**FEniCSx-OpenMC Coupling for Neutronic Calculation with Temperature Feedback**

**Stefano Riva, Lorenzo Loi, Carolina Introini, Antonio Cammi**  
Politecnico di Milano, Dept. of Energy,  
CeSNEF-Nuclear Engineering Division, Nuclear Reactors Group  
Via La Masa, 34  
20156 Milano, Italy  
stefano.riva@polimi.it, lorenzo.loi@polimi.it, carolina.introini@polimi.it, antonio.cammi@polimi.it  

**Xiang Wang**  
College of Nuclear Science and Technology,  
Harbin Engineering University,  
Nantong Dajie 145, 31,  
Harbin, 150001, China  
xiang.wang@hrbeu.edu.cn

**ABSTRACT**

The state of an operating nuclear reactor depends on several interdependent physical phenomena, which can be considered simultaneously by modelling the system using a multi-physics (MP) approach. MP allows a higher level of detail of the system’s properties at the expense of code complexity and computational burden, whereas, in the past, single-physics codes were dominant due to the limited computational resources, and the coupling effects were typically introduced using correlations or boundary conditions to the problem. In the context of nuclear reactors, the fundamental coupling is between neutron physics and thermal hydraulics, as their interaction directly affects the power and temperature profiles, which are quantities of interest during both the design and safety analysis phases. Due to the recent improvements in computational capabilities, the MP approach has become feasible, focusing the efforts on the development of interfaces between different single-physics numerical codes (in most cases, already validated and verified) to catch the MP coupling. This work focuses on developing a tool capable of determining the temperature profile of a characteristic fuel pin of a PWR when the power generated by the system is known: this test case is interesting because the thermal feedback effects produce a shift in the power peak compared to the middle of the rod, which is a well-known phenomenon in nuclear reactor pins. This tool is developed in a Python environment, using the open-source library FEniCSx for the thermal-hydraulic analysis and the OpenMC Monte Carlo code to describe the fissionable system; this choice has been made to have the whole code inside a single open-source environment which, compared to state-of-the-art proprietary codes, offer higher accessibility and community feedback. In the coupling, an explicit method is applied whose convergence is based on a Picard scheme, using an adaptive relaxation scheme: this strategy is one of the most adopted techniques due to its simplicity; however, monolithic approaches can be also adopted which may give better results even though the implementation phase is more challenging. The proposed coupling approach can predict the peak shift in the power density as per the literature, thus more efforts can be made to extend the current model to more complex cases.
1 INTRODUCTION

Thermal reactors are the most widespread nuclear power generation technology in the world. In these reactors, moderation plays a crucial role in reducing the average energy of the neutrons emitted in the fission process, which leads to larger probabilities of fissile materials’ splitting and consequent heat generation [1]. In the case of Light Water Reactors (LWRs), the moderator coincides with the coolant fluid that extracts thermal energy from the core. As such, these systems present strongly coupled neutron physics (NP) and thermal-hydraulics (TH), as they influence each other through feedback effects. Indeed, the power distribution generated in the fissile materials from the neutron flux induces both in the fuel and in the surrounding materials temperature variations, which change the neutron-matter interaction properties. In particular, an increase in the fuel temperature directly influences the microscopic cross sections, increasing the absorption in the resonance zones (Doppler effect) with a consequent reduction in the number of neutrons available for fission events and therefore reducing the reactivity of the reactor [1]. In addition, the produced power impacts the liquid coolant behaviour: the density is affected by the temperature and consequently changes the macroscopic cross sections, which can be relevant, for example, in boron-controlled systems, where the local absorption in the water plays a significant role in the overall control strategy. Since the nuclear field has tight safety criteria when designing reactors, describing these systems requires high reliable and accurate models, and, because of the strong coupling between physics mentioned above, multi-physics (MP) model that simultaneously considers NP and TH effects are preferred.

In principle, a full MP model would require the development of a complete new solver implementing the solution of the Boltzmann transport equation [1] (or its approximation) and of a conjugate heat transfer (CHT) problem between solid and liquid regions. From a computational point of view, this task can be quite challenging: as the problem is strongly non-linear, in addition to its computational weight, numerical instabilities may arise during the solution process. However, in the past years several codes have been developed to solve single-physics problems: thus, it would be reasonable to keep the experience and the knowledge about them instead of discarding them in favour of brand-new MP codes (that are to be subjected to the process of verification and validation before they can be used in safety and regulating-related applications). This lead to a different framework to MP modelling: instead of following the monolithic approach previously presented, a segregated one can be followed: the coupling is obtained by building suitable interfaces between established single-physic codes, solving them separately in an iterative process known in literature as Picard iteration [2]. This approach is still challenging due to the computer issues when developing I/O interfaces between codes as explored in [3], however, it allows to keep the previous experience on single-physics codes. To facilitate writing and reading process between codes, this works aims at developing an enhanced MP-like neutronic code by taking into account temperature effects in the Python programming language, adopting two packages: the community-developed open-source Monte Carlo code OpenMC (v. 0.13.2) [4] for neutronics and the open-source Finite Element library FEniCSx (v. 0.5.2) for the thermal part. Having everything inside the same environment avoids the need of developing an interface between codes, allowing the use of Python capabilities for data-storage and post-process. As such, this paper proposes a coupling interface between OpenMC and FEniCSx in order to improve the results of the neutronic code by taking into account temperature effects, showing how these two solvers can easily exchange information between themselves to make more accurate neutronic predictions compared to the de-coupled case whilst at the same time avoiding the complexity related to implementing and solving a CHT solver for the thermal part.

The proposed approach is tested on a simple but significant numerical case, namely a reactor pin of a PWR reactor. In particular, the shift of the power and temperature peak will be observed as a result of the feedback effects [5], compared to the single-physics NP case and the results of [6]. This work is organised as follows: Section 2 reports the methodology,
the physical models and the coupling algorithms, with their underlying hypothesis; Section 3 is devoted to the numerical results; in the end, Section 4 acts as a conclusion.

2 METHODOLOGY

This section briefly presents the physical models adopted in this work: first, the stand-alone models are discussed; then, the strategy for the coupling will be presented.

2.1 Neutronics

The transport of neutrons inside a multiplying medium is described by the steady Boltzmann equation [1], reported as a balance between the terms of leakage (L), scattering (S) and fission (F) that can be written as a generalised eigenvalue problem:

\[ [L_{N,T} - S_{N,T}] \Phi(s) = \frac{1}{k_{\text{eff}}} F_{N,T} \Phi(s), \]  

where \( k_{\text{eff}} \) is the fundamental eigenvalue, \( \Phi(s) \) is the angular neutron flux evaluated in the phase space \( s = (x, \Omega, E, t) \), \( N \) and \( T \) are the nuclide and temperature field, respectively. In this framework, the neutron transport equation will be solved through the Monte Carlo code OpenMC adopting the \( k \)-eigenvalue method. Within a Monte Carlo code, the transport equation (1) is solved by sampling neutrons as particles: in particular, by the extraction of random numbers, it is possible to follow the path of the particles in the system from their birth to their disappearance. By iterating this process a large number of times, figures of merit that represent the system’s average behaviour are collected. At this moment OpenMC is able to assign a constant temperature value to each material in the geometry, thus it cannot treat temperature distributions within its built-in capacities. To achieve this goal auxiliary materials should be defined and to each of them an average temperature must be imposed [7] to simulate a spatial temperature distribution; for the time-evolution of temperature, coupling with a thermal-hydraulic code is needed.

Regardless of the treatment of the cross-sections, the figure of merit necessary for a coupling calculation is the volumetric power density \( q'''' \left( \frac{W}{cm^3} \right) \), which serves as the input for the thermal problem (see Section 2.2). The power density is estimated at the end of the transport simulation, based on the energy deposition score that occurs in the system during the Monte Carlo routine. For coupling, a proper normalisation with respect to the total power \( Q \left( \frac{W}{cm} \right) \) is also required.

2.2 Thermal diffusion

The neutronics model described above must be coupled with a thermal model to treat space and time temperature distributions and thus to study their effects on cross sections and density. To avoid solving the conjugate heat transfer problem between solid and fluid, a simplified approach has been adopted: the fluid is modelled using a 1D energy balance, whereas the temperature in the solid regions is obtained from the steady heat diffusion equation. Despite being a strong simplification, this approach follows the philosophy of this work, that is, building a self-contained, single-language open-source tool to improve the neutronic calculation compared to a pure single-physics analysis.

As such, in this simplified framework the bulk temperature of the coolant \( T_b \) is estimated through an energy balance as a function of the axial position \( z \):

\[ T_b(z) = T_{in} + \frac{1}{mc_p} \int_{-L/2}^{+L/2} q'(z)dz, \]  

Proceedings of the International Conference Nuclear Energy for New Europe, Portorož, Slovenia, September 11–14, 2023
given \( q' \) the linear power, which is an input coming from OpenMC, \( T_{in} \) the inlet temperature, \( \dot{m} \) the mass flow rate, \( c_p \) the specific heat capacity and \( L \) the total length of the pin. The bulk temperature is then used to impose a convective heat transfer boundary condition (Robin type), together with a proper heat transfer coefficient \( h \) computed by means of a semi-empirical correlation. The temperature in the solid regions \( \Omega_s \) is governed by the steady thermal diffusion equation:

\[
-\nabla \cdot (\lambda \nabla T) = q''' \quad \text{in } \Omega_s \quad \text{with} \quad \left\{ \begin{array}{l}
-\lambda \nabla T \cdot \mathbf{n} = h(T - T_b(z)) \quad \text{on } \Gamma_c \\
-\lambda \nabla T \cdot \mathbf{n} = 0 \quad \text{on } \partial \Omega_s \setminus \Gamma_c
\end{array} \right. \tag{3}
\]

given \( \Gamma_c \) the boundary interface in contact with the coolant, \( \lambda \) the thermal conductivity and \( q''' \) the power density, acting only in the fuel where the fission events occur.

### 2.3 Stationary Algorithm

Now that the stand-alone models have been presented, the coupling strategy will be discussed, referring to the case in which the fuel composition is fixed (hence the moniker Stationary Algorithm). The stationary neutron-thermal coupling can be seen in terms of the successive solution of two fundamental equations, using a Picard/fixed-point iteration scheme, where the operators \( (L, S \text{ and } F) \) in Equation (1) are influenced by the system temperature \( T \). In particular, a change in temperature affects both the microscopic cross sections by Doppler effect and the macroscopic cross sections by moderator expansion; on the other hand, the volumetric power density \( q'''(\mathbf{x}) \) is the source term for the thermal diffusion equation (3).

#### 2.3.1 Treatment of the thermal feedback

As already mentioned, the thermal feedback on the cross-section is quite important, making the mapping of the temperature inside the solid materials an important concern. To deal with this request, the most accurate method falls back to the Delta Tracking algorithm, which handles particle transport in media with continuously varying properties by performing a homogenisation of cross sections. Unfortunately, at the present time, OpenMC has not yet released a version with the ability to implement Delta Tracking; as the source code of OpenMC is freely available, this could be theoretically done, nevertheless its practical implementation is a challenging task and out of the scope of this work. The alternative to keep the implementation as easy as possible, as proposed in [7], is to discretise the geometry into zones at fixed temperatures, to have a system with overall constant properties at intervals. In particular, the fuel and the coolant domain is divided into axial regions, in which a regional average temperature is imposed.

![Figure 1: Portion of the physical domain with the different regions (water, cladding, helium and fuel) and the axial division idea for fuel and coolant.](image)

The spatial domain \( \Omega \subset \mathbb{R}^3 \) is usually composed of different materials \( N_{mat} \) (i.e., fuel, cladding, gap, coolant, see Figure 1). From the point of view of the neutronic solver, the fuel...

---

1. In this work, the Dittus-Boelter correlation is used [8].
2. The effect of \( \gamma \)-heating is neglected.
and the coolant are further subdivided into \( L \) axial zones, each labelled as \( \{ \Omega_l \}_{l=1}^L \), to impose the temperature due to the importance of the Doppler and density effects in these materials; whereas, the other materials are considered at uniform temperature because their overall impact on the neutron transport is negligible with respect to the feedbacks presented above. Furthermore, this analysis divides the fuel and the coolant only axially, averaging the temperature along the radial coordinate, thus considering the gradient down the principal coordinate \( z \) which has greater importance with respect to the radial one; moreover, this choice is coherent with respect to the axial treatment of the coolant energy balance. Therefore, each subdomain \( \Omega_l \) will be referred to as region, onto which an average temperature \( T_l \) is imposed. This modelling choice is a direct consequence of OpenMC inability to treat position-dependent properties. OpenMC can provide as output the power density dependent on the 3D position \( x \); however, since the temperature that OpenMC sees is only axial-dependent for the reasons explained above, it has been chosen to deal only with the axial dependency on \( z \) of the power density. The geometry can be better represented using cylindrical coordinate and since the power density and the bulk temperature of the fluid are only axial dependent, Equation (3) is solved considering only the radial \( r \) and axial \( z \) dependency.

**Algorithm 1: Coupled solver OpenMC-FEniCSx**

**Input**
- Guess Temperature \( T_{i,(0)} \);
- Power \( Q \);
- Number of axial divisions \( L \);

**Output**
- Temperature \( T(r, z) \) and Axial power density \( q''(z) \)

**Initialisation**
\( j = 0, \text{ error } \varepsilon = tol + 1 \);

**while** (\( j < \text{iterMax} \) & \( \varepsilon > tol \)) **do**

\( j = j + 1 \);

\( q''_{(j)}(z) = \text{OpenMC}(T_{(j-1)}(r, z)) \);

\( T_b(z) = \text{EnergyBalance}(q''_{(j)}(z)) \);

\( T_{(j)}(r, z) = \text{FEniCSx}(q''_{(j)}(z), T_b(z)) \);

\( T_{l,(j)} = \text{Average}(T_{(j)}(r, z), l) \);

**if** \( j > 1 \) **then**

\( \varepsilon = \| q''_{(j)} - q''_{(j-1)} \| \).

**2.3.2 Convergence criterion**

The scheme for the stationary case is summarised in Algorithm 1, and it is based on a fixed-point iteration: at first, some guess temperatures are provided to start the loop; OpenMC is run and the power density \( q''(z) \) is obtained, which is then used as an input to compute the bulk temperature of the fluid by means of the energy balance (2); lastly, the thermal diffusion (3) is solved with FEniCSx, using the bulk temperature to assign the Robin boundary conditions. These steps are repeated until convergence is reached. The convergence check is made on the axial power density measured as relative difference in \( L^2 \)-norm. The output of each iteration \( j \) consists of the power density and the temperature field, from which the regional average temperatures \( T_{l,(j)} \) are computed. The fixed-point method used in this work is typically conditionally stable from the numerical point of view, but instabilities may arise when the quantity computed at iteration \( j \) is much different from the one at \( j - 1 \); therefore, the coupling quantity
\(q'''\) for this problem is under-relaxed with a relaxation factor computed through the stochastic approximation algorithm as in [11].

3 NUMERICAL RESULTS

The algorithm described in the previous section will be applied to a PWR reactor pin, surrounded by water. As reported in Figure 1, different materials are used in the numerical model: the fuel is Uranium dioxide (enriched with \(^{235}\text{U} 4\%\text{wt}\)), the cladding is composed of an alloy of Aluminium, the gap is filled with helium gas whereas the coolant is pressurised water. The total power of the pin is 30 kW.

![Graphs comparing axial power density and temperature distributions](image)

Figure 2: Comparison of the normalised axial power density \(q'''\), the regional fuel temperature \(T_f(z)\) and water temperature \(T_w(z)\) assigned to regions \(\Omega_i\) with respect to the ground-truth with 256 divisions and the results of [6].

Since the effect of the water density on the macroscopic cross sections cannot be neglected, its dependence on the temperature must be taken into account. The working pressure of the system is near 155 bar, thus the density as a function of the temperature can be extracted with a linear fitting using the \texttt{pyXSteam} library for Python in the temperature range \([553.15, 603.15]\) (K). This section analyses the effect of the number of axial divisions to the axial power and the temperature distribution. Three different values of \(L = [4, 16, 128]\) are considered to see how this affects the results\(^3\): the results are compared with the best-solution obtained with the present solver, a case with 256 axial divisions, and the results of [6]; for the latter,

\(^3\)The increase of the axial regions makes the predictions more accurate whilst increasing the computational costs. This falls into the trade-off between accuracy and computational complexity: the selected values are the best compromise.
results have been obtained with the power equal to 65 kW, thus a normalisation of the output was carried out to properly compare the two cases. The simulations have been performed on a 8-core Intel Xeon (max clock speed 2.83 GHz) mounted on a workstation.

Figure 2 shows a comparison between the axial power density and the temperature (radially averaged) for different values of divisions with respect to the case with 256 divisions, the de-coupled case and the reference data [6]. The temperature feedback effects provide a shift of the power (and consequently temperature) peak to lower axial positions as observed in the complete Serpent-OpenFOAM MP model [6] (the difference in the fuel temperature can be related to the different power level, because of the different impact of the Doppler effect). Indeed, in the lower part of the pin the density of the coolant is higher, hence neutrons are more effectively moderated, thus increasing the effective cross section of fission reactions resulting in a higher power, which provides a shifted temperature profile [6]. In the end, even the case with 4 axial divisions is able to predict a power distribution very similar to the case with the maximum number of divisions, showing the fact that even a small information about the temperature distribution is enough to obtain a physical power distribution.

![Contour plot of the temperature field](image)

(a) OpenMC-FEniCSx results  
(b) Results from [6]

Figure 3: Contour plot of the temperature field with \( L = 256 \) compared to the de-coupled case and the results from [6] (figures not in scale).

Then, the spatial temperature distribution is reported in Figure 3 considering 256 axial divisions in the fuel and in the coolant. The peak of the temperature is shifted from the centre to the lower part due to the axial distribution of the coolant density.

In the end, for what concerns the computational costs, the main bottleneck is the Monte-Carlo simulation, since the numerical solution of the thermal part is almost instantaneous. As the number of axial divisions increases the time required to solve the \( k \)-eigenvalue problem with OpenMC increases as well because the number of materials in the geometry is proportional to the divisions making the simulation heavier and heavier. The CPU time for the case with 256 divisions is 1.4 min, whereas for all the others is lower. Nevertheless, this procedure is much cheaper than the coupling between OpenFOAM and Serpent proposed in [6].

4 CONCLUSIONS

This work presents a coupling strategy between neutron physics and thermal-hydraulics to obtain improved neutronic results, by taking into account the main temperature feedback effects on the microscopic cross sections through the Doppler effect and on the macroscopic ones due to the variation in the density (especially for the coolant). This paper adopts the Python-based neutronic MonteCarlo code OpenMC and Finite Element library FEniCSx, in order

---

4 Each variable has been re-scaled with a min-max normalisation, since the results from [6] have been obtained for 65 kW.
to have everything inside the same environment. By the time this work is being written, the former is not able to treat materials with continuously varying properties. In order to overcome this, the system is divided into regions onto which a constant temperature is imposed. In this work, only axial divisions are considered. This approach has been tested on a reactor pin from a Pressurised Water Reactor of 30 kW. It is well known in literature that the main effect of the temperature feedbacks consists in the shift to the lower part of the axial distribution of the power density and the numerical results confirm this.

In the future, it is foreseen to extend this approach to more complex cases, introducing also radial divisions to take into account a more realistic temperature distribution in the matter. Moreover, this coupling strategy will be applied to burnup calculations including temperature effects as the fuel is burnt.

REFERENCES


