Development of a New Thermal-Hydraulic Module for FRENETIC, a Code for the Multiphysics Analysis of Liquid Metal–Cooled Reactors

Development of a new Thermal-Hydraulic Module for FRENETIC, a Code for the Multiphysics Analysis of Liquid Metal-Cooled Reactors

Antonino Lombardo,[a] Giuseppe F. Nallo,[*,a] Nicolò Abrate,[a] Yapeng Liu,[a,b] and Sandra Dulla[a]

[a] *NEMO Group - Energy Department, Politecnico di Torino*
Corso Duca Degli Abruzzi 24 - 10129, Turin, Italy

[b] *State Key Laboratory of Multiphase Flow in Power Engineering,*
Shaanxi Key Lab. of Advanced Nuclear Energy and Technology,
School of Nuclear Science and Technology, Xi'an Jiaotong University, Xi'an 710049, China

[*]Email: giuseppefrancesco.nallo@polito.it

Number of pages: 36
Number of tables: 1
Number of figures: 12

Abstract

The present paper describes the development of a new thermal-hydraulic (TH) module for FRENETIC, a multiphysics code for the full-core simulation of liquid metal-cooled fast reactors, developed at Politecnico di Torino. The code performs steady state and transient neutronic (NE) and TH coupled calculations, while maintaining a relatively low computational cost thanks to the adoption of simplified physical models. The NE module implements the nodal formulation of the multi-group neutron diffusion equations with delayed neutron precursors, whereas the TH module treats the reactor hexagonal assemblies as separate channels, which are individually modelled as 1D in the axial direction, accounting for the thermal coupling in the horizontal direction. The new TH module is more robust and portable while providing improved performance with respect to the previous implementation, also thanks to the adopted OpenMP parallelization. Some physical aspects that were previously neglected, such as the thermal inertia of non-fuel rods, have also been included in the model. The development was carried out in accordance with current best practices for code design, implementation and testing, thus rendering the code easier to be maintained and possibly extended in the future. The code usability has also been improved by means of a set of Python classes purposely developed to simplify the input generation and post-processing phases. This can potentially widen the usage of FRENETIC within the fast reactor community for the simulation of full-core coupled NE-TH transients and/or as a platform to test new solution methods. The paper also includes the application of this new FRENETIC version to a representative configuration of the Advanced Lead-cooled Fast Reactor European Demonstrator (ALFRED) core design. First, the nominal operating conditions of the ALFRED reactor were simulated, proving that the new upgrade of the code is faster and more robust with respect to the previous one. Then, two accidental transients where NE-TH feedback effects are relevant - an Unprotected Loss of Flow and a Total Instantaneous Blockage of a single fuel assembly - were considered.

Keywords — liquid metal-cooled fast reactors, thermal-hydraulics, code development, multiphysics simulation, loss of flow accident

I. INTRODUCTION

Heavy Liquid Metal-Cooled Reactors (HLMCRs) with fast neutron spectrum are among the proposed Generation IV reactor concepts [1]. In view of the industrial interest towards these systems, HLMCR core design activities are currently being pursued worldwide. Multiphysics computational tools that are simultaneously reliable and computationally efficient are required both in support of the core design phase and to assess the reactor behaviour during operational and off-operational transients. The specific features of HLMCRs, namely the fast neutron spectrum and the liquid metal coolant, determine a different behaviour with respect to commercial light water reactors, thus calling for the development of specific codes.

In this framework, the FRENETIC code was developed at Politecnico di Torino with the aim of performing full-core simulations of liquid metal-cooled fast reactors. The code is capable of performing steady state and transient neutronic (NE) and thermal-hydraulic (TH) coupled calculations, while maintaining a relatively low computational cost thanks to the adoption of simplified physical models. The development of this tool has included a number of validation activities (e.g. [2, 3]), as well as of benchmarks with other numerical tools (e.g. [4, 5]), which have confirmed the modelling capabilities of FRENETIC and led to the identification of some necessary further developments to be included into the code, such as a module for the simulation of the photon gamma heat and decay heat deposition [6].

In the present paper, the design and development of a new TH module for FRENETIC, carried out in accordance to modern software development practices, is presented. This activity aimed at improving the performance and maintainability of the code while retaining the previously selected simplified physical models. These developments led to considerable performance improvement in steady state and to improved performance in transient calculations. In the final part of the paper, a couple of relevant test cases are presented, where the code is applied to simulate the ALFRED reactor core in steady state operation and in accidental transients.

The paper is organized as follows. The FRENETIC code NE and TH modules are first described, together with the coupling methodology. The governing equations and numerical methods adopted for the development of the new TH module are then discussed. Finally, the application of FRENETIC to the steady state and transient simulation of the ALFRED core is presented.

II. THE FRENETIC CODE

II.A. Code overview

The FRENETIC [7] code aims at the efficient multiphysics (NE and TH) simulation of the steady state and transient behavior of liquid metal-cooled cores arranged in hexagonal closed assemblies. The most common coolant types (lead, lead-bismuth eutectic and sodium) can be simulated. The required computational time is reduced by resorting to simplified models, namely:

- a multi-group diffusion model for neutrons, which is spatially discretised with a coarse mesh nodal method at the assembly level;
- a 1D advection/diffusion model for the flow and heat transfer of liquid metal within each closed assembly, taking into account the thermal coupling between neighbouring assemblies.

This strategy allows FRENETIC to solve the full-core NE-TH problem, providing axial distributions of the fuel and coolant temperatures - averaged over the cross section of each Hexagonal Assembly (HA) - that are consistent with the fission power generation, evaluated by means of the coarse mesh 3D diffusion approach. The spatial detail of such a calculation is much lower than the one which could be achieved by, e.g., coupled CFD - Monte Carlo or sub-channel - S_N codes, but it is sufficiently accurate to provide a reliable yet computationally efficient estimate of the 3D temperature distribution in the core, as required for design and safety calculations.

The code is written in Fortran, following the Fortran-90 standard, and supports cross-platform compilation. Recently, the code has been equipped with a Python binding to make its execution easier and more flexible. The combination of this binding with https://github.com/nicoloabrate/COREutils, a set of open-source, external Python classes for the FRENETIC input generation, allows to easily prepare, execute and post-process several simulations in parallel with a few lines of Python code.

II.B. Multiphysics coupling

The computed temperatures are used in order to evaluate the new multi-group cross sections and diffusion coefficients for the NE module. The temperature dependence of these data for a given homogenised material region is obtained through a bi-variate linear interpolation with respect to

the fuel and the coolant temperatures, averaged inside the computational volume of interest [4]:

$$\Sigma_x(T_c, T_f) = \Sigma_x(T_{c,0}, T_{f,0}) + \left(\frac{\partial \Sigma_x}{\partial T_f}\right)_{T_c} (T_f - T_{f,0}) + \left(\frac{\partial \Sigma_x}{\partial T_c}\right)_{T_f} (T_c - T_{c,0}) \tag{1}$$

The multi-group constants are generated by means of the Monte Carlo code Serpent 2 [8], which is used to collapse the continuous energy data into six groups and to homogenize them over the reactor heterogeneous regions ensuring that the reaction rates for each region are preserved [4]. The six-group energy grid employed in this paper is the same used in previous works [4, 5].

For steady-state calculations, the NE and TH modules are coupled via under-relaxed, fixed-point iterations up to convergence. For transient calculations, instead, the two modules run independently, each with its characteristic time step. The two modules exchange information according to a specified coupling time step: the TH module passes the fuel and coolant averaged temperatures to the NE module, which provides in turn the updated power deposition distribution. The validation and benchmark activities performed in [3] and [4] carried out in the past have confirmed the suitability of the described approach, despite its explicit nature.

II.C. Calculation domain

The FRENETIC calculation domain extends axially from the beginning to the end of the constant flow area region of the HAs inside the core, and radially over the whole core and beyond, possibly including the liquid metal outside the core barrel. Indeed, it was shown in [4] that the NE domain must include that external region to ensure a sufficiently accurate calculation of the radial power deposition profile.

The coolant inside the Inter-Wrapper (IW) region is treated as stagnant, for the sake of simplicity. This is an acceptable approximation since, in nominal conditions, inter-assembly heat transfer can be regarded as a second-order effect.

III. DEVELOPMENT OF THE NEW TH MODULE

III.A. Required spatial resolution and selected modelling approach

As mentioned above, the TH calculation provides the axial distribution of the HA-averaged pressure, density, velocity, coolant and fuel temperatures, accounting for the specific TH charac-

teristics of different HAs.

Since the thermal coupling between neighbouring HAs is relatively slow compared to the coolant transit along the HA, the 3D problem can instead be treated as a series of weakly coupled 1D problems thanks to the expected time scales of the TH phenomena involved. The validity of this approach was already discussed in a previous work [7]. The TH problem at the HA level is closed adopting literature correlations for friction factors and heat transfer coefficients.

In order to possibly improve the current implementation, a dynamic model that resolves additional axial equations for dummy and control rods was added. This allows to account for the heat capacity of non-fuel rods during transients. The possibility to compute the temperature evolution of non-heated rods can possibly smooth the nonphysical coolant temperature peaks that were found in some cases using the previous version of the code, e.g. in the computation of the EBR-II core [3].

III.B. Spatial discretisation

The spatial discretisation is performed by means of a finite volume approach. Each HA is axially subdivided into control volumes with an arbitrary distribution, thereby allowing a flexible mesh refinement - typically employed to increase the spatial resolution in the active region. The mesh is staggered, i.e. the temperatures are evaluated at the cell centers, whereas the velocities are evaluated at cell faces, to avoid checkerboard-type instabilities [9].

III.C. Boundary conditions and source terms

The user can provide time-dependent boundary conditions for each HA in terms of inlet temperature, mass flow rate and inlet pressure. It is also possible to impose the pressure drop across the core, consequently computing the mass flow rate within each channel. In that case, the user provides an assumption on the initial mass flow rates, which represents an initial guess for the iterative calculation, constrained by the condition of having an equal pressure drop among all channels, consistent with the physical situation of parallel channels connected to the same lower and upper plena. The code is provided in this case with under-relaxation factors to ensure convergence during iterations.

As far as source terms are concerned, the code reads an externally provided heat generation

varying in space and time as a result of neutron flux distribution, which can either be user-defined or self-consistently computed via the NE module.

The code was also recently equipped with a preCICE-based coupling library [10], which reduces the computational overhead of the file-based exchange between the NE and TH modules by enabling the memory-based data exchange. Moreover, this new feature could allow, in the future, to receive self-consistent boundary conditions from system-level codes such as, e.g., ATHLET [11].

III.D. Code design approach

The methodology adopted for the development of the new TH module was inspired by the Software Quality Management System in place within the SICNUC division of ENEA (Italian National Agency for New Technologies, Energy and Sustainable Economic Development), incorporating guidelines from standards such as ISO/IEC/IEEE 12207 [12]. After an initial planning phase, a software requirements specification document was produced. A software design compliant with the specified requirements was then developed, discussed and approved before starting the code development phase, which includes testing and verification of the code compliance with the specified requirements.

The software design phase was carried out based on current best practices [13]. First, *data modelling* was carried out, which consisted in identifying the needed input and output quantities, based on the specified requirements and on the specific needs of the selected problem-solving approach, and in identifying the physical quantities to be treated by the code, again based on the selected problem-solving approach and including user inputs to modify the code behavior as needed. This led to the definition of a number of data structures. Then, *functional modelling* was carried out, identifying the transformations undergone by data contained in the data structures defined in the foregoing step. The functional modelling was performed with an increasing level of detail, up to the point where the identified functions were highly consistent, relying on graphical representations in the form of *data flow diagrams*. Finally, the detailed code design was carried out using *pseudocodes*, which were discussed and approved before moving to the actual Fortran-90 implementation. The top-down approach adopted in this phase aimed at promoting the code modularity. Techniques such as data encapsulation and dynamic allocation of variables, alongside with structured data management, were systematically adopted. This resulted in an ordered code

structure, as clearly conveyed by the function call graph shown in figure 1, generated via Doxygen https://www.doxygen.nl.

IV. GOVERNING EQUATIONS AND SPATIAL DISCRETISATION

IV.A. General assumptions

Mass, momentum and energy conservation equations are solved for the coolant within each HA. The control volume is fixed, with no mass flow or momentum exchange in the transverse direction. Both a steady state and a transient solver were implemented. The channel is assumed to be vertical, therefore gravity only contributes as a body force in the axial momentum equation.

The spatial discretisation will be presented according to the nomenclature in figure 2, using the standard finite volume method [9].

IV.B. Governing equations

IV.B.1. Conservation of mass

The equation for conservation of mass is generally expressed as:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot [\rho v] = 0 \tag{2}$$

Specializing it for one-dimensional flow and integrating over the volume for a generic nodal position C:

$$\frac{d\rho}{dt} + \rho \frac{dU}{dz} = 0 \tag{3}$$

$$\int \frac{d\rho}{dt} dV + \int \rho \frac{dU}{dz} dV = 0 \tag{4}$$

Due to the incompressibility of the flow, the first term always vanishes, while the second can be expressed as a function of the mass flow at volume's faces, thus yielding simply:

$$\dot{m}_e - \dot{m}_w = 0 \tag{5}$$

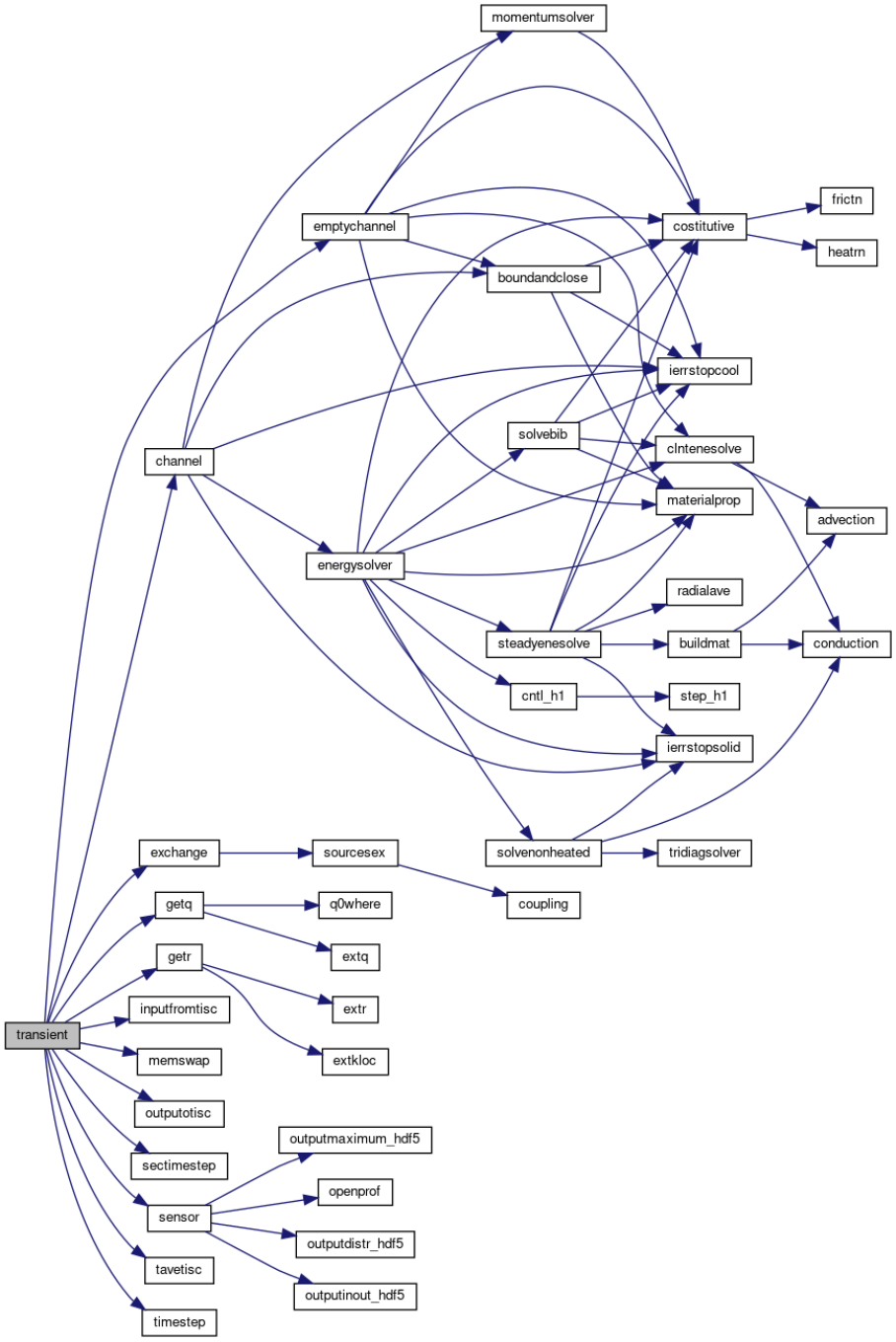

Fig. 1. *Example of function call graph for the driver subroutine handling transient calculations.*

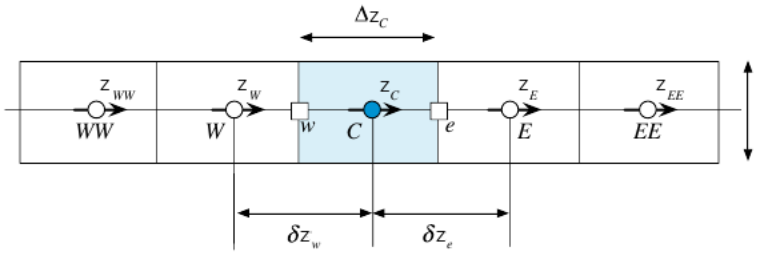

Fig. 2. *Nomenclature for spatial discretisation, reproduced from [9].*

The equation is written taking advantage of the mass flow conservation, and so rewritten in terms of mass flow rate, instead of velocities.

IV.B.2. Conservation of momentum

The same kind of procedure is carried out for the conservation of momentum in the axial direction, which is generally expressed as:

$$\frac{\partial[\rho v]}{\partial t} + \nabla \cdot [\rho v v] = -\nabla p + \mu \nabla^2 v + f_b \tag{6}$$

where f_b represents a dissipation term. Adopting the same procedure as above:

$$\frac{d\dot{m}_C}{dt} + \frac{\dot{m}_C^2}{\Delta z_C}\left(\frac{1}{\rho_e A_e} - \frac{1}{\rho_w A_w}\right) = -\frac{\Delta p}{\Delta z_C} A_{av} - \rho_C g A_{av} - \frac{1}{2 D_h A_{av} \rho_C} f_C \dot{m}_C^2 - K_{loc} \frac{\dot{m}_C^2}{2 \rho_C A_{av} \Delta z_C} \tag{7}$$

The dissipation term is written as a function of the Darcy–Weisbach friction factor f_C, consistently with the selected approach. The presence of gravity as a body force is taken into account, as well as the possible presence of localized pressure drops, which can be characterized by providing in input the desired generic loss coefficient K_{loc}. This allows, e.g., to consider the presence of spacer grids in case of non wire-spaced assemblies. The pressure drop term is expressed in terms of adjacent nodal values with a finite difference approach. The cross sectional areas at cell faces, A_e and A_w, as well as the average value A_{av}, are explicitly retained in the equations for

the sake of generality, to possibly allow cases with axially variable cross sections to be considered, e.g., in case of HA bowing. The density at cell faces is computed in terms of center values of neighboring cells, with weighting factors to enable axially varying spatial discretisation:

$$\rho_e = g_C \rho_C + (1 - g_C)\rho_E = \frac{\Delta z_C}{\Delta z_C + \Delta z_E}\rho_C + \frac{\Delta z_E}{\Delta z_C + \Delta z_E}\rho_E \tag{8}$$

IV.B.3. Conservation of energy for the coolant

A general form of the energy conservation equation expressed in terms of temperature is considered [9]:

$$\frac{\partial(\rho c_p T)}{\partial t} + \nabla[\rho c_p v T] = \nabla[k \nabla T] + \dot{q}_v + \rho T \frac{D c_p}{D t} - \left(\frac{\partial(\mathrm{Ln}\rho)}{\partial(\mathrm{Ln}T)}\right)_p \frac{Dp}{Dt} + \lambda\psi + \mu\Phi \tag{9}$$

This generic equation, which takes care of the pressure and temperature dependence of the specific enthalpy in its derivation, can be specialized for the present purpose, neglecting some terms and taking advantage of the coolant incompressibility. Specifically, the following assumptions are made:

- The viscous dissipation term Φ is negligible with respect to the heat generated in the fuel;
- The continuity equation for incompressible flow implies that $\psi = 0$;
- Since the density is constant inside the volume, also the pressure term is negligible.

The resulting adopted formulation is:

$$\frac{\partial(\rho c_p T)}{\partial t} + \nabla[\rho c_p v T] - \nabla[k \nabla T] - \rho T \frac{D c_p}{D t} = \dot{q}_v + \frac{\Pi h^*}{A_{av}}(T_f - T) + \sum \frac{Q}{A_{av}} \tag{10}$$

where Π is the wetted perimeter. This equation is expressed as:

$$V\frac{d(\rho_C c_{p,C} T_C)}{dt} + (\dot{m}_e c_{p,e} T_e - \dot{m}_w c_{p,w} T_w) - k_e A_e \frac{2(T_E - T_C)}{\Delta z_C + \Delta z_E} + k_w A_w \frac{2(T_C - T_W)}{\Delta z_C + \Delta z_W} +$$
$$- \rho_C T_C \left[\frac{\partial (c_{p,C})}{\partial t} + \frac{\dot{m}_C}{A_C \rho_C} \frac{c_{p,e} - c_{p,w}}{\Delta z_C} \right] V = \tag{11}$$
$$= \dot{q}_{v,C} V + \Pi_f \Delta z_C h^* (T_{f,C} - T_C) + \Pi_{nf} \Delta z_C h_{nf} (T_{nf,C} - T_C) + \sum Q_C \Delta z_C$$

The diffusion term was expressed with a finite difference approach based on cell faces, and the advection term was treated by means of a second order upwind scheme to avoid numerical instabilities. Temperatures at cell faces are computed as follows, taking care of the possibly different mesh distribution in the axial direction:

$$T_e = T_C + \frac{(T_C - T_W)\Delta z_C}{\Delta z_C + \Delta z_W} \tag{12}$$

$$T_w = T_W + \frac{(T_W - T_{WW})\Delta z_W}{\Delta z_{WW} + \Delta z_W} \tag{13}$$

Material properties at cell faces can be evaluated by linear interpolation, weighed according to the spatial discretisation step. For instance, the specific heat at cell face e can be evaluated as:

$$c_{p,e} = g_C c_{p,C} + (1 - g_C) c_{p,E} = \frac{\Delta z_C}{\Delta z_C + \Delta z_E} c_{p,C} + \frac{\Delta z_E}{\Delta z_C + \Delta z_E} c_{p,E} \tag{14}$$

For the thermal conductivity, instead, an equivalent resistance scheme is used, to better treat the diffusion process:

$$\frac{1}{k_e} = \left(\frac{1 - g_e}{k_E} + \frac{g_e}{k_C} \right) = \left(\frac{\Delta z_E}{k_E(\Delta z_C + \Delta z_E)} + \frac{\Delta z_C}{k_C(\Delta z_C + \Delta z_E)} \right) \tag{15}$$

The first term on the right hand side of equation (11) represents a possible volumetric heat generation, allowing to account for, e.g., heat deposition by gamma rays outside the rods. The last two terms account for the thermal coupling with fuel rods and non-fuel rods, respectively. h^* is computed as:

$$h^* = h_C \frac{T_{s,C} - T_{pb,C}}{T_{f,C} - T_{pb,C}} \tag{16}$$

A possible convective heat transfer to non-fuel rods, $h_n f$, can be estimated during transients through an iterative procedure, solving a separate time-dependent energy equation for these rods.

The last term describes the coupling between an assembly and the neighbouring ones, with:

$$Q_C = \Pi_{ij} h_{ij} (T_{C,i} - T_{C,j}) \tag{17}$$

where h_{ij} is an equivalent heat transfer coefficient, which is computed by means of a series of thermal resistances accounting for heat conduction in the inter-wrapper gap and the walls, and for convective heat transfer between the coolant inside each HA and the wall. For two neighboring hexagons i and j:

$$h_{ij} = \frac{1}{R_{tot}} = \left(\frac{1}{h_{conv,i}} + \frac{s_a}{k_{box}} + \frac{s_c}{k_{IW}} + \frac{s_a}{k_{box}} + \frac{1}{h_{conv,j}} \right)^{-1} \tag{18}$$

IV.B.4. Conservation of energy for fuel rods

The energy conservation equation for fuel rods was also written in terms of temperature, and reads:

$$\frac{\partial(\rho c_p T)}{\partial t} - \nabla[k \nabla T] = \dot{q}_v + \frac{\Pi h^*}{A_f} (T_{cool} - T_{fuel}) \tag{19}$$

which can be expressed as:

$$\begin{aligned} & V \frac{d(\rho_c c_{p,c} T_{f,C})}{dt} - k_e A_f \frac{(T_{f,E} - T_{f,C})2}{\Delta z_C + \Delta z_E} + k_w A_f \frac{(T_{f,C} - T_{f,W})2}{\Delta z_C + \Delta z_W} = \\ & = \dot{Q}_f \Delta z_C + \Pi_f \Delta z_C h^* (T_{cool,C} - T_{f,C}) \end{aligned} \tag{20}$$

where $\dot{Q}_f$ is the linear heat generation distribution (either provided by the NE module, in case of coupled simulations, or possibly read from input files), Π_f is the total rod perimeter and A_f the fuel cross sectional area, which is treated as constant along the channel. The diffusion term can be treated with a finite difference approach, as done for the coolant (15).

In steady state, the surface temperature can be expressed analytically as a function of $\dot{Q}_f$,

assuming a parabolic radial temperature distribution inside the pin:

$$T_s = T_f - \frac{\dot{q}_v r_{pin}^2}{8k} \tag{21}$$

For an annular rod with internal radius r_v, this translates into:

$$T_s = T_f - \frac{\dot{q}_v (r_{pin}^2 - r_v^2)}{8k} \left[1 - \frac{ln(r_{pin}/r_v)^2}{(r_{pin}/r_v)^2 - 1} \right] \tag{22}$$

Due to the h^* approach, the steady matrix cannot be solved in one step. To tackle this issue, a pseudo-transient approach is adopted. The time step used can be derived from the actual physical characteristics of the channel, according to the time scales for heat and momentum diffusion. Following the approach used in ANSYS-CFX [14], the pseudo-time step is chosen as:

$$\begin{aligned} \Delta t &= \frac{L_f}{\alpha_f} \\ L_f &= min(\sqrt{V_f}, max(L_{chan}), \sqrt{A_f})) \\ &\text{and user defined : } L_f = D_{fuel} \end{aligned} \tag{23}$$

from the solid material, and for the coolant terms as

$$\begin{aligned} \Delta t &= \frac{L_{cool}}{\nu} \\ \nu &= \frac{\mu}{\rho} \\ L_{cool} &= min(\sqrt{V_{cool}}, max(L_{chan}), D_{eq})) \end{aligned} \tag{24}$$

The pseudo transient practically consists in adopting an under-relaxation factor that is different from cell to cell. Adopting an implicit Euler scheme for the time discretisation in an iterative system, both methods practically imply an additional source term and contribution to central coefficient A_p; the following similarity between under relaxation ϕ and time step can be seen[15]:

$$\Delta t = \frac{\rho \phi \Delta \Sigma}{A_P(1-\phi)} \qquad \phi = \frac{A_P \Delta t}{A_P(\Delta t + \rho \Delta \Sigma)} \tag{25}$$

The application of under-relaxation values is still turned on, to avoid potential problems in start-up

procedures in case the power deposition is high.

During transients, instead, 1D radial heat conduction is numerically solved at each axial step. The corresponding differential equation is:

$$\frac{d(\rho c_p T)}{dt} - \frac{1}{r}\frac{d}{dr}\left(rk\frac{dT}{dr}\right) = \dot{q}_v \tag{26}$$

and the corresponding radial discretisation is:

$$r_C V \frac{d(\rho_C c_{p,C} T_C)}{dt} - \left(r_e k_e \frac{(T_E - T_C)2}{\Delta r_C + \Delta r_E} - r_w k_w \frac{(T_C - T_W)2}{\Delta r_C + \Delta r_W}\right)\Delta z_C = r_C \dot{q}_{v,C} V \tag{27}$$

IV.B.5. Thimble modelling

During the validation campaign conducted considering the EBR-II SHRT-45R transient [3], FRENETIC was featured with the capability to model the possible presence of a thimble between the an inner and an outer wrapper. As the thimble flow may be comparable to the main flow cooling the fuel pins within the inner wrapper, a dedicated model, named Box-In-the-Box (BIB) was developed, in which convective heat transfer within the thimble is added to the thermal resistance of inter-wrapper gap. The convective heat transfer coefficient used is computed from the Seban-Shimazaki formula. The thimble itself is modeled as an additional channel with no power generation, thermally connected to the inner wrapper and to neighboring HAs [16].

The thermal resistance for the coupling between the channel and the thimble is:

$$R_{int->thimb,i} = \frac{1}{h_{conv,i}} + \left(\frac{s}{k}\right)_{inbox} + \frac{1}{h_{conv,thimb,i}} \tag{28}$$

while the one between the thimble and the surrounding assemblies is:

$$R_{tot} = \frac{1}{h_{conv,thimb,i}} + \left(\frac{s}{k}\right)_{outbox,i} + \left(\frac{s}{k}\right)_{clearance} + \left(\frac{s}{k}\right)_{outbox,j} + \frac{1}{h_{conv,j}} \tag{29}$$

The new version of the TH module clearly retains the BiB model, but it solves the thimble directly in the main channel. In other words, eq. (11) is solved by suitably adapting the coupling

terms.

IV.B.6. Boundary conditions

The numerical discretisation of the advection and conduction equations demands the application of suitable boundary conditions. As for the fuel energy conservation equations, adiabatic boundary conditions are considered. This choice is justified by the fact that the upper extremity of the pins, which are thermally insulated, is far from the heated zone. Concerning the bottom extremity of the fuel rod, a convective boundary condition is set in order to overcome numerical issues during the initialisation of the steady state calculations:

$$T_w = \frac{h\, T_{cool,in} + \left(\frac{k_w}{\Delta z_C/2}\right) T_C}{h + \frac{k_w}{\Delta z_C/2}} \tag{30}$$

A similar approach was applied to the diffusive terms in the equation for the coolant, while convective boundary conditions are naturally given by imposing the coolant inlet temperature for each FA. Moreover, to prevent numerical issues that could potentially arise in the case of no initial heating of the coolant, an additional conductive boundary condition is imposed for the coolant in the first node, adding an additional term to the central value and source vector in the form:

$$\begin{aligned} T_w &= \frac{k_w}{\Delta z_C/2} \\ b_C &= \frac{k_w}{\Delta z_C/2} \cdot T_{in}. \end{aligned} \tag{31}$$

Finally, concerning the radial fuel energy conservation equations, an adiabatic boundary condition is adopted on the inner side of the model, while a Robin (mixed) boundary condition is imposed for the outer side.

IV.C. Solution scheme

IV.C.1. Transient solution

During transient simulations, the governing equations are discretised in both space and time. The solution within each HA is treated implicitly, whereas the inter-HA coupling, as well as the coupling with the NE code, is treated explicitly. The finite volume discretised equations presented

in the previous section, once properly arranged and simplified, turn out to be decoupled, except for the temperature dependence of physical properties. This decoupling considerably simplifies the numerical solution scheme since, given the inlet mass flow, the energy equation can be resolved. In this way, the properties for the momentum equation can be determined and, consequently, the pressure distribution can be computed.

IV.C.2. Steady state solution

A different approach is needed for approaching the steady state problem, where the terms associated with the finite difference discretisation of the time derivative disappear. In this case, it is no longer possible to treat explicitly the inter-assembly coupling. The problem was tackled by adopting an iterative procedure, which is initialized by assuming that no heat transfer between assemblies occurs. The entire core is solved iteratively until a tolerance criterion is satisfied, exploiting the temperatures found at the previous step for the transverse coupling. The steady-state solver is also used for initializing the transient calculation.

IV.C.3. Matrix formulation

The two coupled energy equations for each assembly, during steady state, are intended to be solved together in a matrix system in the form:

$$
\begin{bmatrix}
\ddots & & & & & & & \vdots \\
\cdots & BB_{i-1}+EE_{i-1} & -EE_{i-1} & CC_{i-1} & 0 & 0 & \cdots & \cdots \\
\cdots & -ee_{i-1} & bb_{i-1}+ee_{i-1} & 0 & cc_{i-1} & 0 & 0 & \cdots \\
\cdots & AA_i & 0 & BB_i+EE_i & -EE_i & CC_i & 0 & \cdots \\
\cdots & 0 & aa_i & -ee_i & bb_i+ee_i & 0 & cc_i & \cdots \\
\cdots & DD_{i+1} & 0 & AA_{i+1} & 0 & BB_{i+1}+EE_{i+1} & -EE_{i+1} & \cdots \\
\cdots & \cdots & 0 & 0 & aa_{i+1} & -ee_{i+1} & bb_{i+1}+ee_{i+1} & \cdots \\
\vdots & & & & & & & \ddots
\end{bmatrix}
\cdot
\begin{bmatrix}
\vdots \\
TT_{i-1} \\
TT_{f,i-1} \\
TT_i \\
TT_{f,i} \\
TT_{i+1} \\
TT_{f,i+1} \\
\vdots
\end{bmatrix}
=
\begin{bmatrix}
\vdots \\
QQ_{i-1} \\
QQ_{f,i-1} \\
QQ_i \\
QQ_{f,i} \\
QQ_{i+1} \\
QQ_{f,i+1} \\
\vdots
\end{bmatrix}
\tag{32}
$$

The resulting system is diagonally dominant, with the unknown vector written maintaining the coolant and fuel rods at the same axial index. The source terms contain the heat generation terms and also the known quantities and terms associated with the coupling with other channels. The terms DD, AA, aa, BB, bb, CC, cc contain the multiplicative terms of advection, conduction and thermal coupling from equation (11) and (20). In the following, the coolant and fuel equations

are indicated with upper and lower cases, respectively. The coefficients for the coolant reads

$$
\begin{aligned}
DD &= +\dot{m}_w c_{p,w} \left(\frac{\Delta z_W}{\Delta z_{WW} + \Delta z_W} \right) \\
AA &= -\dot{m}_w c_{p,w} \left(1 + \frac{\Delta z_W}{\Delta z_{WW} + \Delta z_W} \right) - \dot{m}_e c_{p,e} \left(\frac{\Delta z_C}{\Delta z_C + \Delta z_W} \right) - k_w A_w \frac{2}{\Delta z_C + \Delta z_W} \\
CC &= \dot{m}_e c_{p,e} \left(1 + \frac{\Delta z_C}{\Delta z_C + \Delta z_W} \right) - k_e A_e \frac{2}{\Delta z_C + \Delta z_E} \\
BB &= \dot{m}_e c_{p,e} \left(1 - \frac{\Delta z_C}{\Delta z_C + \Delta z_E} \right) - \dot{m}_w c_{p,w} \left(\frac{\Delta z_W}{\Delta z_C + \Delta z_W} \right) + k_e A_e \frac{2}{\Delta z_C + \Delta z_E} \\
&+ k_w A_w \frac{2}{\Delta z_C + \Delta z_W} + \Pi_{nf} \Delta z_C h \\
EE &= \Pi_f \Delta z_C h^* \\
QQ &= \dot{q}_{v,C} V + \Pi_{nf} \Delta z_C h T_{nf,C} + \sum Q_C \Delta z_C,
\end{aligned}
\tag{33}
$$

while, for fuel

$$
\begin{aligned}
aa &= -k_w \frac{2}{\Delta z_C + \Delta z_W} \\
cc &= -k_e \frac{2}{\Delta z_C + \Delta z_E} \\
bb &= k_e \frac{2}{\Delta z_C + \Delta z_E} + k_w \frac{2}{\Delta z_C + \Delta z_W} \\
ee &= \frac{\Pi_f \Delta z_C}{A_f} h^* \\
QQ_f &= \dot{Q}_f \frac{\Delta z_C}{A_f}.
\end{aligned}
\tag{34}
$$

The matrix is stored in a band format and solved with `DGBVS`, a general linear solver for double precision banded matrices, available in the Lapack libraries [17]. The adoption of a sparse band format allows to reduce the memory footprint of the calculation and, at the same time, to decrease the computational time for the solution of the linear system by approximately 2 orders of magnitude, compared to the equivalent general solver for matrix systems (called `DGEVS`.

The terms DD, AA, BB, CC in the matrix for the time-dependent coolant equation are built in the same way as for the steady state solution. Additionally, the time discretisation results in a 4x Nodes banded matrix solved with `DGBVS`, iteratively linked to the solver for the pins.

By solving the momentum equation, pressure drops across each volume element are com-

puted. The resulting pressure profile is then reconstructed based on the given inlet or outlet boundary conditions. As mentioned before, the user is also provided with the option to impose the pressure drop across the core rather than the mass flow for each HA. In this case, the system of equations is the same, despite an outer iteration procedure is required to correct the assumption on the mass flow rate per HA at iteration $j+1$,

$$\begin{aligned} \Delta p_{boundary} - \Delta p_j = \Delta p' => \Delta \dot{m}'(f, L, \Delta p', Deq) \\ \dot{m}_{j+1} = \dot{m}_j + \varphi(\Delta \dot{m}') \end{aligned} \tag{35}$$

IV.D. Solution optimization

IV.D.1. Adaptive definition of solution tolerances

Adaptive tolerances are adopted to avoid over-solving during the iterative process. Indeed, due to the initialization performed assuming no inter-HA coupling, the first iterations towards steady-state are naturally affected by errors, thus making an accurate solution of the single HA problem not necessary. The number of iterations required for convergence was nearly halved thanks to the adoption of the Residual Balance method [18], which prevents over-solving without degrading the convergence rate during steady-state initialization. The inner iterations for the solution of the energy equations are carried out keeping as tolerance the residual norm of the constituent core solver, as in eq. (36), which is kept just below the residual norm of the maximum error found in the other channels at the previous core iteration step. In this way, each channel solver will not progress too far beyond the others. Afterwards, the new coupling heat flux between channels is computed, bringing all the core HAs to convergence together,

$$||R_i^{chan}|| < a \ \cdot max(||R_{i-1}^1||, \ ..., \ ||R_{i-1}^{NChan}||) \tag{36}$$

where the constant a is specified by the user as $a = 0.1$. A possible initial reference value for the residual norm is approximately 0.1.

IV.D.2. Parallelization

The solution strategy adopted for the problem lends and, specifically, the explicit inter-channel coupling lends itself to the parallel solution of each channel, which could be assigned, in principle, to a different thread. Taking advantage of existing frameworks, the open-source OpenMP libraries were linked to the code using a dynamic scheduling. This choice is due to the fact that the assemblies can have quite different features (e.g., in terms of heat deposition and number of rods), which can lead to very different computational times. For other tasks such as the assignment of boundary conditions and inter-channel coupling, a faster static schedule is implemented.

V. STEADY-STATE SIMULATION OF THE ADVANCED LEAD-COOLED FAST REACTOR EUROPEAN DEMONSTRATOR CORE

V.A. The ALFRED reactor core

In view of the steps foreseen by the roadmap of the European Lead Fast Reactor in Europe, a full-scale demonstration plant, ALFRED, is planned to demonstrate the feasibility of the LFR technology. In the framework of the progressive up-scaling towards industrial maturity, ALFRED should prove the viability of electricity production from LFR systems before moving to the first-of-a-kind representative of a commercial ELFR [19]. It should be noticed that, within this staged approach, ALFRED has the double function of a prototype for the ELFR large scale reactor and of a demonstrator for the LFR Small Modular Reactor (SMR).

The ALFRED reactor is of pool type, cooled by lead in forced circulation - while natural circulation plays an important safety role in emergency conditions. The core design was driven by the aim of embedding all the features that are favourable in design extension conditions, so to ensure the respect of the safety limits even in the most extreme conditions. The resulting core is calibrated to a thermal power of 300 MW with assemblies in a pseudo cylindrical scheme. The core inlet temperature is 400°C and the average core outlet temperature is 480°C. The core is composed as follows [19]:

- 171 hexagonal fuel assemblies. Each element is made of 127 fuel rods in a triangular lattice inside a 4mm-thick box made of ferritic/martensitic steel T91. The fuel is of MOX type, with a maximum enrichment of 30%. The active region, which is 0.6 m long, starts 1.4 m

above the bottom of the pins;

- 12 control rods, used for both reactor control in normal operation (start-up, reactivity control during the fuel cycle, power tuning and shutdown) and for emergency shutdown;
- 4 safety rods, used only for reactor emergency shutdown;
- 108 dummy elements, surrounding the rest of the core to provide shielding for the inner vessel against neutron damage.

V.B. Steady state simulation in normal operating conditions

To demonstrate the capabilities of the FRENETIC code with the new TH module in place, an application to the core of ALFRED is here presented. Care will be devoted to compare results with those obtained with the previous code version in the same configuration. This is a crucial step in order to be able to extrapolate the results of the past validation activities to the new code version.

The input generation was performed by relying on a purposely developed Python toolkit named COREutils. The adopted core configuration is depicted in figure 3. The HAs marked as BA and EL represent the core barrel and external lead, respectively. Indeed, as explained in section II.C, in order to retrieve the correct radial neutron flux distribution, these assemblies must be included in the NE calculation domain, which must coincide with the TH domain in the current code implementation. However, the TH conservation equations are not actually solved in this region, since this would require complex 3D calculations well beyond the scope of FRENETIC, which is concerned with the core region. Therefore, for the sake of simplicity, the inlet temperature is propagated to the entire HA length in that region, which is therefore actually composed by fictitious HAs in the model.

Both simulations (with the old and new code implementation, respectively) were performed on the same machine, an average market laptop, taking advantage of the introduced parallelization with 4 threads. 351 nodes were used for the axial discretisation, with 200 nodes inside the active region. The resulting computational time required to reach the steady state condition is reported in table I. The new code implementation shows a remarkable reduction in simulation time, resulting in a 14x faster convergence with respect to the previous implementation, notwithstanding a stricter

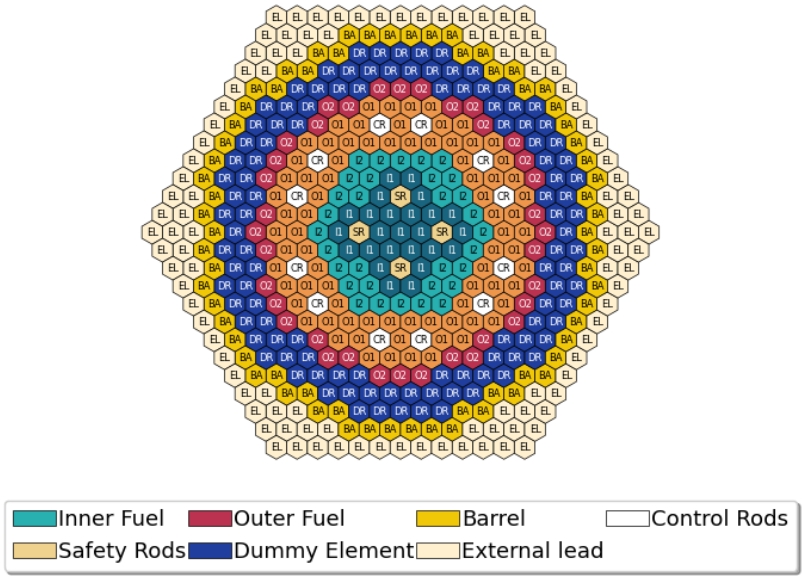

Fig. 3. *ALFRED core configuration.*

tolerance requested.

TABLE I
Execution times and maximum relative error at the last core iteration.

	Execution time	NE iterations	ϵ_r
New	43m 39s	51	8.48E-10
Old	10h 24m	52	4.23E-08

To have a glimpse of the FRENETIC calculation results, the radial distribution of the volumetric heat deposition inside the fuel pins, measured 25 cm above the center of the active region, is shown in figure 4 (left), and the resulting outlet coolant temperature is shown in figure 4 (right).

Figure 5 (left) shows that the new TH module agrees with the previous implementation for what concerns the coolant temperature increase, with only a slight discrepancy which can be explained in terms of different modelling assumptions. It is worth mentioning that the step-wise power deposition profile shown in figure 5 (right) - which again shows the agreement between the old and the new version of the code - is a consequence of the coarse mesh nodal method adopted for the NE module.

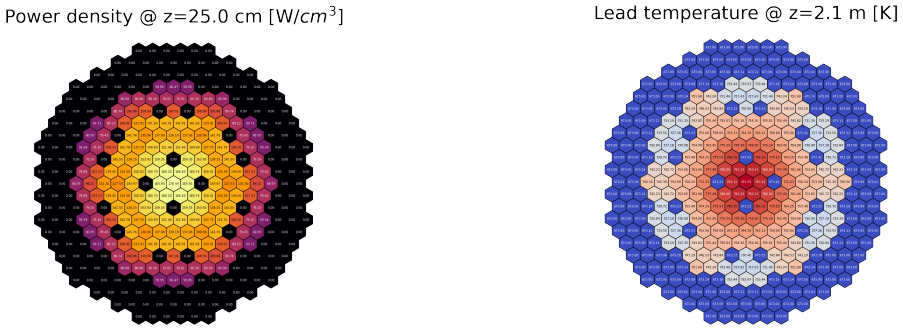

Fig. 4. *Fission power density 25 cm above the center of the active region (left) and coolant temperature at the end of the active region (right).*

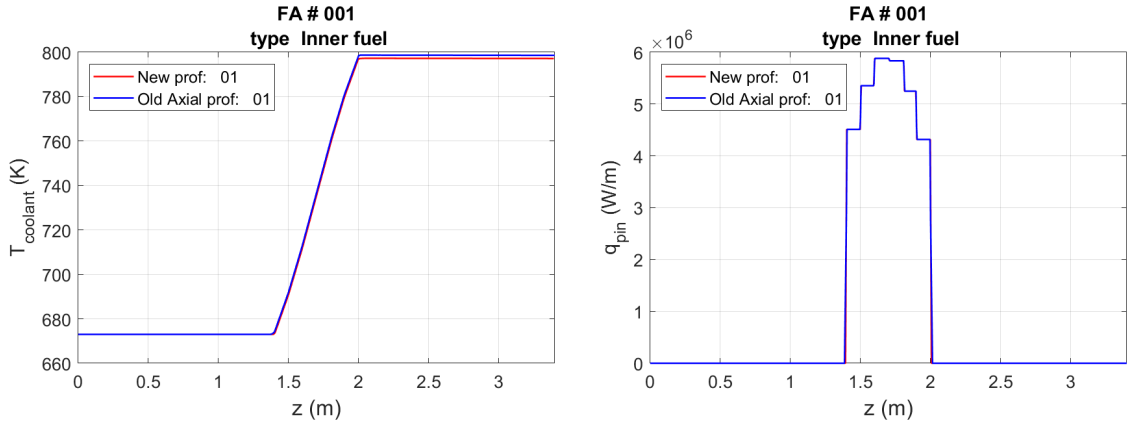

Fig. 5. *Coolant temperature (left) and power deposition (right) for the central HA (# 1).*

Conversely, figure 6 (left) shows a discrepancy regarding the axial profile of the average pin temperature, which is associated to a more accurate evaluation implemented in the new code version. Moreover, it is evident that the previous implementation, based on finite elements, was prone to numerical oscillations unless a careful choice of the axial discretisation was performed, whereas the new model is more robust. Figure 6 (right) shows that, apart from the above-mentioned oscillations that were already present in the older code version, the pin surface temperature is consistently evaluated.

Based on these results, it can be stated that the updated implementation of the TH module is faster and more efficient with respect to the previous one, thanks to a suitable physics-based simplification of the governing equations that does not hinder the calculation accuracy. Moreover,

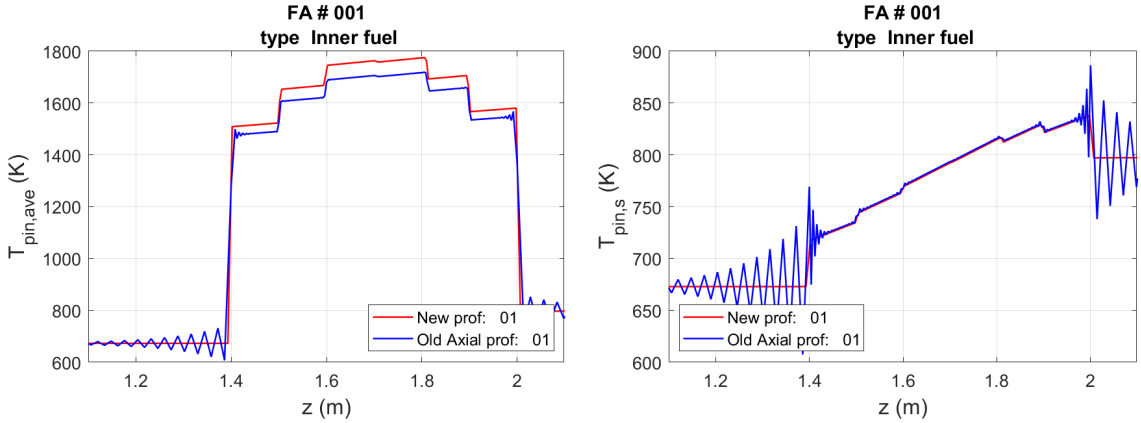

Fig. 6. *Axial profile of the pin average temperature (left) and surface temperature (right) for the central HA (# 1).*

spatial oscillations that could arise in the previous version are now suppressed thanks to the robustness of the finite volume discretisation.

V.C. Transient simulations for accidental scenarios

In this section, the simulation of a couple of safety-relevant accidental scenarios, involving a reduction of the mass flow in the core, is presented and discussed.

V.C.1. Unprotected loss of flow

First, an Unprotected Loss Of Flow (ULOF) transient was considered. This transient is initiated by the simultaneous loss of power supply to all primary pumps, in conjunction to a failure of the SCRAM intervention, and is among the ones considered for Design Extension Conditions (DEC) [20]. For the sake of simplicity, the results here presented were computed taking the time dependence of the mass flow rate profile from a previous study where the same scenario was simulated by coupling the RELAP5-3D and PHYSICS codes [21], see figure 7 (left). Figure 7 (right) shows the reactivity feedback computed by FRENETIC.

The computed transient evolution of the core fission power and temperature distribution was compared to results presented in [21], showing qualitative agreement with some noticeable discrepancies. The discrepancy in the fission power evaluated by FRENETIC and RELAP, shown in fig. 7 (left), is likely caused by the fact that FRENETIC does not include any reactivity feedback associated with thermo-mechanical effects, such as fuel expansion. The discrepancy in fission power

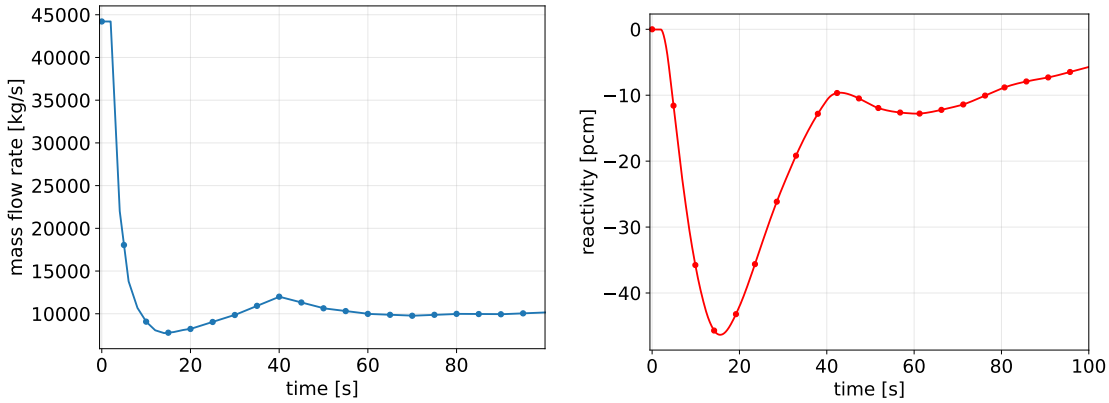

Fig. 7. *Imposed time dependence of the total core mass flow rate (left) and resulting reactivity feedback computed by FRENETIC (right).*

generation is reflected in the temperature distribution, see for instance the maximum cladding surface temperature as reported in figure 7 (right). This discrepancy suggests that further development efforts in FRENETIC should be devoted for including the core expansion effects on the reactivity.

V.C.2. Total instantaneous blockage

An unprotected Total Instantaneous Blockage (TIB) of a single Fuel Assembly (FA) was also considered [22]. In this scenario, the flowrate in a single FA is suddenly halted, again assuming the failure of the SCRAM intervention. The temperature of the affected FA rapidly increases due to the lack of cooling, so that the evolution of this accident naturally involves as a minimum the failure of the cladding, followed by a complex accident evolution possibly involving the propagation of the failure to neighboring FAs (which are heated via inter-FA heat transfer), the melting of the fuel and its relocation. Simulating this evolution of this kind of accidental scenario is well beyond the scope of FRENETIC and requires more sophisticated codes such as SIMMER [23]. Nevertheless, FRENETIC can be used to efficiently simulate the first part of the accidental scenarios, accounting for reactivity feedback.

As a first step, pure TH simulations were considered, hence neglecting the reactivity thermal feedback. In this case, the only cooling mechanism considered in FRENETIC is the inter-FA heat transfer. Three cases were considered, where the mass flow rate was decreased from 172.3 kg/s to 0.1 kg/s in 10, 1 and 0.1 seconds, respectively (see figure 9 (left). The corresponding evolution

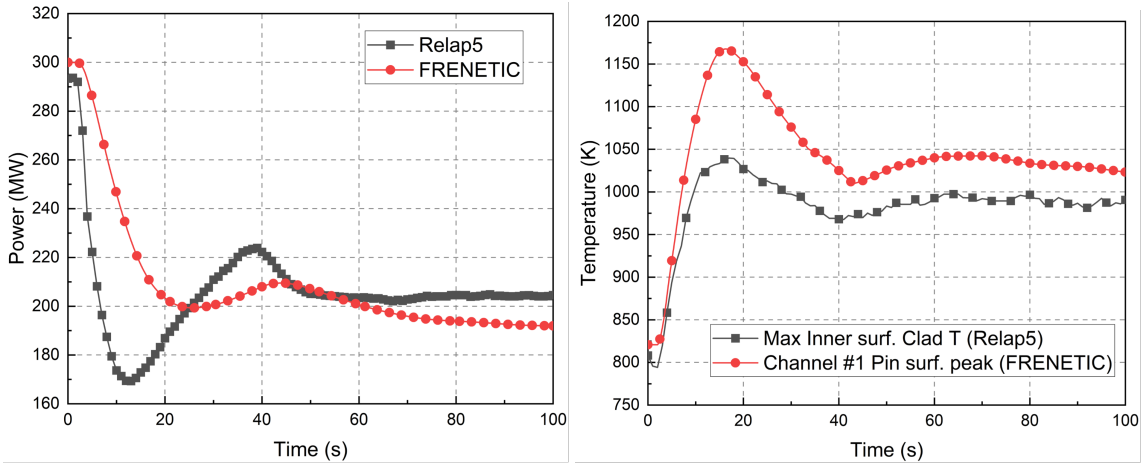

Fig. 8. *Total core power evolution computed by FRENETIC and RELAP (left) and resulting computed maximum surface temperature (right).*

of the maximum coolant temperature is shown in figure 9 (right). The spatial distribution of the peak surface temperature featuring the "average" fuel pin in each HAs is depicted in fig. 10 for different time instants.

A case with NE-TH coupling was then performed. The temperature increase following the TIB is very large, leading to a significant feedback effect, which however is spatially localized to the affected FA, so that the overall core reactivity is only slightly reduced, see figure 11.

Figure 12 (left) shows the computed fuel and coolant temperatures for the blocked FA. Figure 12 (right) shows that NE-TH feeedbacks effectively reduce the local power deposition, thus leading to a slower temperature increase. It is again stressed that FRENETIC can only be used to describe the first part of such a severe transient, i.e., until the integrity of core structures is not lost. In this time span, FRENETIC accounts for coolant inertia, inter-HA heat transfer (thus providing information on the possible propagation of the accident to neighboring HAs) and reactivity feedbacks.

VI. CONCLUSIONS AND PERSPECTIVE

In this paper, the development of a new TH module for the FRENETIC code was presented. The newly implemented finite volume formulation, together with the updated iterative method for the core calculation and a suitable code parallelization, leads to better code performances in terms of computational time, without losing accuracy. Moreover, the approach adopted for the

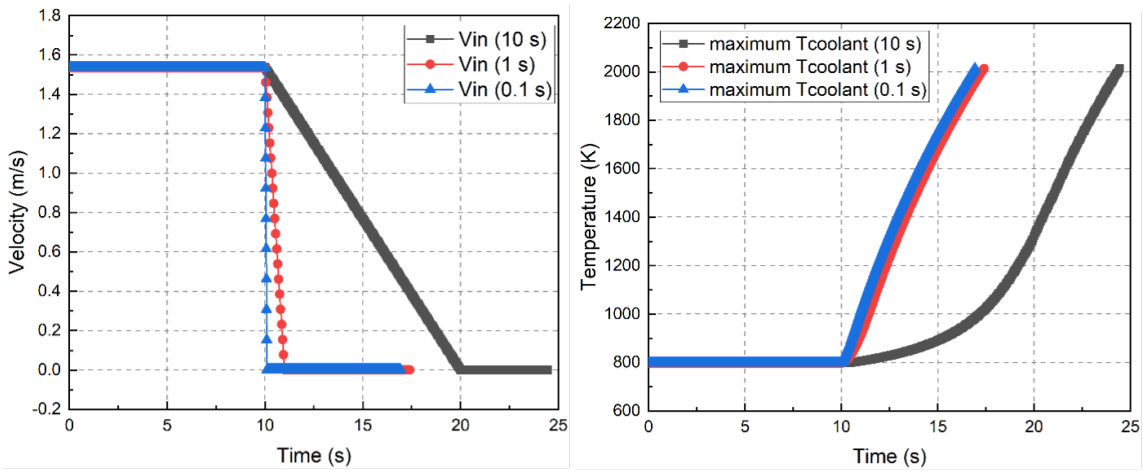

Fig. 9. *Imposed time dependence of the inlet velocity to the central FA (left) and resulting maximum coolant temperature computed by FRENETIC (right).*

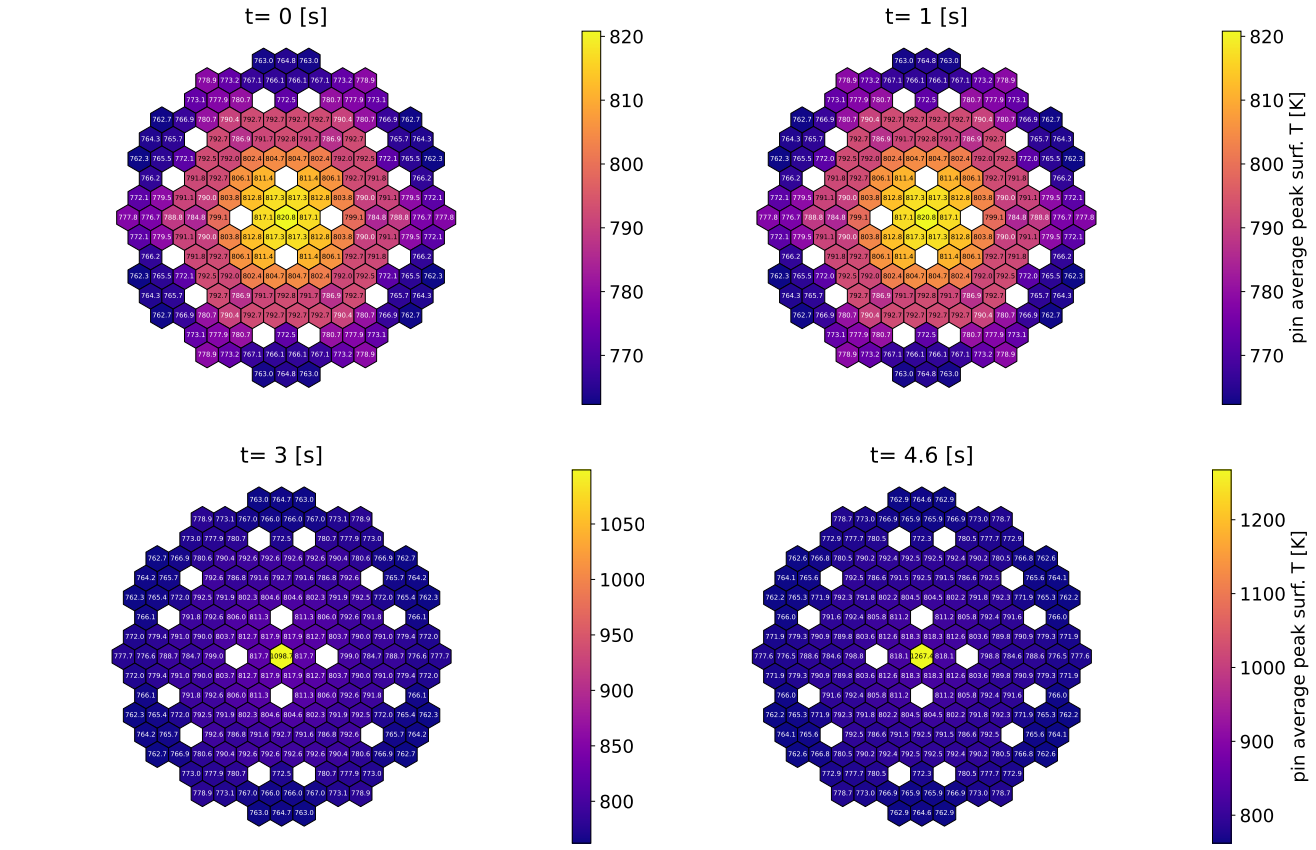

Fig. 10. *Spatial distribution of the peak surface temperature in the average-pin featuring each HAs.*

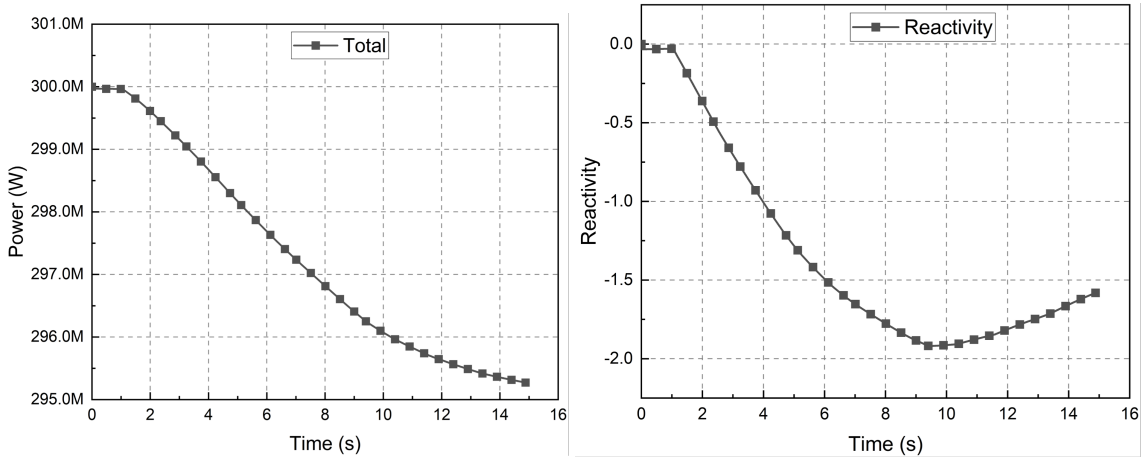

Fig. 11. *Computed time evolution of the total core power (left) and reactivity (right) for the TIB case with NE-TH coupling.*

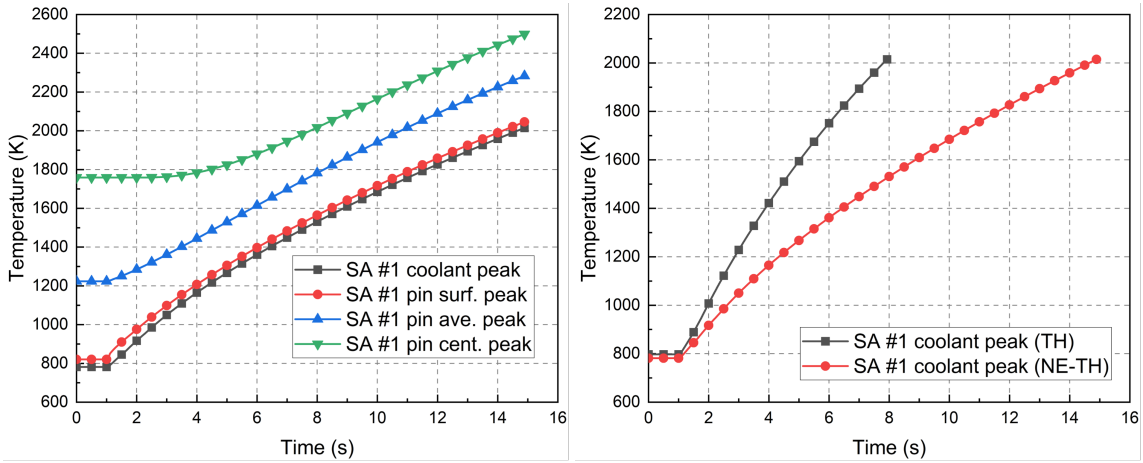

Fig. 12. *Computed temperature evolution for the blocked FA considering NE-TH feedbacks (left) and comparison between the maximum temperature in the case with feedbacks (NE-TH) and in the case without feedbacks (TH) (right).*

code development aimed at enhancing its modularity, thus representing a solid base for future improvements of the code. The agreement of the new code version with the previous one was carefully assessed, and any deviation justified - to ensure that it resulted from an increase of the fidelity of the model rather than from numerical errors.

Possible future improvements, aimed at speeding up coupled transient simulations, will focus on the NE solution scheme and on the NE-TH coupling procedure, which should adopt techniques for avoiding over-solving similar to the residual balance method now in place for the outer iterations of the TH module. Concerning the predictive capabilities of the tool, future efforts should be devoted to incorporate some physical models accounting for the radial and axial expansion of the core, which is responsible for a non-negligible reactivity feedback.

In conclusion, it can be stated that the new TH module, together with a set of Python classes purposely developed to simplify the input generation and post-processing phases, improved the code quality and maturity. Based on these achievements, a short-term plan to render the code open-source appears very feasible. This can potentially increase the number of researchers that adopting FRENETIC for the simulation of full-core coupled NE-TH transients in fast reactors and/or as a platform to test new solution methods.

ACKNOWLEDGMENTS

One of the authors, Y.L., acknowledges funding from the China Scholarship Council (CSC) which allowed him to spend one year as a visiting PhD Student at Politecnico di Torino.

NOMENCLATURE

Δz	Axial discretisation step length in m
$\dot{m}$	Mass flow rate in $\frac{\mathrm{kg}}{\mathrm{s}}$
$\dot{q}_v$	Volumetric power source in $\frac{\mathrm{W}}{\mathrm{m}^3}$
$\dot{Q}_f$	Linear power in $\frac{\mathrm{W}}{\mathrm{m}}$
λ	Bulk viscosity coefficient Pa $\times$ s
μ	Fluid viscosity in Pa $\times$ s
Φ	Viscous dissipation term in $\frac{1}{\mathrm{s}^2}$
Π	Wetted perimeter in m
ψ	Velocity divergence, squared $\frac{1}{\mathrm{s}^2}$
ρ	Fluid density in $\frac{\mathrm{kg}}{\mathrm{m}^3}$
Σ_x	Macroscopic neutron cross section for reaction of type x in $\frac{1}{\mathrm{cm}}$
A	Fluid flow area in m^2
A_{av}	Fluid flow average area in m^2
c_p	Fluid specific heat at constant pressure in $\frac{\mathrm{J}}{\mathrm{kgK}}$
f_b	Body force term in $\frac{\mathrm{Pa}}{\mathrm{m}}$
f_C	Darcy–Weisbach friction factor, dimensionless
g	Gravitational acceleration in $\frac{\mathrm{m}}{\mathrm{s}^2}$
h^*	Heat transfer coefficient in $\frac{\mathrm{W}}{\mathrm{m}^2\mathrm{K}}$
k	Fluid heat conductivity in $\frac{\mathrm{W}}{\mathrm{mK}}$
K_{loc}	Localised pressure drop coefficient, no dimension
p	Fluid pressure in Pa

Q External heating term in $\frac{\mathrm{W}}{\mathrm{m}}$

r Radial spatial coordinate in m

r_{pin} Outer cladding radius in m

R_{tot} Total thermal resistance in $\frac{\mathrm{W}}{\mathrm{m^2K}}$

r_v Inner cladding radius in m

s Thickness of the inter-wrapper gap in m

T Fluid temperature in K

t Time instant in s

T_c Node-averaged coolant temperature in K

T_f Node-averaged fuel temperature in K

U Axial component of the velocity in $\frac{\mathrm{m}}{\mathrm{s}}$

v Fluid velocity in $\frac{\mathrm{m}}{\mathrm{s}}$

z Axial spatial coordinate in m

V Volume in m^3

REFERENCES

[1] A. Stanculescu, "GIF R&D outlook for generation IV nuclear energy systems," *Generation IV International Forum, Paris, France*, Paris, France (2009).

[2] R. Zanino, R. Bonifetto, A. Chiampichetti, I. Di Piazza, L. Savoldi, and M. Tarantino, "First validation of the FRENETIC code thermal-hydraulic model against the ENEA integral circulation experiment," *Transactions of the American Nuclear Society*, **107**, 1395 (2012).

[3] D. Caron, R. Bonifetto, S. Dulla, V. Mascolino, P. Ravetto, D. Valerio, and R. Zanino, "Full-core coupled neutronic/thermal-hydraulic modelling of the

EBR-II SHRT-45R transient," *International Journal of Energy Research*, **42**, 134 (2018); https://doi.org/10.1002/er.3571.

[4] G. F. Nallo, N. Abrate, S. Dulla, P. Ravetto, and D. Valerio, "Neutronic benchmark of the FRENETIC code for the multiphysics analysis of lead fast reactors," *The European Physical Journal Plus*, **135:238** (2020); https://doi.org/10.1140/epjp/s13360-020-00171-8.

[5] M. Massone, N. Abrate, G. F. Nallo, D. Valerio, S. Dulla, and P. Ravetto, "Code-to-code SIMMER/FRENETIC comparison for the neutronic simulation of lead-cooled fast reactors," *Annals of Nuclear Energy*, **174**, 109124 (2022); https://doi.org/10.1016/j.anucene.2022.109124., URL https://www.sciencedirect.com/science/article/pii/S0306454922001591.

[6] D. Caron, S. Dulla, P. Ravetto, L. Savoldi, and R. Zanino, "Models and methods for the representation of decay and photon heat in spatial kinetics calculations," *Proceedings of PHYSOR 2016*, 2416–2425, USA (2016).

[7] R. Bonifetto, S. Dulla, P. Ravetto, L. S. Richard, and R. Zanino, "A full-core coupled neutronic/thermal-hydraulic code for the modeling of lead-cooled nuclear fast reactors," *Nuclear Engineering and Design*, **261**, 85 (2013); 10.1016/j.nucengdes.2013.03.030., URL http://dx.doi.org/10.1016/j.nucengdes.2013.03.030.

[8] J. Leppänen, M. Pusa, T. Viitanen, V. Valtavirta, and T. Kaltiaisenaho, "The Serpent Monte Carlo code: Status, development and applications in 2013," *Annals of Nuclear Energy*, **82**, 142 (2015); 10.1016/j.anucene.2014.08.024.

[9] F. Moukalled, L. Mangani, and M. Darwish, *The Finite Volume Method in Computational Fluid Dynamics: An Advanced Introduction with OpenFOAM® and Matlab®*, Springer (2021).

[10] G. Chourdakis, K. Davis, B. Rodenberg, M. Schulte, F. Simonis, B. Uekermann, G. Abrams, H. Bungartz, L. Cheung Yau, I. Desai, K. Eder, R. Hertrich, F. Lindner, A. Rusch, D. Sashko, D. Schneider, A. Totounferoush, D. Volland, P. Vollmer, and O. Koseomur, "preCICE v2: A sustainable and

user-friendly coupling library [version 2; peer review: 2 approved]," *Open Research Europe*, **2**, *51* (2022); 10.12688/openreseurope.14445.2., URL https://doi.org/10.12688/openreseurope.14445.2.

[11] A. Schaffrath, M. Sonnenkalb, and A. Wielenberg, "GRS Code System AC2," *Kerntechnik*, **84**, *5*, 356 (2019); doi:10.3139/124.019051., URL https://doi.org/10.3139/124.019051.

[12] "Systems and software engineering – Software life cycle," Standard, International Organization for Standardization, Geneva, CH (2017).

[13] R. S. Pressman, *Software Engineering: a Practitioner's approach*, McGraw-Hill (2001).

[14] "ANSYS CFX-Solver Theory Guide," , ANSYS, Inc. (2009).

[15] J. H. Ferziger and M. Perić, *Computational Methods for Fluid Dynamics, third, rev. edition*, Springer.

[16] R. Bonifetto, D. Caron, S. Dulla, V. Mascolino, P. Ravetto, L. Savoldi, D. Valerio, and R. Zanino, "Advances in the development of the code FRENETIC for the coupled dynamics of lead-cooled reactors," , CIRTEN: CERSE-POLITO RL 1572/2015 (2015).

[17] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen, *LAPACK Users' Guide*, Society for Industrial and Applied Mathematics, Philadelphia, PA (1999).

[18] J. P. Senecal and W. Ji, "Development of an Efficient Tightly Coupled Method for Multiphysics Reactor Transient Analysis," *Nuclear Energy*, **103** (2018); https://doi.org/10.1016/j.pnucene.2017.10.012.

[19] G. Grasso, "The core design of ALFRED, a demonstrator for the European lead-cooled reactors," *Nuclear Engineering and Design*, **278**, 2040 (2014); https://doi.org/10.1016/j.nucengdes.2014.07.032.

[20] M. Bandini, G. Polidori, "Report on the Results of Analysis of DEC Events for the ETDR (ALFRED)," Technical report, ENEA (2013).

[21] C. Ciurluini, V. Narcisi, F. Giannetti, L. Cretara, and G. Caruso, "Preliminary neutron kinetic – thermal hydraulic coupled analysis of the ALFRED reactor using PHISICS/RELAP5-3D," *Journal of Physics: Conference Series*, **1599**, *1*, 012023 (2020); 10.1088/1742-6596/1599/1/012023., URL https://dx.doi.org/10.1088/1742-6596/1599/1/012023.

[22] T. Rahm, J.-B. Droin, N. Marie, F. Bertrand, and A. Marrel, "Analysis of the Total Instantaneous Blockage accident consequences in the innovative inherently-safe CADOR SFR core," *Nuclear Engineering and Design*, **348**, 78 (2019); https://doi.org/10.1016/j.nucengdes.2019.04.019., URL https://www.sciencedirect.com/science/article/pii/S0029549318310471.

[23] T. Suzuki, X.-N. Chen, A. Rineiski, and W. Maschek, "Transient analyses for accelerator driven system PDS-XADS using the extended SIMMER-III code," *Nuclear Engineering and Design*, **235**, *24*, 2594 (2005); https://doi.org/10.1016/j.nucengdes.2005.06.012., URL https://www.sciencedirect.com/science/article/pii/S0029549305002335.