Laboratory of Advanced Reactor Engineering and Safety of Ministry of Education, Tsinghua University, Beijing 100084, China

a completely new code framework with good extensibility compared to the previously developed DAYU code. It will also be a good platform for investigating innovative methodologies in HTGR thermal hydraulics.

This paper presents the latest developments regarding DAYU3D. The remainder of this paper is organized as follows. Section 2 presents the mathematical models and code framework implemented in DAYU3D. Section 3 introduces the new features of DAYU3D. In Sect. 4, some numerical results of code verification and validation are presented. The concluding remarks are presented in Sect. 5.

2 Mathematical models and code framework of DAYU3D

2.1 Thermal-hydraulic models

The governing equation for the solid temperature is as follows:

$$\frac{\partial\left(\rho_{\mathrm{s}} c_{\mathrm{p,s}} T_{\mathrm{s}}\right)}{\partial t}=\nabla\cdot\left(\lambda_{\mathrm{s}}\nabla T_{\mathrm{s}}\right)+q_{\mathrm{s}}+q_{\mathrm{gs}}, \tag{1}$$

where ρ, c_{p}, and λ represent the density, specific heat, and thermal conductivity, respectively. t denotes the time, and T is the temperature. Subscripts s and g represent solid and gas, respectively. q_{s} is the heat source of the solid. q_{gs} is the heat transferred between gas and solid, and it is calculated by Eq. (2) as below:

$$q_{\mathrm{gs}}=\frac{1}{V_{\mathrm{mesh}}}A_{\mathrm{gs}}h_{\mathrm{gs}}\left(T_{\mathrm{g}}-T_{\mathrm{s}}\right), \tag{2}$$

where A_{gs} and h_{gs} represent the heat transfer area and convective heat transfer coefficient between the solid and gas, respectively; V_{mesh} is the mesh volume.

DAYU3D code has incorporated various empirical correlations for the thermal conductivity and specific heat of commonly used materials in HTGRs [17], such as the pebble bed, matrix graphite of fuel pebble, reflector, carbon bricks, and concrete. In the pebble-bed region, the Zehner–Schlunder formula [18], the Robold formula [19], and the Barthels formula [20] are employed to calculate the pebble-bed effective thermal conductivity.

Radiation heat transfer is also considered in DAYU3D. In order to simplify the calculation, the anisotropic effective thermal conductivity is utilized to simulate the radiation heat transfer, which is a proven practice in the TINTE program [21]. Moreover, DAYU3D incorporates a more accurate heat radiation model based on the view factor [22].

As for the gas flow and gas temperature calculation, the porous media model is employed. The mass, momentum, and energy conservation equations are as follows:

$$\frac{\partial\left(\varphi\rho_{\mathrm{g}}\right)}{\partial t}+\nabla\cdot\left(\rho_{\mathrm{g}}\vec{u}\right)=S, \tag{3}$$

$$\nabla p+W\rho_{\mathrm{g}}\vec{u}=\rho_{\mathrm{g}}\vec{g}, \tag{4}$$

$$\frac{\partial\left(\varphi\rho_{\mathrm{g}}c_{\mathrm{p,g}}T_{\mathrm{g}}\right)}{\partial t}+\nabla\cdot\left(\vec{u}\rho_{\mathrm{g}}c_{\mathrm{p,g}}T_{\mathrm{g}}\right)=\nabla\cdot\left(\varphi\lambda_{\mathrm{g}}\nabla T_{\mathrm{g}}\right)-q_{\mathrm{gs}}+q_{\mathrm{src}}, \tag{5}$$

where φ is the porosity, **u** represents the superficial velocity, S is the mass source, p denotes the gas pressure, W represents the resistance coefficient, and **g** is the gravity acceleration. The subscript 'g' represents the gas phase. q_{gs} is calculated by Eq. (2). q_{src} denotes the energy carried by the mass source and is calculated by the following equation:

$$q_{\mathrm{src}}=Sc_{\mathrm{p,src}}T_{\mathrm{src}}, \tag{6}$$

where the subscript 'src' represents the mass source.

The properties of helium are calculated by KTA formulas [23]. The flow resistance and convective heat transfer inside the pebble bed are calculated according to KTA standards as well [24]:

$$W=\xi\frac{1-\varphi}{\varphi^3}\frac{\left|\rho\vec{u}\right|}{2d\rho}, \tag{7}$$

$$\xi=\frac{320}{\left(\frac{Re}{1-\varphi}\right)}+\frac{6}{\left(\frac{Re}{1-\varphi}\right)^{0.1}}, \tag{8}$$

$$Nu=1.27\frac{Pr^{1/3}}{\varphi^{1.18}}Re^{0.36}+0.033\frac{Pr^{1/2}}{\varphi^{1.07}}Re^{0.86}, \tag{9}$$

where Re, Nu, and Pr represent the Reynolds number, the Nusselt number, and the Prandtl number, respectively; d is the hydraulic diameter.

In the pebble-bed region, the temperature distribution inside the fuel pebble is calculated. The governing equation for fuel pebble temperature is the one-dimensional (1D) heat conduction equation in spherical coordinate:

$$\frac{\partial\left(\rho_{\mathrm{fe}}c_{\mathrm{p,fe}}T_{\mathrm{fe}}\right)}{\partial t}=\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\lambda_{\mathrm{fe}}\frac{\partial T_{\mathrm{fe}}}{\partial r}\right)+q_{\mathrm{fe}}, \tag{10}$$

where r is the radius, subscript 'fe' represents the fuel element, i.e., the fuel pebble, and q_{fe} denotes the nuclear power density inside the fuel pebble.

Since the online refueling is used in pebble-bed HTGR, there may be different batches of fuel pebbles in one porous media mesh, as shown in Fig. 1. Because different batches of fuel elements have different power, they have different surface temperatures. Therefore, radiation

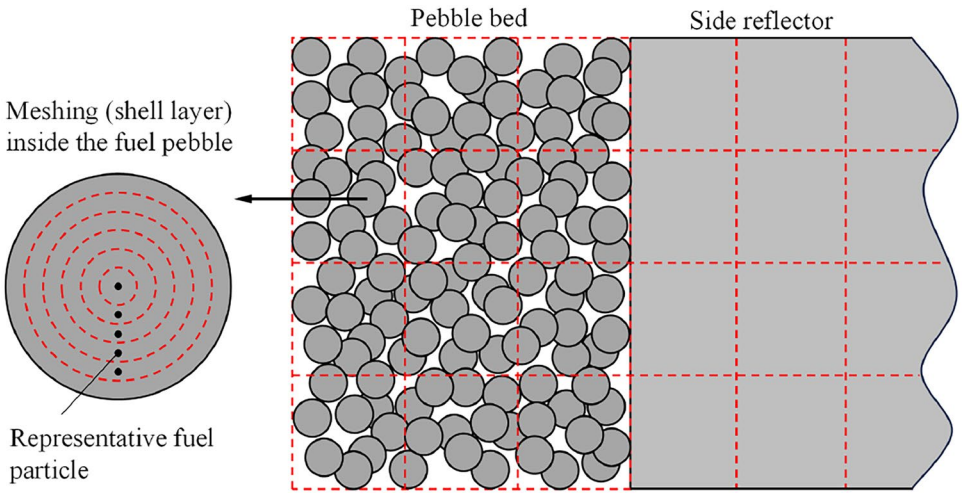

Fig. 1 Meshing in pebble-bed region and side reflector

heat transfer would occur between different batches of fuel spheres within the same mesh. The amount of radiation heat received by a single sphere of jth batch from other batches can be calculated by [21, 25]:

$$Q_{r,j} = 4\pi R^2 \cdot \frac{\varepsilon_j \sigma}{\bar{\varepsilon}} \sum_n \left[p_n \varepsilon_n \left(T_n^4 - T_j^4 \right) \right], \tag{11}$$

where R is the pebble radius; ε and T denote the surface emissivity and surface temperature of fuel pebble, respectively; subscripts n and j represent the n^{th} and j^{th} batch of fuel pebble, respectively; $\bar{\varepsilon}$ is the average surface emissivity; and p_n is the fraction of nth batch.

The calculation of fuel particle temperature is conducted through lumped-heat-capacity method, considering the size of fuel particle is relatively small. Once the temperature of each shell layer in fuel sphere has been determined, the temperature of representative particle in each shell layer (as shown in Fig. 1) can be quickly calculated by the following equation [26]:

$$\left(\rho c_{\text{p}}\right)_{\text{fp}} \frac{\mathrm{d}T_{\text{fp}}}{\mathrm{d}t} = q_{\text{fp}} - \frac{1}{\alpha_{\text{f}}} \left(T_{\text{fp}} - T_{\text{fe}} \right), \tag{12}$$

where subscript 'fp' represents the fuel particle, T_{fe} is the fuel pebble temperature in one shell layer, and α_{f} represents the thermal resistance between the fuel particle and the matrix graphite.

2.2 Neutronics models

The neutronics kinetics calculation for HTGR is implemented in DAYU3D code system, employing a 3D cylindrical (r-z-θ) geometry to model the coupled neutronic/thermal-hydraulic (N/TH) transient behavior.

The predictor–corrector quasi-static method (PCQS) is utilized in solving space-time neutronics kinetics equation, which is written as:

$$\begin{aligned}
&\frac{1}{\nu_g(\vec{r},t)} \frac{\partial \phi_g(\vec{r},t)}{\partial t} - D_g(\vec{r},t)\nabla^2 \phi_g(\vec{r},t) + \Sigma_{r,g}(\vec{r},t)\phi_g(\vec{r},t) = \sum_{g' \neq g}^{G} \Sigma_{s,g' \to g}(\vec{r},t)\phi_{g'}(\vec{r},t) \\
&+ \chi_g^{\text{p}}(\vec{r},t) \sum_{g'=1}^{G} \upsilon\Sigma_{f,g'}(\vec{r},t)\phi_{g'}(\vec{r},t) + \sum_{i=1}^{6} \chi_{i,g}^{\text{d}}(\vec{r},t)\lambda_i(\vec{r},t)C_i(\vec{r},t) \\
&\frac{\partial C_i(\vec{r},t)}{\partial t} = \beta_i(\vec{r},t) \sum_{h=1}^{G} \upsilon\Sigma_{f,g}(\vec{r},t)\phi_g(\vec{r},t) - \lambda_i(\vec{r},t)C_i(\vec{r},t), i = 1,2...6
\end{aligned} \tag{13}$$

where $\phi_g(\vec{r},t)$ is the g^{th} group (similarly hereinafter) neutron flux, $\nu_g(\vec{r},t)$ is averaged velocity, $D_g(\vec{r},t)$ is diffusion coefficient, $\Sigma_{r,g}(\vec{r},t)$ is removal cross section, $\Sigma_{s,g' \to g}$ is scattering cross section from the g'^{th} group to the g^{th} group, $\upsilon\Sigma_{f,g'}(\vec{r},t)$ is fission production cross section, $\chi_g^{\text{p}}(\vec{r},t)$ is prompt fission spectrum, $C_i(\vec{r},t)$ is concentration of the i^{th} group delayed precursor, $\lambda_i(\vec{r},t)$ is decay constant of the i^{th} precursor group, and $\beta_i(\vec{r},t)$ is production fraction of the i^{th} precursor group.

The equation includes six groups of delayed neutron precursor groups. The quasi-static approach decomposes the neutron flux into two components:

$$\begin{aligned}
\phi_g(\vec{r},t) &= N(t) \cdot \tilde{\phi}_g(\vec{r},t) \\
C_i(\vec{r},t) &= T_i(t) \cdot \tilde{C}_i(\vec{r},t)
\end{aligned} \tag{14}$$

where $N(t)$ and $T_i(t)$ are amplitude functions, representing the rapidly varying global population of flux and precursor concentration, respectively. $\tilde{\phi}_g(\vec{r},t)$ and $\tilde{C}_i(\vec{r},t)$ are shape functions describing the spatial distribution of flux and precursor concentration, which changes slowly over time. In order to ensure the uniqueness of the decomposition, the normalization condition is imposed on the shape function by using the adjoint flux:

$$\begin{aligned}&\sum_{g=1}^{G}\int_V \frac{\phi_g^*(\vec{r})\tilde{\phi}_g(\vec{r},t)}{\nu_g(\vec{r},t)}\mathrm{d}\vec{r}=1\\&\sum_{g=1}^{G}\int_V \phi_g^*(\vec{r})\,\chi_{i,g}^{\mathrm{d}}(t)\tilde{C}_i(\vec{r},t)\mathrm{d}\vec{r}=1,\quad i=1,...,6\end{aligned}\tag{15}$$

The adjoint flux is obtained by solving the eigenvalue adjoint diffusion equation.

PCQS consists of predictor and corrector phase, which solve shape and amplitude function, respectively. In predictor phase, the space-time equation is temporally discretized into large steps and is solved by implicit scheme, as shown in Eq. (16).

$$\begin{aligned}&\frac{1}{\nu_g(\vec{r},t_{n+1})}\frac{\phi_g^{\mathrm{pre}}(\vec{r})-\phi_g(\vec{r},t_n)}{\Delta t_n}-D_g(\vec{r},t_{n+1})\nabla^2\phi_g^{\mathrm{pre}}(\vec{r},t_{n+1})\\&+\Sigma_{r,g}(\vec{r},t_{n+1})\phi_g^{\mathrm{pre}}(\vec{r},t_{n+1})=\sum_{g'\neq g}^{G}\Sigma_{s,g'\to g}(\vec{r},t_{n+1})\phi_{g'}^{\mathrm{pre}}(\vec{r},t_{n+1})\\&+\chi_g^{p}(\vec{r},t_{n+1})\sum_{g'=1}^{G}\upsilon\Sigma_{f,g'}(\vec{r},t_{n+1})\phi_{g'}^{\mathrm{pre}}(\vec{r},t_{n+1})\\&+\sum_{i=1}^{6}\chi_{i,g}^{\mathrm{d}}(\vec{r},t_{n+1})\lambda_i(\vec{r},t_{n+1})C_i^{\mathrm{pre}}(\vec{r},t_{n+1})\\&\frac{C_i^{\mathrm{pre}}(\vec{r},t_{n+1})-C_i(\vec{r},t_n)}{\Delta t_n}=\beta_i(\vec{r},t_{n+1})\sum_{h=1}^{G}\upsilon\Sigma_{f,g}(\vec{r},t_{n+1})\phi_g^{\mathrm{pre}}(\vec{r},t_{n+1})\\&-\lambda_i(\vec{r},t_{n+1})C_i^{\mathrm{pre}}(\vec{r},t_{n+1})\end{aligned}\tag{16}$$

Then, by using the normalization condition of Eq. (15), the shape function is obtained as follows:

$$\begin{aligned}&\tilde{\phi}_g(\vec{r},t_{n+1})=\phi_g^{\mathrm{pre}}(\vec{r},t_{n+1})\Big/\sum_{g=1}^{G}\int_V\frac{\phi_g^*(\vec{r})\phi_g^{\mathrm{pre}}(\vec{r},t_{n+1})}{\nu_g(\vec{r},t_{n+1})}\mathrm{d}\vec{r}\\&\tilde{C}_i(\vec{r},t_{n+1})=C_i^{\mathrm{pre}}(\vec{r},t_{n+1})\Big/\\&\sum_{g=1}^{G}\int_V\phi_g^*(\vec{r})\,\chi_{i,g}^{\mathrm{d}}(\vec{r},t_{n+1})C_i^{\mathrm{pre}}(\vec{r},t_{n+1})\mathrm{d}\vec{r},\quad i=1,...,6\end{aligned}\tag{17}$$

In corrector phase, by substituting Eq. (14) and Eq. (15) into Eq. (13), the point-kinetics equation for amplitude function is obtained as:

$$\begin{aligned}&\frac{\mathrm{d}N(t)}{\mathrm{d}t}=\frac{\rho(t)-\tilde{\beta}(t)}{\Lambda(t)}N(t)+\sum_{i=1}^{6}\tilde{\lambda}_i(t)T_i(t),\\&\frac{\mathrm{d}T_i(t)}{\mathrm{d}t}=\frac{\tilde{\beta}_i(t)}{\Lambda(t)}N(t)-\tilde{\lambda}_i(t)T_i(t),\end{aligned}\tag{18}$$

where the point-kinetics parameters are defined by the following equation:

$$\begin{aligned}&\Lambda(t)=\frac{1}{\sum_{g=1}^{G}\int_V\phi_g^*(\vec{r})\chi_g(\vec{r},t)\sum_{h=1}^{G}\upsilon\Sigma_{f,g'}(\vec{r},t)\tilde{\phi}_{g'}(\vec{r},t)\mathrm{d}\vec{r}},\\&\rho(t)\\&=\Lambda(t)\cdot\sum_{g=1}^{G}\int_V\mathrm{d}\vec{r}\left\{\begin{array}{l}-[\Sigma_{r,g}(\vec{r},t)-\Sigma_{r,g}(\vec{r},0)]\tilde{\phi}_g(\vec{r},t)\phi_g^*(\vec{r})+\\\phi_g^*(\vec{r})\{\sum_{g'\neq g}^{G}[\Sigma_{s,g'\to g}(\vec{r},t)-\Sigma_{s,g'\to g}(\vec{r},0)]\\\tilde{\phi}_{g'}(\vec{r},t)+\sum_{g'=1}^{G}[\chi_g(\vec{r},t)\upsilon\Sigma_{f,g'}(\vec{r},t)-\\\frac{\chi_g(\vec{r},0)\upsilon\Sigma_{f,g'}(\vec{r},0)}{k_{\mathrm{eff}}^*}]\tilde{\phi}_{g'}(\vec{r},t)\}\end{array}\right\},\\&\tilde{\beta}_i(t)\\&=\Lambda(t)\cdot\sum_{g=1}^{G}\int_V\mathrm{d}\vec{r}\left(\phi_g^*(\vec{r})\chi_{i,g}^{\mathrm{d}}\sum_{g'=1}^{G}\beta_i\upsilon\Sigma_{f,g'}(\vec{r},t)\tilde{\phi}_{g'}(\vec{r},t)\right),\\&\tilde{\beta}(t)=\sum_{i=1}^{6}\tilde{\beta}_i(t),\\&\tilde{\lambda}_i(t)=\sum_{g=1}^{G}\int_V\mathrm{d}\vec{r}\left\{\phi_g^*(\vec{r})\chi_{i,g}^{\mathrm{d}}(t)\lambda_i(t)\tilde{C}_i(\vec{r},t)\right\}.\end{aligned}\tag{19}$$

In order to precisely capture the rapid varying of the amplitude function, the point-kinetics equation is discretized and solved over the minor time steps, which also requires the simultaneous updating of point-kinetics parameters. Considering the inclusion of the shape function in the expression of point-kinetics parameters as is shown in Eq. (19), it is necessary to interpolate the shape functions obtained over large steps into the nested minor steps. Fig. 2 summarizes the time-step division and solution process of PCQS.

Compared with the traditional implicit method used in TINTE, PCQS enables the use of large time steps for solving high-dimensional shape equation and minor time steps

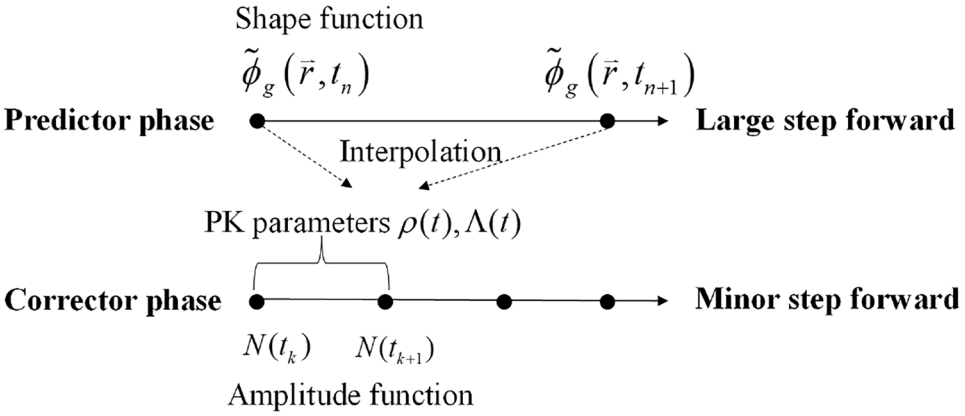

Fig. 2 Schematic of the time-step division and solution process of PCQS

for low-dimensional amplitude equation, which remarkably improves the computation efficiency while maintaining the accuracy.

Besides, the neutronics calculation module of DAYU3D employs two-level meshing system. The coarse-level mesh is used to characterize the spatial distribution of macroscopic cross sections and cover some of mesh grids for solid temperature computation, which are user-defined as neutronics zone. The fine-level mesh is generated by imposing a refined division on the coarse-level mesh and is used to spatially discretize the space-time kinetics equation by finite-volume method.

The multi-group macroscopic cross section, corresponding to each coarse-level grid, is in the representation of four-order polynomial expansions parameterized by temperatures of fuel particles and graphite moderators, which is provided as follows:

$$\Sigma(T_{\text{fuel}}, T_{\text{mod}}) = \bar{\Sigma}_{\text{base}} + \sum_{n=1}^{4} e_n (T_{\text{fuel}} - \bar{T}_{\text{fuel}})^n + \sum_{n=1}^{4} f_n (T_{\text{mod}} - \bar{T}_{\text{mod}})^n \tag{20}$$

where T_{fuel} and T_{mod} are the temperatures of fuel particle and graphite moderator; $\bar{T}_{\text{fuel}}$ and $\bar{T}_{\text{mod}}$ are the basis temperatures of fuel particle and graphite moderator; $\bar{\Sigma}_{\text{base}}$ is the macroscopic cross section at basis state; e_n and f_n are the higher-order expansion coefficients of parameters.

The polynomial expansions are generated by the PANGU program [27, 28], which is developed for HTGR physics design. During the coupling calculation of neutronics and thermal hydraulics, the macroscopic cross section will be dynamically generated by fitting the real-time temperatures into the polynomial expansions.

2.3 Code framework

The DAYU3D code employs a modular design to facilitate software development and code maintenance. The code structure and modular composition are shown in Fig. 3. The top-level modules of the code include the input module, calculation module, and output module. The input and output modules primarily handle the XML-formatted input files, the restart files, and the result outputs. The calculation module manages thermal hydraulics and neutronics calculations. The thermal-hydraulic calculation comprises several sub-modules: 3D solid temperature module, multi-batch fuel pebble and particle temperature module, 3D fluid flow module, and 3D fluid temperature module. The calculation module also contains the sub-modules for initialization, matrix solving, and material property calculations.

The computation flowchart of DAYU3D is shown in Fig. 4. The Picard iteration framework is employed to address the N/TH coupling problems. Different time steps are allowed for neutronics and thermal-hydraulic calculations to accommodate their respective time scales and numerical stability requirements. A nested Picard iteration is incorporated within the inner iteration of thermal-hydraulic fields to enhance both convergence and computational efficiency.

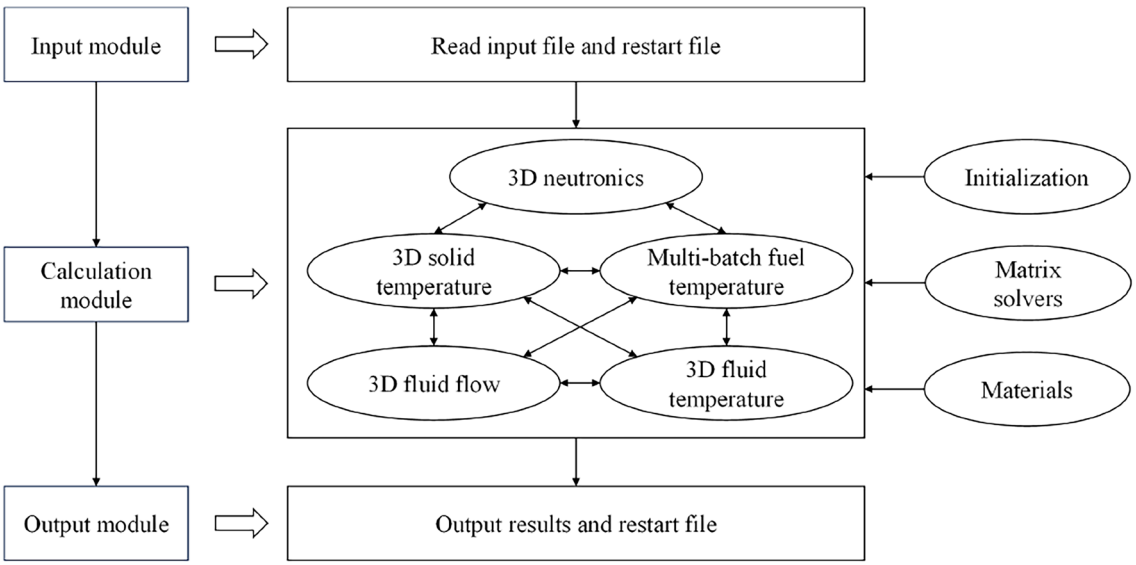

Fig. 3 Diagram of DAYU3D code structure and modular composition

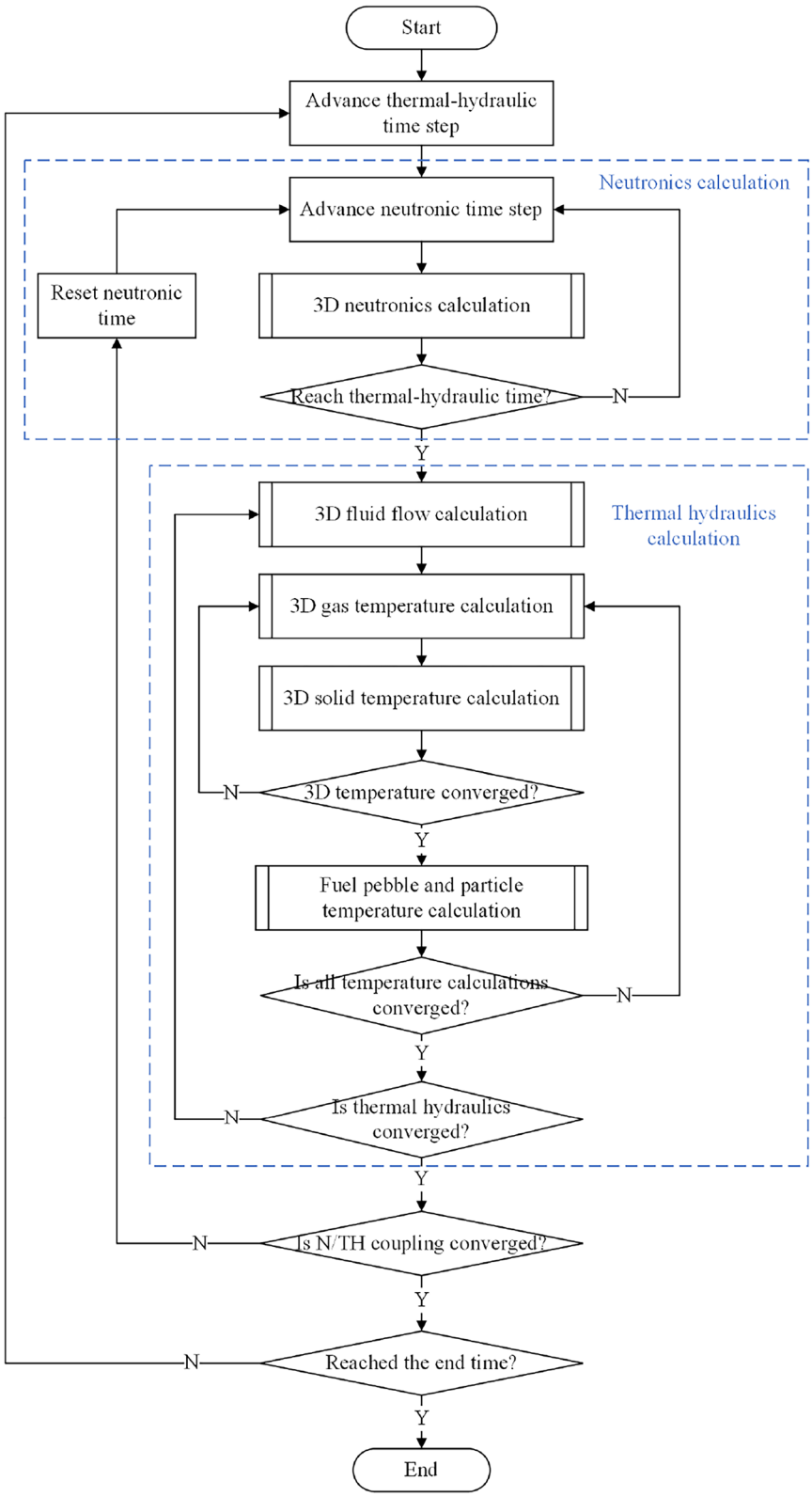

Fig. 4 Computation flowchart of DAYU3D

3 New features of DAYU3D

While ensuring high level of user-friendliness and code maintainability, DAYU3D has implemented new capabilities and computational methods beyond the traditional TINTE, THERMIX, and DAYU codes. This section summarizes some new features that have been considered in DAYU3D by now.

3.1 Modeling of control rod's continuous movement

In transient simulations, the insertion of control rods leads to drastic localized neutron flux changes, particularly near the control rod tip, known as the cusping effect. Due to the limitation of mesh division, the rod is often partially inserted into a coarse mesh in the axial direction, thereby making it necessary to homogenize mesh grid. The difficulty lies in the determination of homogenized macroscopic cross sections of the mesh grid. The simple volume-weighted scheme cannot take into consideration the drastic gradient of the local flux distribution. In order to address this issue, TINTE code introduces the ROMO model and uses Eq. (21) to calculate the weighting factor [21]. However, since the parameter S in Eq. (21) is an empirical parameter, users need to continuously adjust its value to obtain a smooth control rod worth curve [21].

$$\frac{c^*(l)}{c} = \frac{\ln\left(1 - \frac{l}{L}S\right)}{\ln(1-S)} \tag{21}$$

where l is the insertion length of control rod. L is the axial length of mesh. $c^*(l)$ refers to the absorber concentration when the rod is partially inserted. c is the absorber concentration when the rod is fully inserted. S is the interpolation factor.

In order to overcome the limitations of ROMO model, the adaptive axial splitting method is developed into DAYU3D code, which fundamentally circumvents the homogenization process. This method is summarized as follows: in the movement of the control rod, an extra splitting is imposed on the axial layer of fine mesh grids, at where the bottom of control rod is in real-time located. Once the bottom leaves this layer, the adaptive axial merging will be imposed to restore the previously split layer, while another splitting will be implemented according to the new position. Figure 5 presents this process.

It should be noted that the adaptive axial splitting method is imposed on the fine-level computation mesh, rather than the coarse-level mesh, as is mentioned in Sect. 2.2. This means the additional axial division does not have impact on the mesh mapping of neutronics and thermal-hydraulic fields.

3.2 Rigorous radiation heat transfer model

In the initial stage of DAYU3D's development, the gap radiation heat transfer model is primarily based on the TINTE code, in which the radiation heat transfer was treated as anisotropic heat conduction. The idea of this method is to equate the radiation heat transferred in each direction to the corresponding conductive heat transfer, thereby establishing the relationship between thermal conductivities in different directions [17]. Therefore, the principal components of thermal conductivity (λ_1, λ_2, λ_3) can be determined. The schematic of TINTE's heat radiation model is shown in Fig. 6. However, this method introduces an assumption based on heat radiation between two infinite surfaces when calculating λ_1 [21], as shown in Eq. (22). This assumption may lead to significant computational errors in wider gaps, such as the cavity at the top of the reactor.

$$\lambda_1 = \frac{\sigma d \varepsilon_s \left(T_1^4 - T_2^4\right)}{T_1 - T_2} \tag{22}$$

where λ_1 refers to one of the principal components of thermal conductivity; T_1 and T_2 are the temperatures of two surfaces, respectively; d indicates the width of the gap; σ

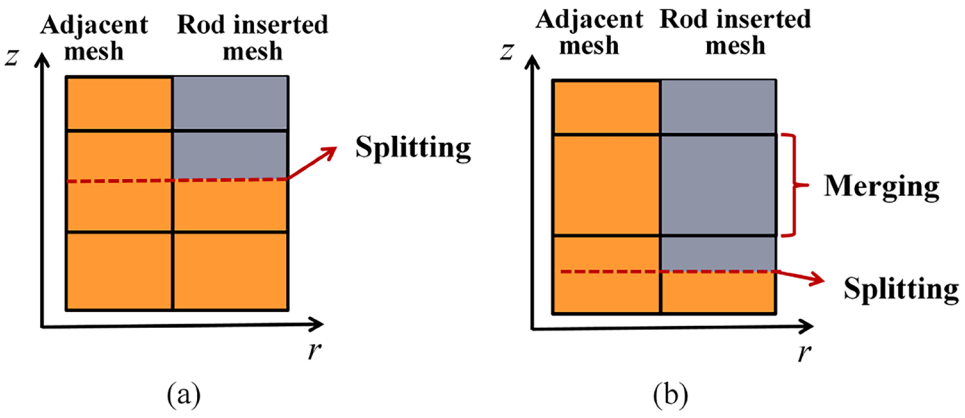

Fig. 5 (Color online) Schematic of adaptive axial splitting method. **a** Splitting at current step. **b** Merging and splitting at the next step

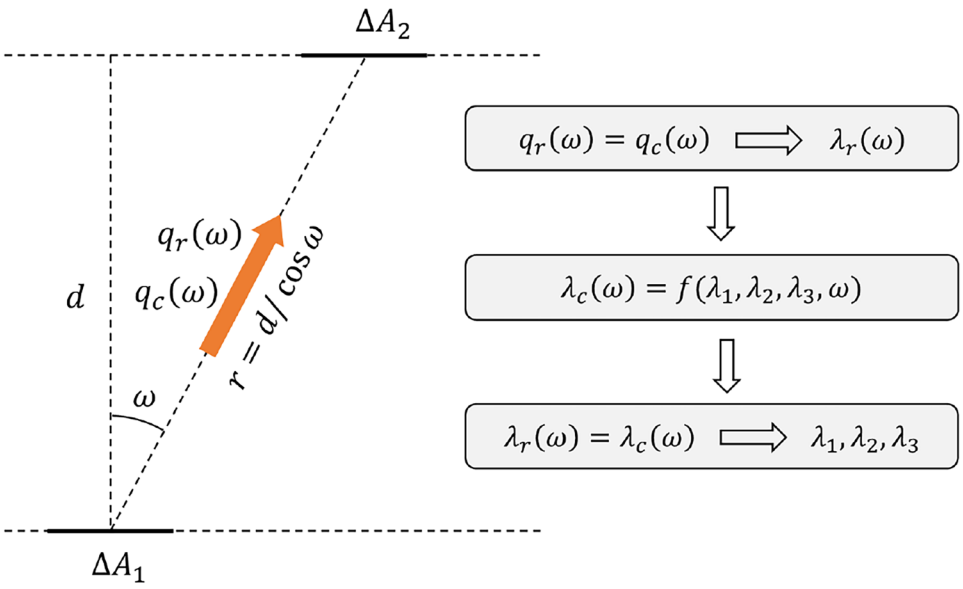

Fig. 6 Schematic of TINTE's heat radiation model

refers to the Stefan–Boltzmann constant; ε_s is the system emissivity of two surfaces.

To address the problem of TINTE's heat radiation model, a more accurate method based on 3D view factor and the net radiation method is being developed in DAYU3D. Due to the complex geometry in HTGRs, a numerical integration method is employed to compute view factors between surface elements. The calculated view factors agree well with the reference results from the Computational Fluid Dynamics (CFD) program [22].

3.3 Improved break mass flow model

In the TINTE code, the mass flow model for break conditions does not account for the effects of pipe length and inlet pressure, leading to an overestimation of the flow rate. To address this issue, a new set of formulas developed by Dou et al. [29] are incorporated into DAYU3D to consider the impact of different pipe lengths and inlet pressures on critical flow. The critical mass flow rate is calculated by Eq. (23). The numerical comparisons with CFD results indicate that these formulas provide accurate calculations [29]. It will effectively improve the accuracy of break flow simulations in the DAYU3D code.

$$M = fA\sqrt{\kappa p_0 \rho_0}\left(\frac{2}{\kappa+1}\right)^{\frac{\kappa+1}{2(\kappa-1)}} \tag{23}$$

where A is the cross section area of break; κ refers to the adiabatic index; p_0 and ρ_0 are the pressure and density of gas at the inlet of pipe; f is a correction factor calculated as follows:

If $L < 2$ m:

$$f = 0.8123 - 0.0472e^{-4.52p_0} \tag{24}$$

If 2 m $\le L \le$ 10 m:

$$f = 11.72 - 10.87e^{0.0017L} - 1.17e^{-0.35p_0} + 1.087e^{0.0017L-0.35p_0} \tag{25}$$

If $L >$ 10 m:

$$f = 0.3226 + 0.4881e^{-0.032L} - 0.01523e^{-4.52p_0} \tag{26}$$

where L refers to the length of the pipe.

3.4 Efficient methods for solving flow field in HTGRs

In traditional TINTE code, the fluid pressure equation is typically solved using TINTE matrix ordering and Gauss elimination [17]. However, this fixed solution strategy may be inefficient under complex conditions. To address this, the DAYU3D code introduces an efficient flow solver based on advanced matrix ordering, symbolic factorization, and block matrix techniques.

In the DAYU3D matrix ordering method, the longest path in one-dimensional (1D) flow regions is prioritized and placed at the front of the matrix. This matrix ordering method is termed as "One-Dimensional Flow-path Priority Ordering" (ODFPO). The ODFPO reduces the number of nonzero fill-ins during the Gaussian elimination, and makes the reordered matrix exhibit a block-structured characteristic [30]. Therefore, the ODFPO method can be effectively combined with symbolic factorization and block matrix solving, respectively, enhancing the computational efficiency for 2D and 3D simulations. The

computational time of DAYU3D is significantly reduced compared to that of TINTE [30].

3.5 Global and parallel calculation of multi-batch fuel temperature

In the calculation of multi-batch fuel temperature, the TINTE code adopts a batch-by-batch and grid-by-grid iteration approach, which results in low computational efficiency. In the DAYU3D code, the temperatures of all batches of fuel pebbles within one grid are solved simultaneously, to avoid the iteration between batches and improve the calculation efficiency. Additionally, the DAYU3D code supports parallel computation of fuel pebble temperatures across all pebble-bed meshes, so the efficiency can be further enhanced.

4 Code verification and validation

Since the DAYU3D code is developed from the scratch, all the equations and models of thermal-hydraulic modules in DAYU3D have been verified and validated through a series of constructed test cases, the SANA experiment, and the HTGR cases.

4.1 Constructed cases

The verification matrix of thermal-hydraulic calculation is shown in Table 1. First, verification was carried out for individual calculation modules covering 1D, 2D, and 3D conditions, different materials, diverse flow regions, distinct boundary conditions, and both steady state and transient states. After the verification of individual modules, the verification of some coupled modules was conducted. Reference solutions for the verification include analytical solutions, as well as the calculated results by the TINTE code using the same input parameters. The verification results of DAYU3D are summarized in Table 2. The relative deviation of DAYU3D compared to the reference solution is below 1% and mostly below 0.2%.

Table 2 Maximum relative deviation in verification cases of DAYU3D

Bin of maximum relative deviation	Frequency			
	Solid temperature	Pebble temperature	Gas pressure	Gas temperature
0.0–0.2%	35	8	29	11
0.2–0.4%	3	2	4	2
0.4–0.6%	1	0	1	1
0.6–0.8%	1	0	1	1
0.8–1.0%	2	0	0	0

Among all the cases, a fully coupled example is presented in more detail to demonstrate DAYU3D's accuracy. It is a simplified test case simulating the steady-state condition of a pebble-bed HTGR with 15 batches of fuel elements. The simulation results were compared against those from TINTE, as shown in Fig. 7. The solid temperature, helium temperature, and particle temperature exhibit good agreement with the results from TINTE.

4.2 SANA experiments

SANA experiments were conducted in 1990 s in Germany with the objective of illustrating the mechanisms of heat transfer in the pebble bed [31]. It consists of a pebble bed, some heating elements, insulating materials, and a pressure vessel. Some thermocouples are installed in the experimental facility to measure the temperature.

In SANA experiments, a comprehensive range of conditions have been tested [32, 33]. It includes not only 2D heating conditions but also 3D heating conditions, making it highly suitable for validating 3D thermal-hydraulic codes for HTGRs. The DAYU3D code has performed calculations for all the conditions involving helium filled and 60 mm graphite pebbles. Some example cases are provided to demonstrate the accuracy of DAYU3D.

The comparison between the calculated results of the DAYU3D code and the experimental data for the 2D

Table 1 Verification matrix of DAYU3D thermal-hydraulic calculation

Calculation module	Verification of single physical field				Verification of multiple coupled physical field		
	Heat conduction and radiation	Single and multiple batches	Gas flow	Gas temperature	Couple of solid and fuel pebble	Couple of gas flow and gas temperature	Couple of all modules
Solid temperature	✓	–	–	–	✓	–	✓
Pebble and particle temperature	–	✓	–	–	✓	–	✓
Gas flow	–	–	✓	–	–	✓	✓
Gas temperature	–	–	–	✓	–	✓	✓

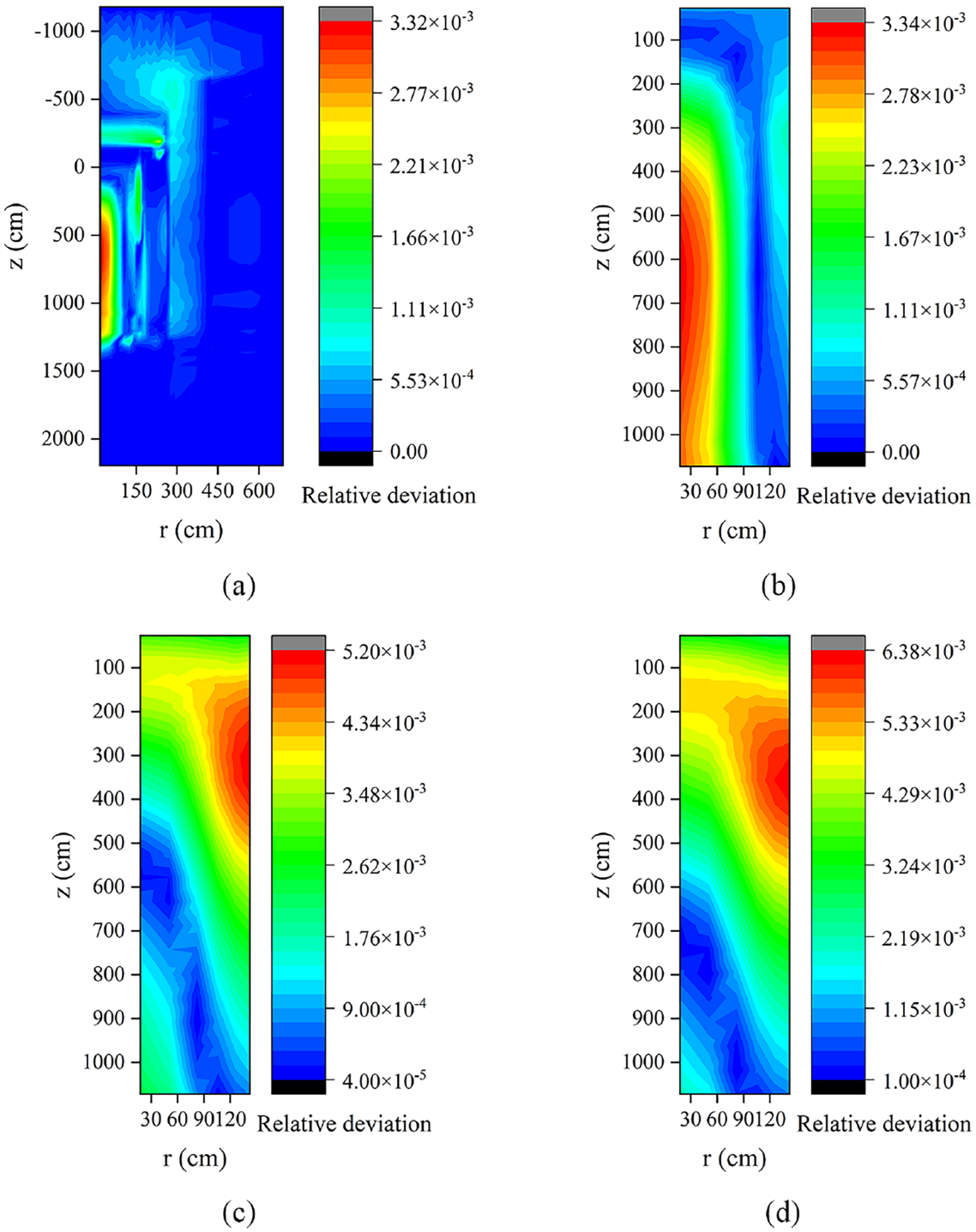

Fig. 7 (Color online) Relative deviation between DAYU3D and TINTE results. **a** Solid temperature. **b** Core gas temperature. **c** Central particle temperature of the 1st batch pebbles. **d** Central particle temperature of the 15th batch pebbles

conditions of SANA is shown in Fig. 8. For the 2D conditions including full-length heating, upper-section heating, and lower-section heating of the central heating element, the results of DAYU3D accurately reflect the temperature distribution within the pebble bed.

In 3D conditions, taking the cases of central 0 kW with radial 3×9 kW (0 + 3×9 kW) heating power and central 10 kW with radial 3×5 kW (10 + 3×5 kW) heating power as examples, a comparison between DAYU3D and SANA experimental data is shown in Figs. 9 and 10. The results

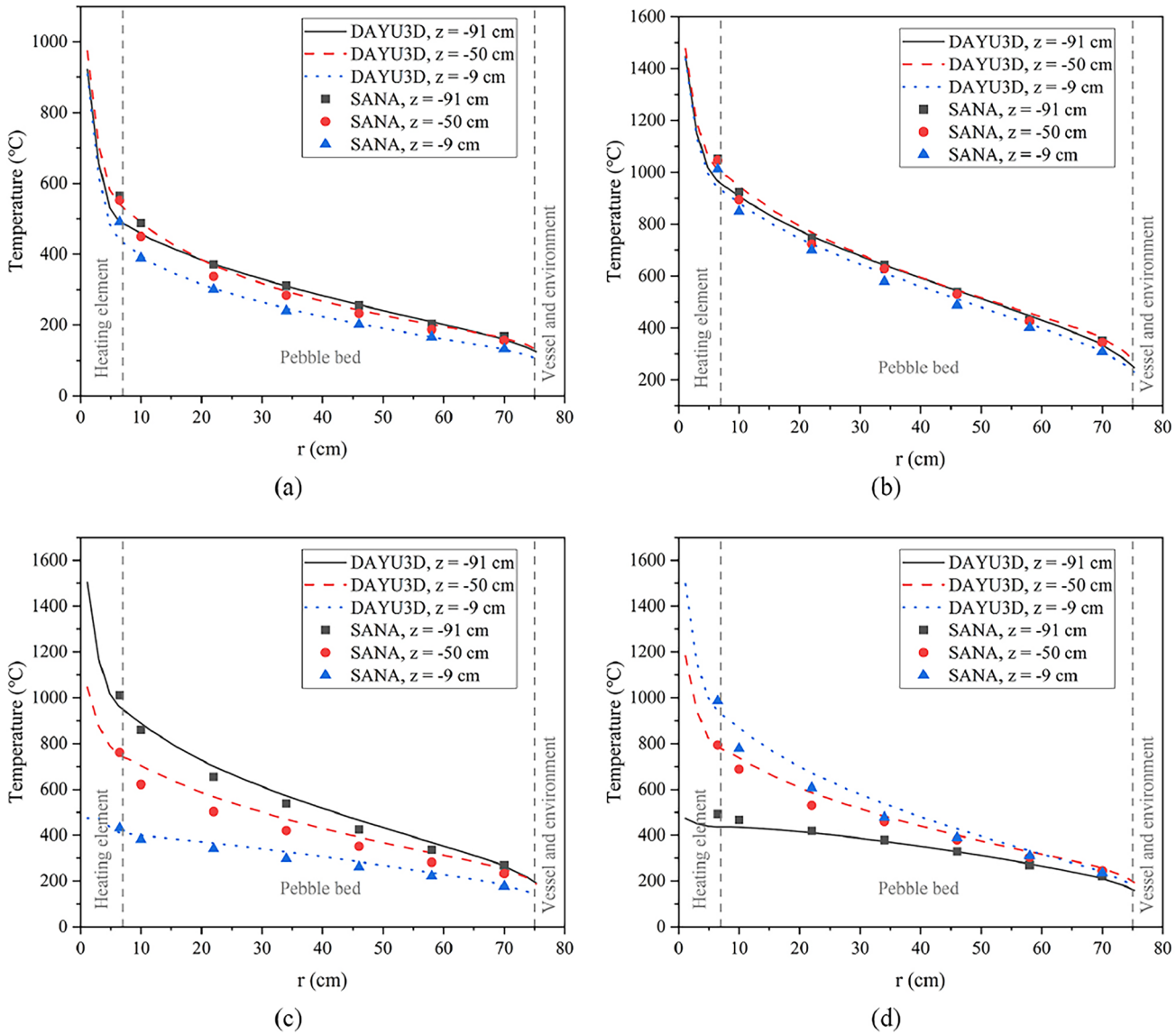

Fig. 8 Comparison of DAYU3D to SANA in 2D conditions. **a** Central full-length heating: 10 kW. **b** Central full-length heating: 30 kW. **c** Central upper-section heating: 20 kW. **d** Central lower-section heating: 20 kW

illustrate that the DAYU3D code accurately calculates the 3D temperature distribution caused by the radial heating elements, showing good agreement with the experimental data.

4.3 3D HTR-PM case

Figures 11 and 12 show the 3D calculation model of HTGR, with the meshing and material settings. The helium inlet and outlet of the core are located only in the first sector. It is assumed that the control rod channels are cooled by helium, whereas the absorber ball channels remain uncooled. The coolant mass flow rate is 96 kg/s, with the temperature at the core inlet set to 250 °C and the core outlet pressure maintained at 7 MPa. All the control rods are inserted at $z =$ 275 cm, while the absorber balls are not dropped.

The solid temperature distribution within the pebble bed is shown in Fig. 13. As helium flows from top to bottom of the pebble bed, the temperature at the bottom of the core is higher, while the temperature at the top is lower. Additionally, due to the influence of the control rod and absorber ball channels, the temperature at the periphery of the pebble bed is higher near $\theta = 0°$, $\theta = 60°$ and $\theta = 120°$, while the temperature in other sectors is lower. The solid temperature near 180° is generally lower than that near 0°, primarily due to the influence of the helium inlet in the core.

The temperature distribution on the sidewall of pressure vessel is shown in Fig. 14. The vessel temperature

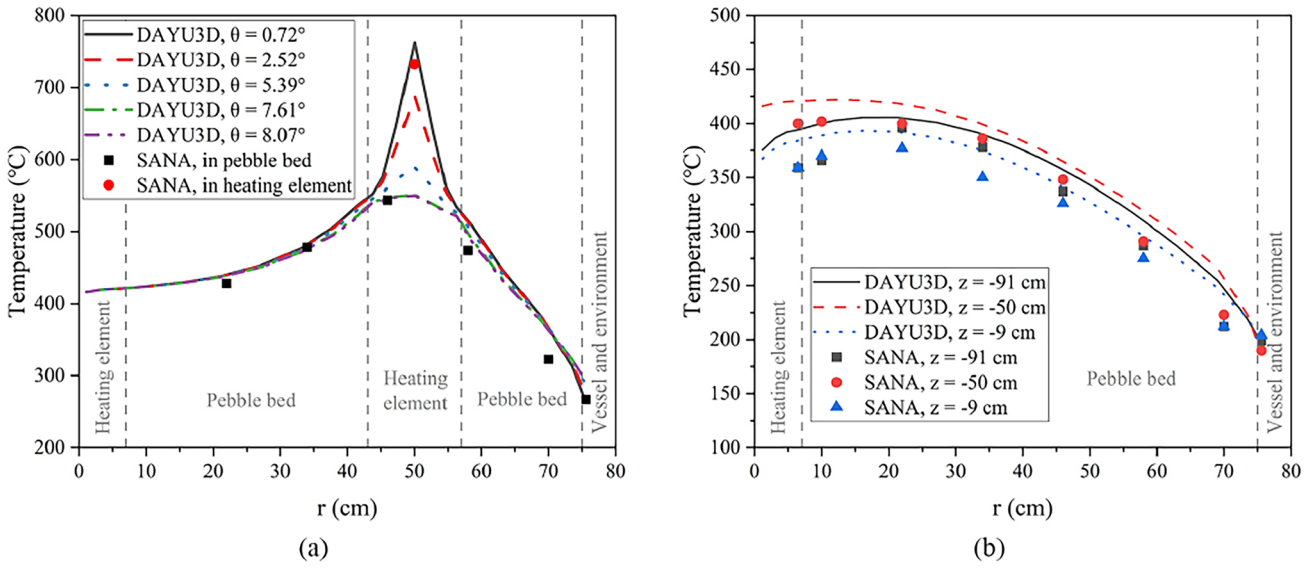

Fig. 9 (Color online) Comparison of DAYU3D to SANA at 0 + 3×9 kW heating power. **a** Sectors near the radial heating elements ($0° < \theta < 8.11°$). **b** Sector far from the radial heating elements ($57° < \theta < 60°$)

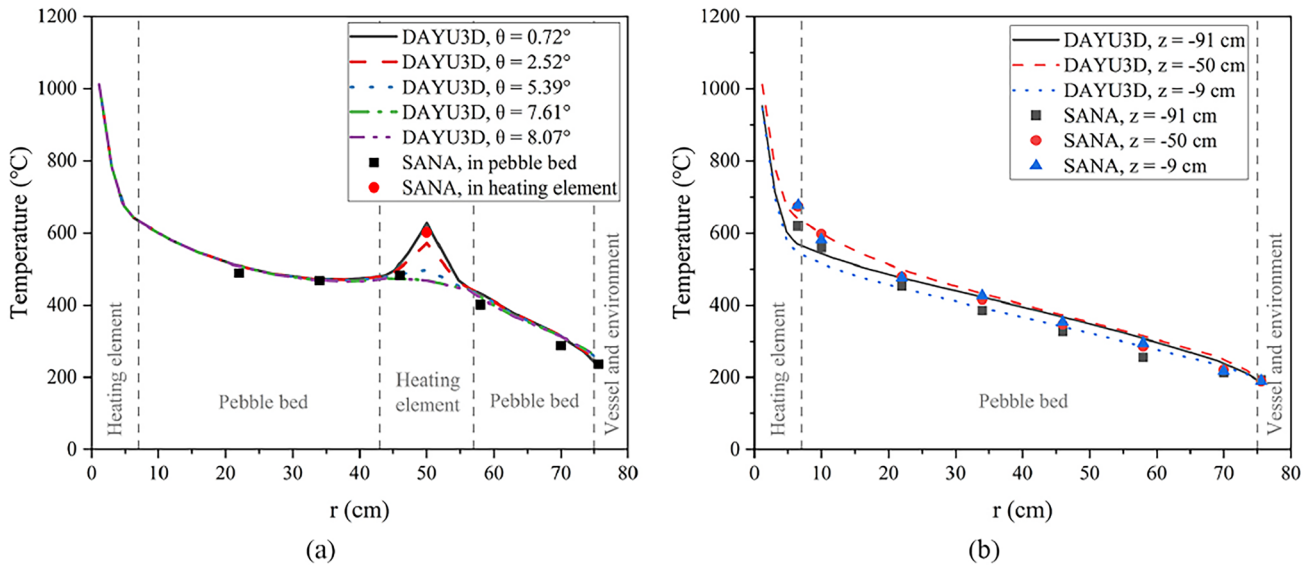

Fig. 10 (Color online) Comparison of DAYU3D to SANA at 10 + 3×5 kW heating power. **a** Sectors near the radial heating elements ($0° < \theta < 8.11°$). **b** Sector far from the radial heating elements ($57° < \theta < 60°$)

distribution exhibits two characteristics. First, the temperature is lower in the upper region and higher in the lower region. Second, the temperature is higher near the helium inlet and lower farther from it. The first characteristic is mainly influenced by the temperature distribution in the pebble-bed region, where the upper core temperature is lower and the lower core temperature is higher. The second characteristic is mainly because the 250 °C helium entering the reactor pressure vessel heats the vessel. Therefore, the helium inlet affects not only the temperature distribution of the pressure vessel but also that of the pebble-bed region.

4.4 Performance analysis

Typical test cases on HTR-PM, including steady state and depressurized loss of forced cooling accident (DLOFC), are calculated to examine the computational efficiency of the DAYU3D code. In the DLOFC case, the coolant

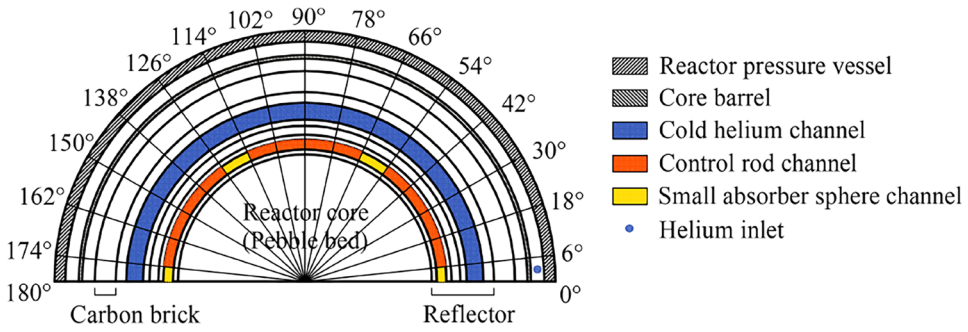

Fig. 11 (Color online) Meshing of the reactor inside pressure vessel in r-θ plane

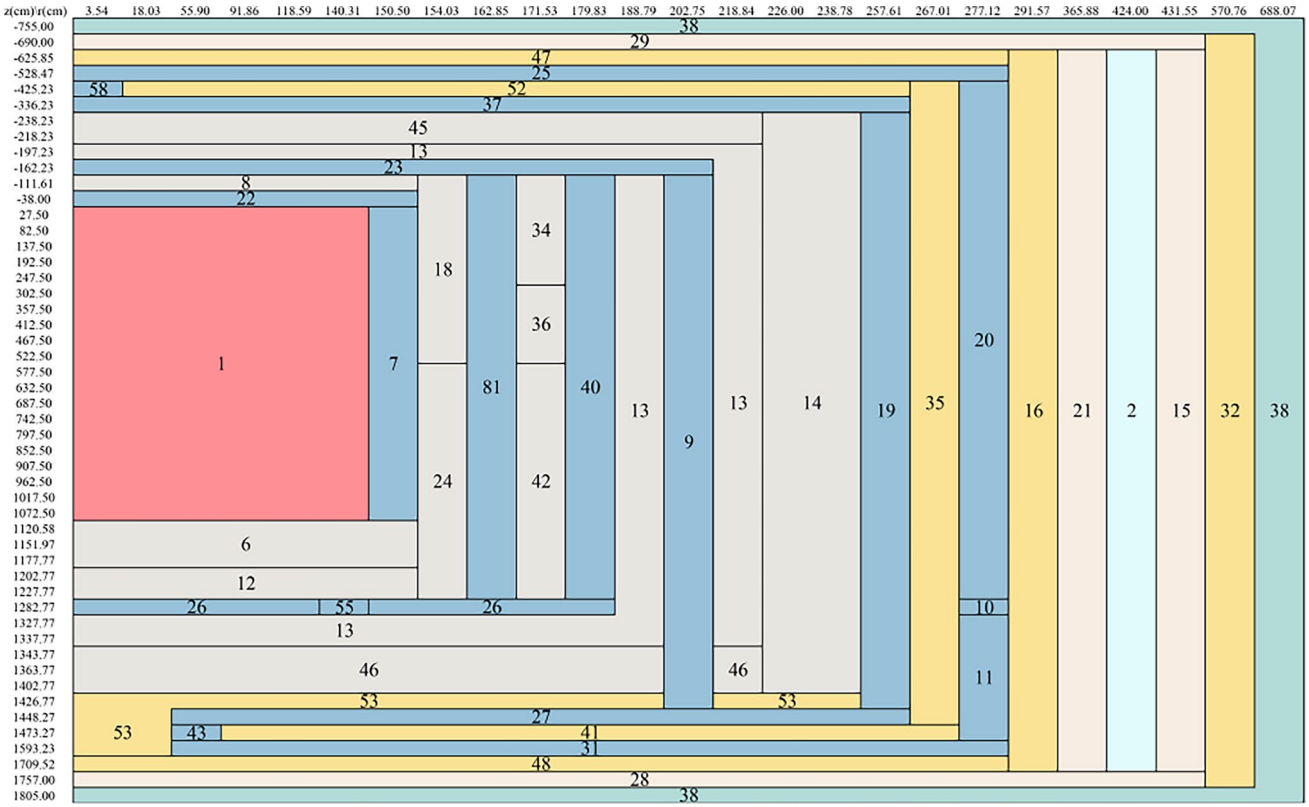

Fig. 12 (Color online) Meshing and material setting in the first sector ($0° < \theta < 6°$) in r-z plane. (Material setting: 1–Core; 2–Water panel; 6,8,12,13,18,24,34,36,42–Reflector; 7–Material between pebble bed and reflector; 9–Cold helium channel; 10–Helium inlet; 11,15,19,20,21,25,28,29,37–Gap; 14,45,46–Carbon brick; 16,47,48–Reactor pressure vessel; 22–Core top cavity; 23–Cold helium plenum; 26–Hot helium plenum; 27–Core bottom cavity; 31–Pressure vessel bottom cavity; 32–Cavity concrete wall; 35,41,52,53–Metal internals and core barrel; 38–Air boundary; 40–Leakage flow; 43–Bottom metallic supporter flow passage; 55–Helium outlet; 81–Small absorber sphere channel

pressure is assumed to drop instantaneously to 1 atm at the onset of the accident. To ensure a fair comparison, all the codes employ identical model setups, and all the cases are executed sequentially without using parallel acceleration. The calculation time of TINTE, MGT-3D, and DAYU3D is compared in Table 3. The total calculation time of DAYU3D is reduced by about 90% compared to TINTE in the 2D cases. As for the 3D cases, the time reduction is also remarkable compared with MGT-3D. The efficiency advantage of the DAYU3D code can be attributed to two main factors. First, at the algorithm level, DAYU3D employs more advanced linear system solvers, such as the efficient flow field solving method. Second, at the implementation level, extensive optimizations have been applied to the data structures and memory management, reducing the overhead associated with dynamic memory allocation and improving overall computational performance.

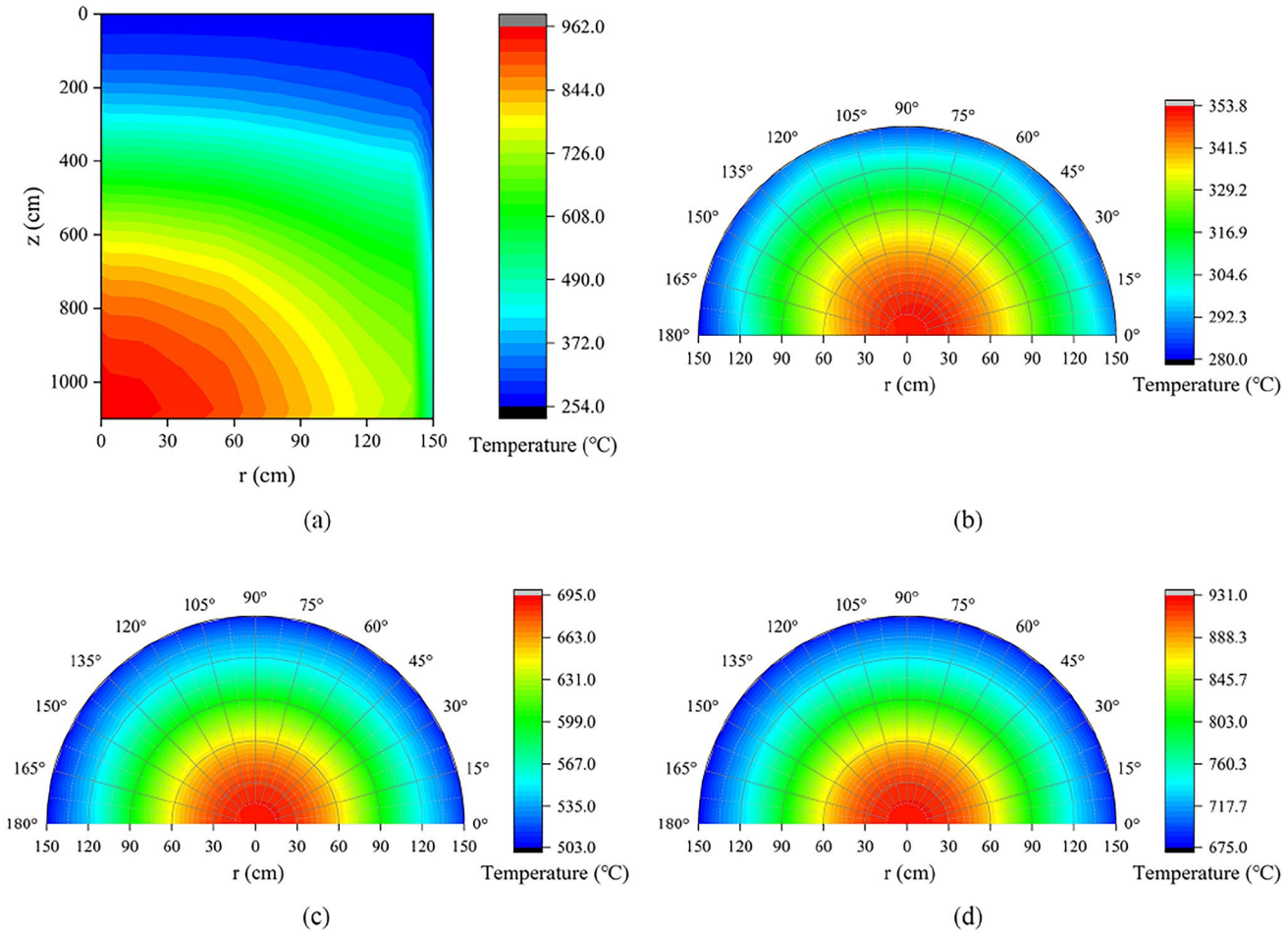

Fig. 13 (Color online) Solid temperature distribution within the pebble bed. **a** Distribution in r-z plane of the first sector ($0° < \theta < 6°$). **b** Distribution in r-θ plane at $z = 192.5$ cm. **c** Distribution in r-θ plane at $z = 522.5$ cm. **d** Distribution in r-θ plane at $z = 907.5$ cm

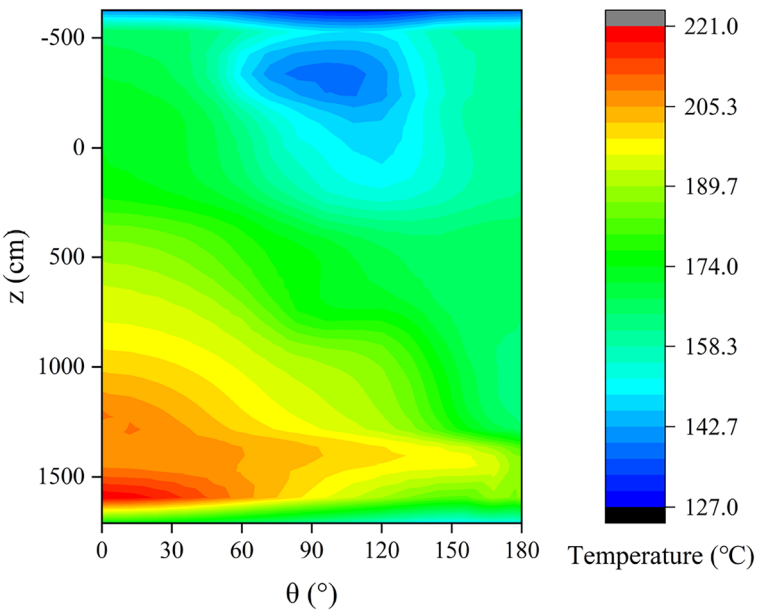

Fig. 14 (Color online) Temperature distribution on the sidewall of pressure vessel

Table 3 Comparison of efficiency among TINTE, MGT-3D, and DAYU3D

Cases	Total calculation time			
	TINTE	MGT-3D	DAYU3D	Time reduction
2D steady state	7 s	–	0.6 s	91.4%
2D DLOFC	216 s	–	18.4 s	91.5%
3D steady state	–	107 s	9.7 s	90.9%
3D DLOFC	–	249 s	83.8 s	66.3%

5 Conclusion

DAYU3D is a modern and advanced tool for thermal-hydraulic design of HTGRs. This paper presented the mathematical models, code framework and new features of the DAYU3D code, as well as the numerical results for code verification and validation.

DAYU3D has made significant improvements over the traditional THERMIX and TINTE codes. These include 3D calculation capabilities, continuous movement of control rods, more rigorous model of radiation heat transfer and break mass flow, advanced and efficient algorithms, improved user-friendliness and code maintainability, etc. Besides, DAYU3D has undergone extensive verification and validation using more than 100 cases. For most cases, the relative deviation from the reference solution is within 0.2%. The calculation time for HTR-PM cases is reduced by over 60% compared to TINTE. It demonstrated DAYU3D's good accuracy and significant advantage in computational efficiency.

In the future, the DAYU3D code will be further extended to enable more refined evaluation of residual heat, non-local energy deposit, chemical corrosion, etc. It will also be coupled with the PANGU code [27, 28] for 3D neutronics, thermal hydraulics, and fuel cycle simulation.

Acknowledgements The authors would like to thank Dr. Bo Tan, Dr. Xin-Sheng Xu and Dr. Hao-Ming Dou for their valuable contributions in the development of the DAYU3D code.

Data availability Data will be made available on request.

Declarations

Conflict of interest Ding She is an editorial board member for Nuclear Science and Techniques and was not involved in the editorial review, or the decision to publish this article. All authors declare that there are no conflict of interest.

References

1. J. Banaschek, *Berechnungsmethoden und Analysen zum dynamischen Verhalten von Kraftwerksanlagen mit Hochtemperaturreaktor* (Zentralbiliothek, Verlag, Jülich, Kernforschungsanlage Jülich GmbH, 1983)
2. J.C. Cleveland, S.R. Greene, *Application of THERMIX-KONVEK Code to Accident Analyses of Modular Pebble Bed High Temperature Reactors (HTRs)* (Oak Ridge National Lab, TN (USA), 1986)
3. H. Gerwin, W. Scherer, E. Teuchert, The TINTE modular code system for computational simulation of transient processes in the primary circuit of a pebble-bed high-temperature gas-cooled reactor. Nucl. Sci. Eng. **103**, 302–312 (1989). https://doi.org/10.13182/NSE89-A23682
4. K. Petersen, *Zur Sicherheitskonzeption des Hochtemperaturreaktors mit natürlicher Wärmeableitung aus dem Kern im Störfall* (Zentralbiliothek, Verlag, Jülich, Kernforschungsanlage Jülich GmbH, 1983)
5. H. Gerwin, *Das zweidimensionale Reaktordynamikprogramm TINTE Teil 2: Anwendungsbeispiele* (Zentralbiliothek, Verlag, Jülich, Kernforschungsanlage Jülich GmbH, 1989)
6. F. Reitsma, G. Strydom, J.B.M. de Haas et al., The PBMR steady-state and coupled kinetics core thermal-hydraulics benchmark test problems. Nucl. Eng. Des. **236**, 657–668 (2006). https://doi.org/10.1016/j.nucengdes.2005.12.007
7. Z. Gao, L. Shi, Thermal hydraulic calculation of the HTR-10 for the initial and equilibrium core. Nucl. Eng. Des. **218**, 51–64 (2002). https://doi.org/10.1016/S0029-5493(02)00198-X
8. Z. Gao, L. Shi, Thermal hydraulic transient analysis of the HTR-10. Nucl. Eng. Des. **218**, 65–80 (2002). https://doi.org/10.1016/S0029-5493(02)00199-1
9. Y. Zheng, L. Shi, Y. Dong, Thermohydraulic transient studies of the Chinese 200 MWe HTR-PM for loss of forced cooling accidents. Ann. Nucl. Energy **36**, 742–751 (2009). https://doi.org/10.1016/j.anucene.2009.02.007
10. Y. Zheng, M.M. Stempniewicz, Z. Chen et al., Study on the DLOFC and PLOFC accidents of the 200 MWe pebble-bed modular high temperature gas-cooled reactor with TINTE and SPECTRA codes. Ann. Nucl. Energy **120**, 763–777 (2018). https://doi.org/10.1016/j.anucene.2018.06.041
11. D. Shi, Extension of the reactor dynamics code MGT-3D for pebblebed and blocktype high-temperature-reactors. RWTH Aachen University (2015)
12. K. Hossain, M. Buck, W. Bernnat et al., TH3D: a three-dimensional thermal hydraulic tool, for design and safety analysis of HTRS, In *Proceedings of the Fourth International Topical Meeting on High Temperature Reactor Technology*, *Volume 1,* Washington, DC, USA, 28 September–1 October 2009, pp. 709–724
13. A. Seubert, A. Sureda, J. Lapins et al., The transient 3-D transport coupled code TORT-TD/ATTICA3D for high-fidelity pebble-bed HTGR analyses. Trans. Theory Stat. Phys. **41**, 133–152 (2012). https://doi.org/10.1080/00411450.2012.671212
14. A.J. Novak, R.W. Carlsen, S. Schunert et al., Pronghorn: a multidimensional coarse-mesh application for advanced reactor thermal hydraulics. Nucl. Technol. **207**, 1015–1046 (2021). https://doi.org/10.1080/00295450.2020.1825307
15. Y. Zheng, L. Shi, F. Chen, Coated fuel particle temperature analysis of the pebble-bed modular high temperature gas-cooled reactor, In: Guiyang, China (2011)
16. H. Dou, M. Lang, H. Zhang et al., Analysis and improvement of flow field calculation in thermal hydraulic program for high temperature gas-cooled reactor. American Soc. Mech. Eng. Dig. Collect. (2022). https://doi.org/10.1115/ICONE29-92090

17. H. Gerwin, *Das zweidimensionale Reaktordynamikprogramm Tinte Teil 1: Grundlagen und Lösungsverfahren* (Zentralbiliothek, Verlag, Jülich, Kernforschungsanlage Jülich GmbH, 1987)
18. P. Zehner, E.U. Schlünder, Einfluß der Wärmestrahlung und des Druckes auf den Wärmetransport in nicht durchströmten Schüttungen. Chem. Ing. Tec. **44**, 1303–1308 (1972). https://doi.org/10.1002/cite.330442305
19. K. Robold, *Wärmetransport im inneren und in der Randzone von Kugelschüttungen* (Zentralbiliothek, Verlag, Jülich, Kernforschungsanlage Jülich GmbH, 1982)
20. H. Barthels, *Die effektive Wärmeleitfähigkeit in Kugelschüttungen unter besondererBerücksichtigung des Hochtemperatur-Reaktors* (Zentralbiliothek, Verlag, Jülich, Kernforschungsanlage Jülich GmbH, 1984)
21. H. Gerwin, W. Scherer, A. Lauer et al., *TINTE—A Two-dimensional Code for Reactor Dynamics* (Verlag, Jülich, Forschungszentrum Jülich GmbH Zentralbibliothek, 2009)
22. X. Xu, H. Zhang, Y. Zheng et al., Three-dimensional radiation view factor calculation in dayu3d program, In: Beijing, China (2024)
23. Nuclear safety standards commission (KTA), KTA 3102.1 Reactor core design for high-temperature gas-cooled reactor. Part 1: calculation of the material properties of helium (1978)
24. Nuclear safety standards commission (KTA), KTA 3102.3 reactor core design of high-temperature gas-cooled reactors. Part 3: loss of pressure through friction in pebble bed cores (1981)
25. H. Zhang, D. She, L. Shi, Progress on the development of DAYU3D code for HTR thermal-hydraulic design and system analysis, in: american society of mechanical engineers digital collection, Prague, Czech Republic (2024). https://doi.org/10.1115/ICONE31-134375.
26. H. Gerwin, W. Scherer, A. Lauer et al., *TINTE—Nuclear Calculation Theory Description Report* (Verlag, Jülich, Forschungszentrum Jülich GmbH Zentralbibliothek, 2010)
27. D. She, B. Xia, J. Guo et al., Prediction calculations for the first criticality of the HTR-PM using the PANGU code. Nucl. Sci. Tech. **32**, 90 (2021). https://doi.org/10.1007/s41365-021-00936-5
28. D. She, J. Guo, Z. Liu et al., PANGU code for pebble-bed HTGR reactor physics and fuel cycle simulations. Ann. Nucl. Energy **126**, 48–58 (2019). https://doi.org/10.1016/j.anucene.2018.11.005
29. H. Dou, M. Lang, Y. Ji, Study on the critical flow of removal pipe breakage in high-temperature gas-cooled reactor. Prog. Nucl. Energy **176**, 105367 (2024). https://doi.org/10.1016/j.pnucene.2024.105367
30. H. Zhang, B. Tan, D. She et al., An efficient method for solving flow field in high temperature gas-cooled reactor. Prog. Nucl. Energy **180**, 105599 (2025). https://doi.org/10.1016/j.pnucene.2024.105599
31. B. Stoecker, H.F. Niessen, *Data Sets of the SANA Experiment 1994–1996* (Forschungszentrum, Jülich Institute for Safety Research and Reactor Technology, Jülich, 1997)
32. M. Lange, *Experimente zur selbsttätigen Abfuhr der Nachwärme bei Hochtemperatur-Reaktoren: Planung* (Vorbereitung und Ergebnisse, Forschungszentrum, Jülich Institute for Safety Research and Reactor Technology, Jülich, 1995)
33. J. Baggemann, D. Shi, S. Kasselmann, S. Kelm et al., Use of SANA experimental data for validation and verification of MGT-3D and a CFD porous media model for VHTR application. Nucl. Eng. Des. **305**, 678–687 (2016). https://doi.org/10.1016/j.nucengdes.2016.05.030