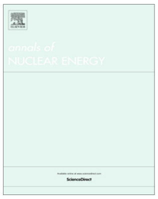
Annals of Nuclear Energy 169 (2022) 108943

Contents lists available at ScienceDirect

Annals of Nuclear Energy

journal homepage: www.elsevier.com/locate/anucene



Development and preliminary application of deterministic code NECP-FISH for neutronics analysis of fusion-reactor blanket

Jianxin Miao [a,b], Chao Fang [b], Chenghui Wan [b,*], Wanchao Mao [a], Hongyun Xie [a]

[a] *State Key Laboratory of Nuclear Power Safety Monitoring Technology and Equipment, China Nuclear Power Engineering Co Ltd, Shenzhen of Guangdong Prov, 518172, China*
[b] *School of Nuclear Science and Technology, Xi'an Jiaotong University, Xi'an, Shaanxi 710049, China*

ARTICLE INFO

Article history:
Received 9 September 2021
Received in revised form 20 December 2021
Accepted 23 December 2021

Keywords:
Fusion-reactor blanket
Deterministic method
Neutronics analysis
NECP-FISH

ABSTRACT

The fusion-reactor blanket is very important, as it is responsible for the tritium self-sustaining, energy gain, and radiation shielding. Due to the complex structure, large geometry size and inhomogeneous neutron-flux distribution, the Monte-Carlo method is widely used for the neutronics analysis of fusion-reactor blanket, but it takes a large amount of computational time for acceptable simulation results. The deterministic code is the better choice for the fusion-reactor blanket. The geometry capability and accuracy limitation are the most important issues for the deterministic code to simulate the fusion-reactor blanket. Therefore, the newly deterministic code named NECP-FISH has been developed for the fusion-reactor blanket, in which the spherical harmonic function and finite element method were applied. Moreover, the open-source platform SALOME has been applied to generate the complex geometry as pre-process of NECP-FISH. As code verification, NECP-FISH has been applied to simulate the Breeding Unit of HCCB-DEMO, using Monte-Carlo code to provide the reference results. It can be observed that the simulation results of the tritium breeding ratio (TBR), neutron flux and heat release rate provided by NECP-FISH are agreed well with corresponding values by the Monte-Carlo code.

1. Introduction

The fusion-reactor blanket is very important, as it is responsible for the tritium self-sustaining, energy gain, and radiation shielding. However, the geometry of the fusion-reactor blanket is very complicated, as there are many irregular structures applied in the blanket. Therefore, the Monte-Carlo method has been widely applied to perform the neutronics analysis of the fusion-reactor blanket, because of its advantage in powerful capability for complex geometry. As improvement in geometry modeling, the Monte-Carlo codes McCad (Große et al., 2013) and MCAM (Wu, 2009) have been updated to automatically generate the input card based on the CAD files of fusion-reactor blanket. However, large amount of particles are required for the Monte-Carlo code to simulate the fusion-reactor blanket, which takes a long computational time. Therefore, the deterministic method is another choice for the modeling and simulation of the fusion-reactor blanket.

For the deterministic method, the irregular structures in the fusion-reactor blanket bring challenges to the neutronics modeling and simulations. Only a few codes, such as the TORT (Rhoades and Simpson, 1997) and Hydra (McGhee et al., 2007), have been applied in the calculation of fusion-reactor blanket. It is the issue for the deterministic method that how to characterize the irregular structures of the fusion-reactor blanket. Due to the notable advantage in treating the irregular-structure geometry, the finite-element method can be adopted in deterministic method for the fusion-reactor blanket. The deterministic code named ATTILA (Zhang et al., 2020) based on the finite-element method has been preliminarily applied for the neutronics analysis of ITER (International Thermonuclear Experimental Reactor). The discontinuous finite-element code named as AETIUS, developed by KAERI in South Korea, applied the unstructured mesh to calculate the Korea helium-cooled ceramic reflector test blanket module (Kim, 2014). These successful applications of finite-element method indicated its capability for the fusion-reactor blanket.

Therefore, a newly deterministic code named as NECP-FISH has been designed and developed for the fusion-reactor blanket, which applies the finite-element method for the complicated and irregular-structure geometry and the spherical harmonic function method to discretize the angular variable to avoid the ray effect. Moreover, in order to simplify the geometry-modeling process using the finite-element method, we applied the open-source platform SALOME ("SALOME: The open-source integration platform for numerical simulation." http://www.salome-platform.org, 2005) for the geometry pre-process and the simulation-result post-process.

* Corresponding author.
E-mail address: wan.ch@mail.xjtu.edu.cn (C. Wan).

https://doi.org/10.1016/j.anucene.2021.108943
0306-4549/© 2021 Elsevier Ltd. All rights reserved.

As code verification, the preliminary neutronics simulation for the Breeder Unit of helium-cooling ceramic breeder (HCCB) for fusion DEMO (Feng, 2009) has been performed by NECP-FISH, with the reference results provided by the Monte-Carlo code. It can be observed from the numerical results that the simulation results of neutron flux, tritium breeder ratio and heat release rate provided by NECP-FISH were agreed well with corresponding reference values by the Monte-Carlo code.

2. Theory and method

In NECP-FISH, the spherical harmonic function is applied to solve the first-order steady-state neutron-transport equation, and the stabilized finite-element method is applied for the geometry mesh. This paper only gives a brief description of the algorithm used in NECP-FISH, the detailed derivation process can be seen in references (Cao, et al., 2021; Cao, 2020; Fang et al., 2020).

It should be noted that for the fusion reactor, there is no fission-source term in the right-hand side of the neutron-transport equation. Therefore, the mono-energetic neutron-transport equation can be characterized as Eq. (1).

$$\boldsymbol{\Omega}\cdot\nabla\phi(r,\boldsymbol{\Omega})+\Sigma_t\phi(r,\boldsymbol{\Omega})=\int_{\mathbb{S}^2}\Sigma_s(r,\boldsymbol{\Omega}'\cdot\boldsymbol{\Omega})\phi(r,\boldsymbol{\Omega}')d\boldsymbol{\Omega}'+s_e(r,\boldsymbol{\Omega}). \tag{1}$$

where $\phi(\boldsymbol{r},\boldsymbol{\Omega})$ denotes the neutron angular flux, Σ_t and Σ_s represent the total and scattering cross section respectively, $s_\mathrm{e}(\boldsymbol{r},\boldsymbol{\Omega})$ is the external-source term. Move the scattering-source term from the right-hand side to the left-hand side, and then define the transport operator $\boldsymbol{L}$ as Eq. (2):

$$\boldsymbol{L}\phi(\boldsymbol{r},\boldsymbol{\Omega})=\boldsymbol{\Omega}\cdot\nabla\phi(\boldsymbol{r},\boldsymbol{\Omega})+\boldsymbol{\Sigma}\phi(\boldsymbol{r},\boldsymbol{\Omega}), \tag{2}$$

where $\boldsymbol{\Sigma}\phi(\boldsymbol{r},\boldsymbol{\Omega})$ can be characterized as Eq. (3):

$$\Sigma\phi(r,\Omega)=\Sigma_t(r)\phi(r,\Omega)-\int_{\mathbb{S}^2}\Sigma_s(r,\Omega'\cdot\Omega)\phi(r,\Omega')d\Omega'. \tag{3}$$

The phase space boundary Γ is divided into the inflow Γ^- and outflow Γ^+ boundaries, which are defined as:

$$\Gamma^-=\{(\boldsymbol{r},\boldsymbol{\Omega})\in\Gamma|\boldsymbol{\Omega}\cdot\mathbf{n}<0\},\ \Gamma^+=\{(\boldsymbol{r},\boldsymbol{\Omega})\in\Gamma|\boldsymbol{\Omega}\cdot\mathbf{n}\geqslant 0\}, \tag{4}$$

where $\boldsymbol{n}$ denotes the unit normal pointing outward at point $\boldsymbol{r}$ on the boundary. Due to the hyperbolic characteristics of the neutron-transport equation, the boundary condition is to be prescribed only at the inflow boundary Γ^-, which can be characterized as Eq. (5):

$$\phi(\boldsymbol{r},\boldsymbol{\Omega})=\mathrm{g}(\mathbf{r},\boldsymbol{\Omega}),(\mathbf{r},\boldsymbol{\Omega})\in\Gamma^-. \tag{5}$$

For the simplicity, we define the following inner product operator:

$$\begin{aligned}(f,g)_V &:= \textstyle\int_V\int_{\mathbb{S}^2}f(\boldsymbol{r},\boldsymbol{\Omega})\mathrm{g}(\boldsymbol{r},\boldsymbol{\Omega})\mathrm{d}\boldsymbol{\Omega}\mathrm{d}\boldsymbol{r}\\ \langle f,g\rangle_{\partial V} &:= \textstyle\int_{\partial V}\int_{\mathbb{S}^2}\boldsymbol{n}\cdot\boldsymbol{\Omega}f(\boldsymbol{r},\boldsymbol{\Omega})\mathrm{g}(\boldsymbol{r},\boldsymbol{\Omega})\mathrm{d}\boldsymbol{\Omega}\mathrm{d}\boldsymbol{r}\\ \langle f,g\rangle^-_{\partial\mathbf{V}} &:= \textstyle\int_{\partial\mathbf{V}}\int_{\boldsymbol{n}\cdot\boldsymbol{\Omega}<0}\boldsymbol{n}\cdot\boldsymbol{\Omega}f(\boldsymbol{r},\boldsymbol{\Omega})\mathrm{g}(\boldsymbol{r},\boldsymbol{\Omega})\mathrm{d}\boldsymbol{\Omega}\mathrm{d}\boldsymbol{r}\\ \langle f,g\rangle^+_{\partial\mathbf{V}} &:= \textstyle\int_{\partial\mathbf{V}}\int_{\boldsymbol{n}\cdot\boldsymbol{\Omega}\geqslant 0}\boldsymbol{n}\cdot\boldsymbol{\Omega}f(\boldsymbol{r},\boldsymbol{\Omega})\mathrm{g}(\boldsymbol{r},\boldsymbol{\Omega})\mathrm{d}\boldsymbol{\Omega}\mathrm{d}\boldsymbol{r}\end{aligned} \tag{6}$$

The non-physical oscillations occurs when the first-order transport equation is solved by the standard Galerkin finite-element method. In order to avoid this problem, the Algebraic Sub-Grid Scale (ASGS) finite-element method is used in NECP-FISH, which is optimized from A.G. Buchan's Sub-Grid Scale finite-element method (Buchan, 2010). The derivation of this method needs to start from the weak form of transport equation. And the more general form is obtained by the weighted residual method. The residual of equation (1) is defined as:

$$R(\phi)=\boldsymbol{L}\phi-s. \tag{7}$$

For the exact solution ϕ of the equation, the residual is always zero. Therefore, the residual still equals to zero when it is multiplied by the weight function w. With the integration by parts, the derivative acting on ϕ can be transferred to weight function. Meanwhile, the boundary integral term can also be obtained. Finally, the above equation becomes:

$$\langle w,\phi\rangle_{\partial V}+(\boldsymbol{L}^*w,\phi)_V=(w,s)_V. \tag{8}$$

where the adjoint operator is defined as Eq. (8):

$$\boldsymbol{L}^*:=-\boldsymbol{\Omega}\cdot\nabla+\boldsymbol{\Sigma}. \tag{9}$$

The boundary conditions are substituted into the boundary integral term. And the weak form of the transport equation is finally obtained:

$$\langle w,\phi\rangle^++\langle w,g\rangle^-+(\boldsymbol{L}^*w,\phi)=(w,s). \tag{10}$$

The main idea of ASGS finite-element method is to divide the unknown variable into two different parts as shown in Eq. (9):

$$\phi=\phi_c+\phi_r. \tag{11}$$

This first part ϕ_c represents the continuous part which is resolvable scale of ϕ and can be captured by the finite-element mesh. The second part ϕ_r represents the residual part which accounts for the unresolvable scales of ϕ. Generally, the residual part is not defined in the global scale, while it is defined in every element scale.

The global scale is approximated by continuous finite-element method, and the outer boundary conditions are only applied to the continuous part, hence the weak form can be characterized as Eq. (10):

$$\langle w,\phi_c\rangle^++\langle w,g\rangle^-+(\boldsymbol{L}^*w,\phi_c+\phi_r)=(w,s). \tag{12}$$

Meanwhile, the residual equation can be written as:

$$\boldsymbol{L}\phi_r=s_r \tag{13}$$

where, $s_r=s-\boldsymbol{L}\phi_c$.

The ASGS method uses algebraic equation to approximate the residual part of the flux:

$$\phi_r=\lambda(s-\boldsymbol{L}\phi_c), \tag{14}$$

where the parameter λ is the approximation of $\boldsymbol{L}^{-1}$, which is usually a constant or a diagonal matrix related to the element size and the cross section. The reason why the residual equation is approximated by the above is that Sub-Grid Scale method requires constant matrix inverse calculation when assembling the global algebraic equations, which is very time-consuming With the above approximation, the matrix is changed to a diagonal matrix. The value of parameter λ has a great influence on the stability of the code. As an approximation of $\boldsymbol{L}^{-1}$, its dimension should be cm. At the same time, combining the results of discrete extremum method and Fourier analysis, in this paper, we take the parameter λ as follow:

$$\lambda=\left(\frac{d+1}{h}+\Sigma_\mathrm{t}\right)^{-1} \tag{15}$$

Insert equation (14) to equation (12), the weak form of this method can be characterized as Eq. (16):

$$\langle\varphi,\phi_c\rangle+(\boldsymbol{L}^*\varphi,\phi_c)-(\boldsymbol{L}^*\varphi,\lambda\boldsymbol{L}\phi_c)=(\varphi,s)-(\boldsymbol{L}^*\varphi,\lambda s). \tag{16}$$

The angular flux can be expanded by the basis function as shown in Eq. (14):

$$\phi(\mathbf{r},\boldsymbol{\Omega})=\boldsymbol{\varphi}^\mathrm{T}(\mathbf{r},\boldsymbol{\Omega})\boldsymbol{\phi}, \tag{17}$$

where the basis function $\boldsymbol{\varphi}(\boldsymbol{r},\boldsymbol{\Omega})$ can be written as the Kronecker product of the most frequently used piecewise polynomial function $\mathbf{N}(\boldsymbol{r})$ of the finite-element method in spatial and the spherical harmonic function $\boldsymbol{Y}(\boldsymbol{\Omega})$ in angular space.

$$\boldsymbol{\varphi}(\mathbf{r},\boldsymbol{\Omega}) = \mathbf{N}(\mathbf{r}) \otimes \mathbf{Y}(\boldsymbol{\Omega}). \tag{18}$$

The characteristic of the piecewise polynomial $\mathbf{N}(\boldsymbol{r})$ is the value is 1 at a specific node, while others are 0 for the element. The gradient of the spatial basis function and the integral over the element can be derived analytically. The $\boldsymbol{Y}(\boldsymbol{\Omega})$ is a vector which is obtained from rearranging the initial spherical harmonic function $Y_{n,m}(\boldsymbol{\Omega})$. The $Y_{n,m}(\boldsymbol{\Omega})$ is defined as the product of normalization coefficient $a_{n,m}$ and function $Y'_{n,m}(\boldsymbol{\Omega})$. Their expressions are showed as follows:

$$a_{n,m} = \sqrt{\frac{2n+1}{2\pi(1+\delta_{0m})}\frac{(n-|m|)!}{(n+|m|)!}}, \tag{19}$$

$$Y'_{n,m}(\boldsymbol{\Omega}) = \begin{cases} P_n^m(\cos\theta)\cos(m\varphi), m \geqslant 0 \\ P_n^{|m|}(\cos\theta)\sin(|m|\varphi), m < 0 \end{cases}. \tag{20}$$

In the end, the expression of the discretization form can be characterized as Eq. (21):

$$\begin{aligned} &[<\boldsymbol{\varphi},\boldsymbol{\varphi}>^{+} - (\boldsymbol{\Omega}\cdot\nabla\boldsymbol{\varphi},\boldsymbol{\varphi}) + (\boldsymbol{\varphi},\Sigma\boldsymbol{\varphi})]\boldsymbol{\phi}+ \\ &\begin{bmatrix} (\lambda\boldsymbol{\Omega}\cdot\nabla\boldsymbol{\varphi},\boldsymbol{\Omega}\cdot\nabla\boldsymbol{\varphi}) + (\lambda\boldsymbol{\Omega}\cdot\nabla\boldsymbol{\varphi},\Sigma\boldsymbol{\varphi}) \\ \text{-}\ (\lambda\Sigma\boldsymbol{\varphi},\boldsymbol{\Omega}\cdot\nabla\boldsymbol{\varphi})\ \text{-}\ (\lambda\Sigma\boldsymbol{\varphi},\Sigma\boldsymbol{\varphi}) \end{bmatrix}\boldsymbol{\phi} \\ &= [(\boldsymbol{\varphi},\boldsymbol{\varphi}) + (\lambda\boldsymbol{\Omega}\cdot\nabla\boldsymbol{\varphi},\boldsymbol{\varphi})\ \text{-}\ (\lambda\Sigma\boldsymbol{\varphi},\boldsymbol{\varphi})]\boldsymbol{s} - <\boldsymbol{\varphi},\boldsymbol{\varphi}>^{\text{-}}\boldsymbol{g} \end{aligned} \tag{21}$$

The spherical harmonic function method uses Riemann decomposition to deal with boundary conditions (Buchan et al., 2011). For the boundary angular matrix:

$$\boldsymbol{A}_n = \int_{\Omega} \boldsymbol{n}\cdot\boldsymbol{\Omega}\boldsymbol{Y}(\boldsymbol{\Omega})\mathbf{Y}^{\mathrm{T}}(\boldsymbol{\Omega})\mathrm{d}\boldsymbol{\Omega} \tag{22}$$

It is decomposed into two parts, the outgoing part $\boldsymbol{A}_n^{+}$ with positive eigenvalues and the incoming part $\boldsymbol{A}_n^{-}$ with negative eigenvalues. The $\boldsymbol{A}_n^{-}$ is zero with the vacuum boundary condition. In each element and boundary, all the items in the Eq. (16) are obtained and assembled into the global stiffness matrix. The way of assembly is element-by element. Finally, the algebraic equations with global stiffness matrix and vector are formed:

$$\boldsymbol{K}\boldsymbol{\phi} = \boldsymbol{F} \tag{23}$$

The neutron flux can then be obtained by solving the equations with GMRES algorithm.

3. Framework and code development

3.1. Overall framework of NECP-FISH

The overall framework of NECP-FISH is as shown in Fig. 1. As the geometry structure of the fusion reactor is complicated and the geometry-modeling process using finite-element method is not convenient manually, the SALOME platform has been applied in NECP-FISH. The SALOME platform integrates many open-source programs, such as Gmsh and ParaView, to provide pre-processing and post-processing functions for the numerical simulation. Especially noted here, the powerful functions including the three-dimension CAD modeling, geometry-mesh generation and numerical-result visualization are integrated in the SALOME platform. Therefore, based on the imported CAD file of the fusion-reactor blanket, the SALOME platform can establish the analysis model and hence generate the meshes for geometry automatically. Through solving the first-order neutron-transport equation for the fusion reactor, the numerical results are provided to the SALOME platform for the visualization display. Through the SALOME platform, it is convenient and automatic to simulate the fusion-reactor blanket using the NECP-FISH code.

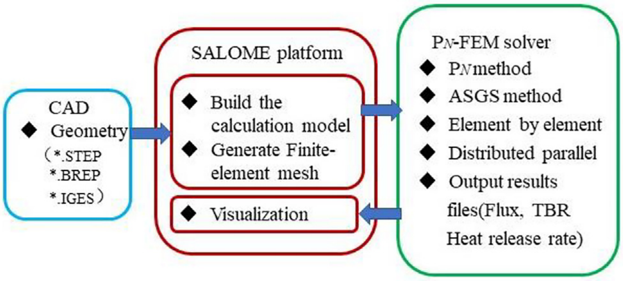


Fig. 1. Overall framework of NECP-FISH.

3.2. Code development of NECP-FISH

From the above introduction we know that there are two main parts to develop this program, one is the finite-element solver, and the other is the interface with SALOME platform. The finite-element solver is developed by the Fortran language using the unstructured mesh, which has the capability for two-dimensional triangle mesh and three-dimensional tetrahedron mesh. The multigroup cross section with MATXS format has been applied in NECP-FISH to solve the neutron-transport equation. For the external-source definition, both the volume source and mesh source can be supported in NECP-FISH. As there are large amount of data required for the neutronics modeling and simulation for the fusion-reactor blanket, the HDF5-format file has been applied in NECP-FISH as the input file, with which the geometry mesh, material component, external-source data and boundary condition can be defined appropriately. Compared with the ordinary document input, the HDF5-format file requires smaller storage space. What's more important, the HDF5 library implements a high-level application programming interface with the Fortran language, with which the code can be read and written easily.

As the number of unstructured mesh increases, the scale of computation increases rapidly. Therefore, the parallel technique has been designed and developed in NECP-FISH to reduce the computation time. The implementation of parallel is based on the Message Passing Interface(MPI). The overall idea is to decompose the spatial finite-element mesh into different parts to the specific CPUs and solve the response matrix of the neutron-transport equation on each single CPU. The overall parallel technique scheme is as shown in Fig. 2. The assembly of response matrix and the communication between different CPUs are realized by PSBLAS. The PSBLAS is an open-source library, which is designed to handle the implementation of iterative solvers for sparse linear systems on distributed memory parallel computers. At present, we can calculate the problem with hundreds of CPUs.

As mentioned above, it is necessary to realize the data exchange between the SALOME platform and the finite-element solver. Therefore, the interface module has been designed and developed in the platform to generate the input card of the finite-element solver. The interface module is developed by the Python language. Moreover, in order to increase the user friendliness, we develop a simple user graphic interface with Qt library, which can generate input card visually. Visual file used in this code is the XDMF format which is generated by XH5For. The XH5For is a library to read and write parallel partitioned finite-element meshes taking advantage of the collective or independent input/output provided by the HDF5 library. The NECP-FISH code calls XH5For library to generate the visual file, which can display the neutron-flux distribution and finite-element mesh on different processes.

4. Numerical results and analysis

As the verification and preliminary application, we apply the NECP-FISH code to simulate the Breeder Unit (BU) of

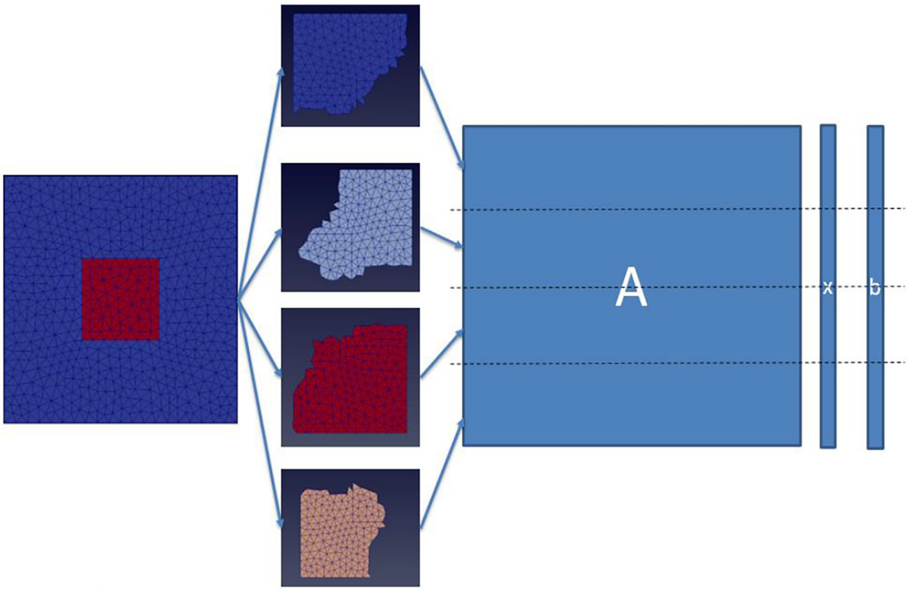


Fig. 2. Parallel technique scheme of NECP-FISH.

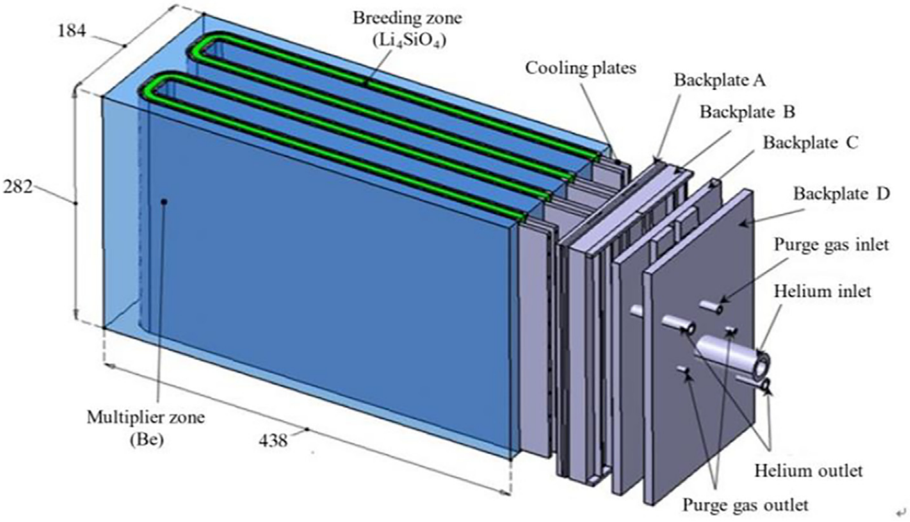


Fig. 3. The detail view of the breeding unit.

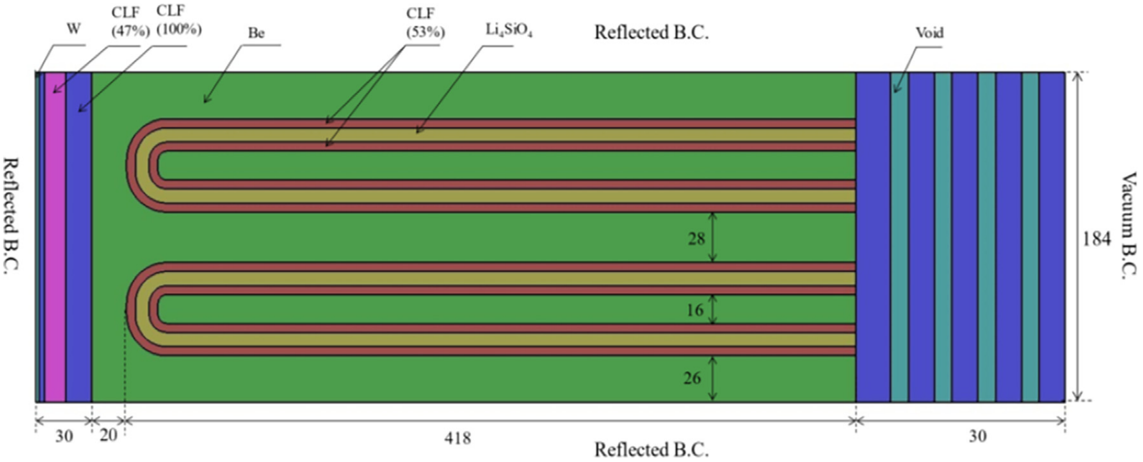


Fig. 4. The materials layout and boundary conditions.

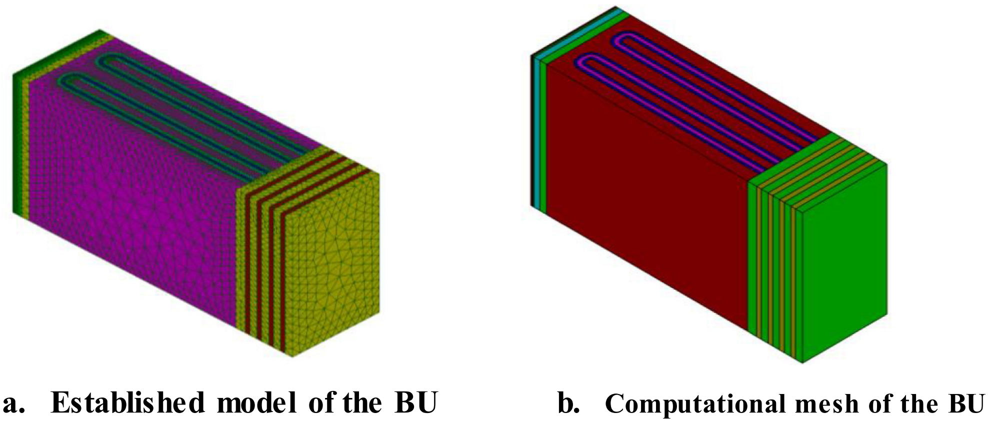

a. Established model of the BU
b. Computational mesh of the BU

Fig. 5. The pre-processing result of the geometry model.

Table 1
Sensitivity-analysis results towards number of finite-element meshes.

Number of the mesh	TBR
41,817	1.277
80,375	1.297
196,335	1.313
400,245	1.323

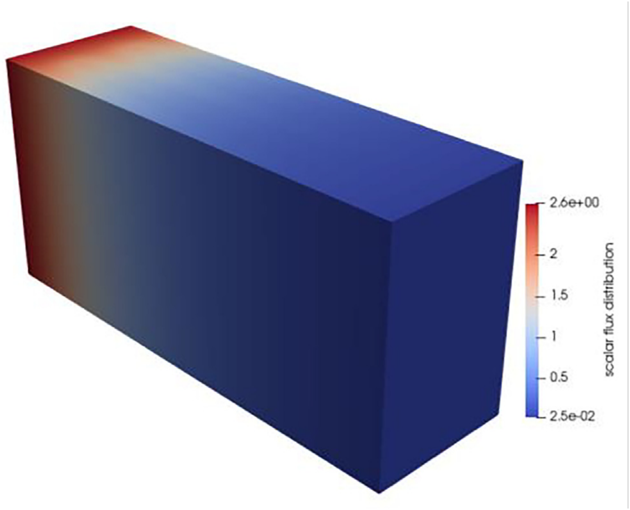


Fig. 6. Visualized total neutron flux of the BU.

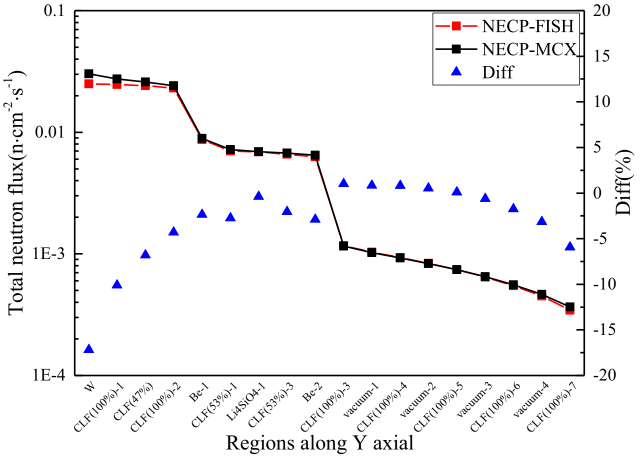


Fig. 7. Comparison of total neutron flux averaged in regions along Y axial.

Helium-Cooling Ceramic Breeder (HCCB) for the fusion DEMO, which was designed by Southwestern Institute of Physics. The reference results are provided by our home-developed Monte-Carlo code named NECP-MCX (Zheng et al., 2021), which is the hybrid Monte-Carlo-Deterministic particle-transport code with high efficiency.

The detail view of the breeding unit is as shown in Fig. 3, and the material layout and boundary condition as shown in Fig. 4. This design applied the Li_4SiO_4 pebble as the tritium-breeder material and the beryllium pebble as the neutron-multiplier material. These materials are separated with the U-shape cooling panels, which are connected with the back plates. The Reduced Activation Ferritic/Martensitic (RAFM) steel CLF-1 is applied as the structure material. The First Wall (FW) is protected by the tungsten armor with thickness of 2 mm. In order to complement the modeling and simulation, we define an isotropic volume source in front of the tungsten armor, with strength of 1 $cm^{-3}s^{-1}$. The energy of source neutron is 14.1 MeV and the thickness of volume source is 1 mm.

Fig. 5 shows the three-dimension model and geometry mesh generated by the NECP-FISH code. There are totally 80,375 tetrahedral elements having been generated and applied for simulation by NECP-FISH. For the neutronics simulation, the Vitamin-J 175-group library based on FENDL-2.1 (2004) has been generated and applied in NECP-FISH. For the expansion of the scattering source, the angular spherical harmonic function P_4 and P_3 have been applied. For the reference simulation by NECP-MCX, the pointwise cross-section library based on FENDL-2.1 has been generated and applied.

For NECP-FISH, the division of finite-element meshes would impact the simulation results. Therefore, the sensitivity analysis towards the number of finite-element meshes has been performed. The results are as shown in Table 1. It should be noted that the TBR is calculated as the tritium-atom produced on the BU by the normalized fusion neutron in our calculation and verification. The results indicate that with increasing of the mesh number, the simulated TBR values are getting closer to the converged value. However, more number of finite-element meshes means more computational cost. As for the balance between accuracy and efficiency, the model with 80,375 finite-element meshes has been applied for the further verification.

Table 2
Comparison of heat release rate in different regions.

Region	Reference/Mev	NECP-FISH/Mev	Diff./%	Region	Reference/Mev	NECP-FISH/Mev	Diff./%
W	2.625×10^{-2}	1.806×10^{-2}	-3.118×10^{1}	**CLF(53%)-3**	4.681×10^{-2}	4.832×10^{-2}	3.228×10^{0}
CLF(100%)-1	1.350×10^{-1}	1.178×10^{-1}	-1.276×10^{1}	**CLF(53%)-4**	4.682×10^{-2}	4.832×10^{-2}	3.185×10^{0}
CLF(47%)	1.964×10^{-1}	1.954×10^{-1}	-5.515×10^{-1}	**Be-2**	3.315×10^{-1}	3.334×10^{-1}	5.546×10^{-1}
CLF(100%)-2	3.604×10^{-1}	4.007×10^{-1}	1.118×10^{1}	**Be-3**	3.316×10^{-1}	3.334×10^{-1}	$5.468 \times 10\text{-}^{1}$
Be-1	2.845×10^{0}	2.960×10^{0}	4.047×10^{0}	**CLF(100%)-3**	9.467×10^{-3}	9.935×10^{-3}	4.944×10^{0}
CLF(53%)-1	5.579×10^{-2}	5.770×10^{-2}	3.429×10^{0}	**CLF(100%)-4**	5.332×10^{-3}	5.630×10^{-3}	5.592×10^{0}
CLF(53%)-2	5.580×10^{-2}	5.818×10^{-2}	4.225×10^{0}	**CLF(100%)-5**	4.180×10^{-3}	4.415×10^{-3}	5.619×10^{0}
Li4SiO4-1	3.506×10^{0}	3.440×10^{0}	-1.874×10^{0}	**CLF(100%)-6**	3.232×10^{-3}	3.395×10^{-3}	5.031×10^{0}
Li4SiO4-2	3.506×10^{0}	3.425×10^{0}	-2.310×10^{0}	**CLF(100%)-7**	2.393×10^{-3}	2.490×10^{-3}	4.050×10^{0}

The parallel calculation has been applied for simulations of NECP-FISH and NECP-MCX on the same computational platform. For NECP-FISH, 48 different processors have been used and the simulation cost 480 s in total. It took 32 iterations for NECP-FISH to get the converged simulation results. For NECP-MCX, there are 10^{8} particles having been simulated, for which 72 different processors cost 1370 s in total. The relative statistical errors for the neutron flux are all below 1% for NECP-MCX. Therefore, much less computation time is required by NECP-FISH than the NECP-MCX code. The TBR of the BU calculated by NECP-FISH is 1.297 and corresponding reference value given by NECP-MCX is 1.334, which has the difference −0.037 and agreed well with each other. The three-dimensional visualization for the total neutron flux by NECP-FISH is as shown in Fig. 6. From the numerical results, it can be observed that the total neutron flux decreases with the distance from the volume source in front of the tungsten armor. This numerical-result phenomenon is agreed well with the theoretical expectation.

For the verification, the total neutron flux has been integrated into the one-dimension values along the Y-axial regions. The averaged total neutron flux is compared as shown in Fig. 7. Through the result comparisons, it can be observed that the almost all the differences are within 10%, except for the first several regions near the source. The heat release rate of different regions calculated by NECP-FISH is also compared with corresponding reference by NECP-MCX, as shown in Table 2. Through the comparison, we can find that the relative differences for the heat release rate between NECP-FISH and NECP-MCX are large near the source regions. The differences of heat release rate have the same trend with the averaged total neutron flux. And the differences are mainly caused by the differences existed in the cross sections applied by NECP-FISH and NECP-MCX.

5. Conclusions

The deterministic code named NECP-FISH has been designed and developed for the neutronics analysis of fusion-reactor blanket. The finite-element method has been applied in NECP-FISH to treat the complicated and irregular structures in the fusion-reactor blanket. Moreover, the open-source platform SALOME has been integrated into NECP-FISH for the geometry-model pre-process and simulation-result post-process. The preliminary application of NECP-FISH is used to model and simulate the Breeder Unit of HCCB-DEMO, with the reference results provided by the Monte-Carlo code NECP-MCX.

The result of TBR is in good agreement with that of the Monte-Carlo code. However, the computational time is shortened by nearly 10 times for NECP-FISH than that of NECP-MCX. The discrepancies of the heat release rate and neutron flux with reference are small in most regions. The large discrepancy may be caused by difference in nuclear data libraries.

In the further research works, we intend to optimize the NECP-FISH code and apply it to model and simulate the 22.5° sector of the fusion reactor.

CRediT authorship contribution statement

Jianxin Miao: Methodology, Software, Validation, Writing – original draft. **Chao Fang:** Methodology, Software, Validation, Writing – original draft. **Chenghui Wan:** Conceptualization, Methodology, Resources, Writing – review & editing. **Wanchao Mao:** . **Hongyun Xie:** .

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is financially supported by National Key R&D Program of China (Grant No. 2017YFE0302200).

References

Große, D., Fischer, U., Kondo, K., Leichtle, D., Pereslavtsev, P., Serikov, A., 2013. Status of the McCad geometry conversion tool and related visualization capabilities for 3D fusion neutronics calculations. Fusion Eng. Des. 88 (9-10), 2210–2214.

Wu, Y., 2009. CAD-based interface programs for fusion neutron transport simulation. Fusion Eng. Des. 84 (7-11), 1987–1992.

W. A. Rhoades, D. B. Simpson, "The TORT Three-Dimensional Discrete Ordinates Neutron/Photon Transport Code (TORT Version 3)," Oak Ridge National Laboratory (1997).

J. M. McGhee, T. A. Wareing, D. A. Barnett Jr, "Attila User's Manual," Transpire, Inc (2007).

Zhang, G.-C., Liu, J., Cao, L.-Z., Wu, H.-C., Yuan, X.-B., 2020. Neutronic calculations of the China dual-functional lithium-lead test blanket module with the parallel discrete ordinates code Hydra. Nucl Sci. Tech. 31 (8). https://doi.org/10.1007/s41365-020-00789-4.

Kim, J.W. et al., 2014. Development of discrete ordinates code supporting unstructured tetrahedral mesh and applied in neutronics analysis for the Korea Helium Cooled Ceramic Reflrctor Test Blanket Module. Fusion Eng. Des. 89, 1172–1176.

"SALOME: The open-source integration platform for numerical simulation." http://www.salome-platform.org (2005).

Feng, K.M. et al., 2009. Conceptual design study of fusion DEMO plant at SWIP. Fusion Eng. Des. 84, 2109–2113.

L. Z. Cao et al, "Numerical methods for deterministic neutron transport with complex geometry: Chapter 6: P_N-FEM method," *Woodhead Publishing Series in Energy*, 2021, Pages:109-165

Cao, L.Z. et al., 2020. A unified form of stabilized finite element methods for solving the fisrt-oedre neutron transport equation. Tracs of the American Nuclear Society 122, 731–733.

Fang, C., Wu, H., Cao, L., Li, Y., Miao, J., 2020. Modified least-squares finite element method for solving the Boltzmann transport equation with spherical harmonic angular approximation. Ann. Nucl. Energy 140, 107126. https://doi.org/10.1016/j.anucene.2019.107126.

Buchan, A.G. et al., 2010. The Inner-Element Subgrid Scale Finite Element Method for the Boltzmann Transport Equation. Nucl. Sci. Eng. 164, 105–121.

Buchan, A.G., Merton, S.R., Pain, C.C., Smedley-Stevenson, R.P., 2011. Riemann boundary conditions for the Boltzmann transport equation using arbitrary angular approximations. Ann. Nucl. Energy 38 (5), 1186–1195.

Zheng, Q.i., Shen, W., He, Q., Li, J., Cao, L., Wu, H., 2021. High-efficiency simulation of VENUS-3 neutron-shielding problem with an automatic and enhanced hybrid Monte-Carlo-Deterministic method. Ann. Nucl. Energy 153, 108039. https://doi.org/10.1016/j.anucene.2020.108039.

D. L. Aldama, A. Trkov, ''FENDL-2.1: evaluated nuclear data library for fusion applications,'' INDC(NDS)-467, FENDL (2004).