TRIGA FUEL ELEMENT SIMULATION USING OPENFOAM

V.V.A. SILVA, A.A.C. SANTOS, P.S.B.L. DA SILVA, A.Z. MESQUITA
{vitors, aacs, psbls, amir}@cdtn.br
Serviço de Tecnologia de Reatores, CDTN/CNEN
Av. Antônio Carlos 6627, CEP 31270-901, Belo Horizonte, MG - Brazil

C.PEREIRA
claubia@nuclear.ufmg.br
Departamento de Engenharia Nuclear, UFMG
Av. Antônio Carlos 6627, CEP 31270-901, Belo Horizonte, MG - Brazil

ABSTRACT

Computational fluid dynamics (CFD) codes have been extensively used in engineering problems, with increasing use in nuclear engineering. One of these computer codes is OpenFOAM. It is freely distributed with source code and offers a great flexibility in simulating particular conditions like those found in many problems in nuclear reactor analysis. In order to simulate the IPR-R1 TRIGA Mark I reactor located at the Centro de Desenvolvimento da Tecnologia Nuclear, in Belo Horizonte, Brazil, it is important to accurately describe the fuel element thermal profile. Due to the particular cosinusoidal shape of TRIGA fuel power distribution, a specific boundary condition had to be implemented. Based in boundary conditions classes provided by OpenFOAM, a new class was developed taking in account not only the power profile of the TRIGA fuel but also implementing the heat transfer accurately. In this work the details of this implementation are presented with examples of use and results of a simple heat transfer case simulation.

1. Introduction

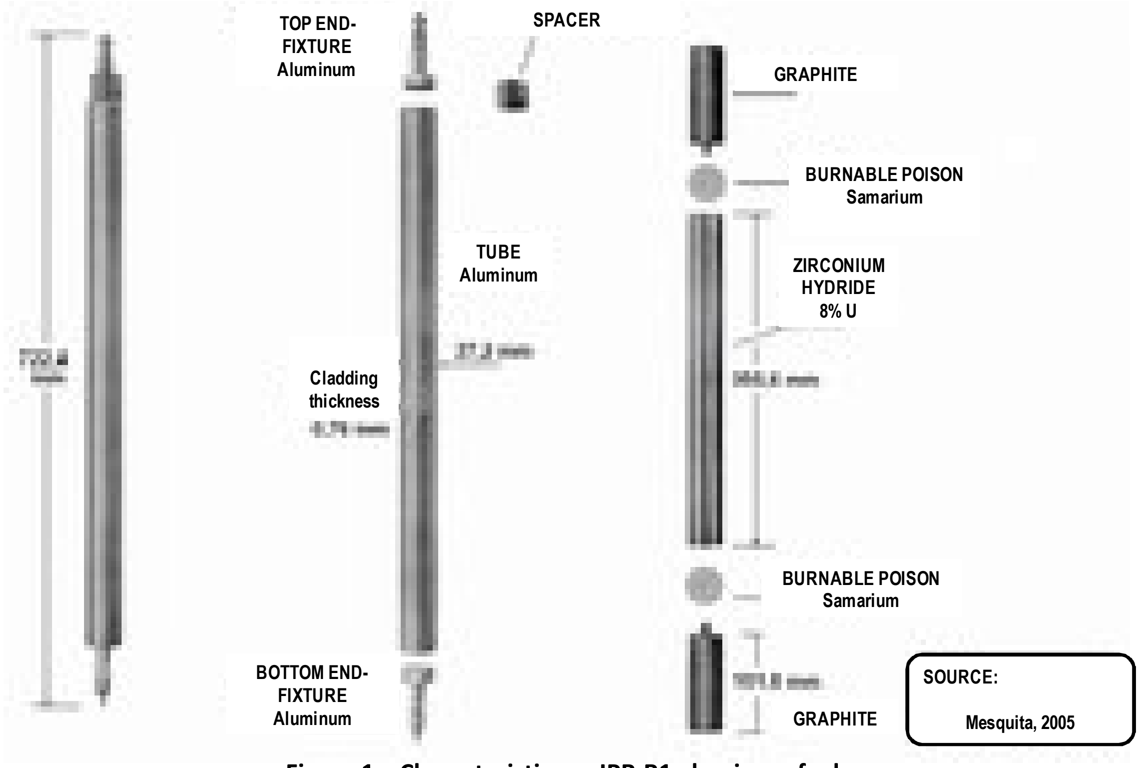
The TRIGA IPR-R1 Reactor is in operation since 1961 at the Centro de Desenvolvimento da Tecnologia Nuclear (CDTN). During this time, many data about its operation was collected. This valuable information provides a fundamental basis for any computer simulation, allowing the validation of the results obtained from computer codes. In order to simulate the TRIGA reactor operating under different conditions, the first step is to simulate it at a well experimentally investigated operating condition and compare the results with the experimental data collected at the same operation. However, many simulation aspects can make the results diverge from real data. To improve the accuracy of the simulation, it is obvious that the modelled reactor must behave as close as possible of its real counterpart.

The software used to perform the simulation is a free-distributed Computational Fluid Dynamics code called OpenFOAM [1]. OpenFOAM offers very flexible platform to perform fluid dynamics simulations. However, this flexibility does not come for free. To be able to change some very specific aspects of the simulation, the user must write its own code using the extensive set of classes provided by OpenFOAM [2].

That said, a key element in this case is the TRIGA fuel behavior. The cosinusoidal axial power distribution of TRIGA fuel is well known [3]. This work describes how this power shape is achieved by means of the development of a new boundary condition class implementing the heat transfer from the TRIGA fuel to the coolant.

2. Development

The IPR-R1 TRIGA Reactor is filled with 63 fuel elements, 59 of them of aluminum cladding and 4 of stainless steel cladding. In the Figure 1 [3] there is a description of different characteristics of aluminum fuel. The stainless steel fuel is slightly different in size and with some changes in geometry, for example the spacer. However, at this point of the work, only the aluminum fuel is being modeled. Moreover, the main difference from both fuels, the thermal properties of the cladding, is not yet being modeled. This conjugate heat transfer simulation in envisaged in the future works.



SOURCE: Mesquita, 2005

Figure 1 – Characteristics os IPR-R1 aluminum fuel.

The focus in this work is to be able define a heat flux as similar as possible to the fuel. Its power distribution along the length of the fuel can be described as a cosinusoidal (or sinusoidal) function as shown in Figure 2.

The superior graph shown in Fig. 2 shows the power distribution along the active fuel surface L_0 and the ℓ on both extremities shows the extrapolation of flux which covers part of the non-active portion of graphite of the fuel. The second bar below the graph shows the real fuel rod and how should the power distribution match to it.

In the present case, an extreme simplification was done: the distribution is applied towards the fuel length and this distribution is then truncated to zero under both ℓ portions. It means that both areas in the boundaries have zero flux.

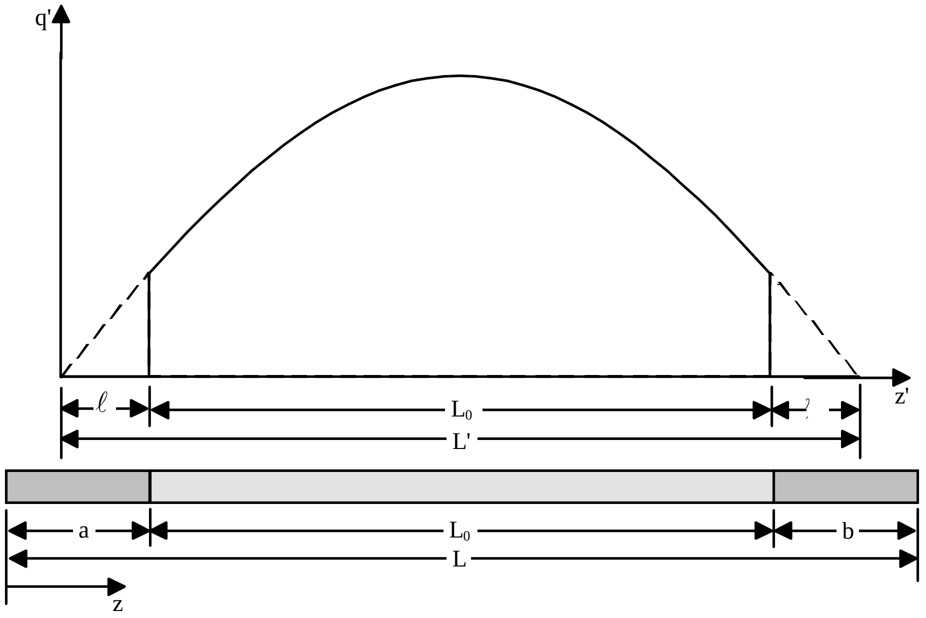


Figure 2 – Sinusoidal power distribution along the fuel element.

This coarse simplification introduces an error between the nominal power and the power really being applied. Analysis of this difference is performed in next sessions.

2.1 Implementation

The implementation is based in an already existent OpenFOAMâs class: `turbulentHeatFluxTemperatureFvPatchScalarField`. This class implements a fixed heat boundary condition to specify temperature gradient. The input heat source can be either specified in terms of an absolute power [`W`] or as a flux [`W/m`^{2}]. The key element in the new `TRIGAHeatFluxTemperature` boundary condition is the sinusoidal shape along the fuel and, in order to change this power shape, three main changes were made in the reference OpenFOAM class:

1) In the `TRIGAfvPatchField` constructor which takes a finite volume patch, a dimensioned field and a dictionary parameters. By taking all face centers in the fuel patch, the longest dimension (x, y or z) is taking after calculated the differences between the minimum and maximum points in Cartesian coordinates. Then, a new scalar field is created with the same size of the fuel patch. For all faces belonging to this patch the new correspondent face value is calculated and attributed to the face by means of the function:

$$face = q\_\frac{\pi}{2}\sin\left(\frac{face_i - \min x}{x}\pi\right) \tag{1}$$

were, $face$ is the scalar value at the face, $q\_$ is the constant gradient value given by the user, $face_i$ is the position of the face in the axis, $\min x$ the minimum coordinate value of the axis and x the axis size.

In the boundaries, the values are set to zero if the face position is approximately 7.74% of the total axis size. This is due to not having a heat flux gradient at the inactive parts of the fuel. Remembering this is not completely accurate since there are is a small heat flux gradient of entering the non-active part of the fuel (Figure 2).

In order to be able to attribute to the class gradient the values previously calculated and stored in a scalar field, a dynamic cast must be performed. The source code to perform this action is shown below:

```
scalarField& oPatch = dynamic_cast<scalarField&>(*this);
oPatch = (*patchField);
q_ = oPatch;
```

2) The second change is in the TRIGAfvPatchField::updateCoeffs()method. The specific water density (*rhow*) is read in the transport dictionary and added to the term in the denominator which calculates the surface gradient.

3) The values of specific water density (*rhow*) and specific heat capacity for the water (*Cp0*) were added to the RASProperties file in the simulation directory. These variables give the water properties necessary to the turbulence model and the boundary condition to calculate the heat flux.

2.2 Testing

Ideally, after modeling and/or simulating a problem it must be *verified* and *validated.* Validation consists in confirming if the values obtained are representative of the real world. Verification is to confirm if the model and numerical solution work according the proposed [5]. That said, the first step to test the boundary condition, is to simulate a simple case and verify the results. OpenFOAM setup for boundary conditions consists in defining them in files corresponding to each variable of the solver. Here are presented the boundary conditions for the temperature using the TRIGAHeatFluxTemperature boundary condition. In the file T in the simulation directory 0, this boundary condition is applied to patches representing the fuel wall, or, in other words, the wall where the heat flux will be defined.

```
boundaryField
{
  fuel_c4
    {
        type TRIGAheatFluxTemperature;
        heatSource power;
        q uniform 1000.0;
        alphaEff kappaEff;
        Cp Cp0;
        value uniform 300;
    }
  ...
```

The first line shows that this is the set of boundary fields to be defined. The `fuel_c4` is the name of the patch and after it the patch details. The type is the boundary condition to be used, than the source characteristics, the power field in [`W`], the name of turbulent thermal diffusivity field, the specific heat capacity variable and the default value for the temperature field. Details on how to use this boundary condition are in comments in the source code.

The basic test to guarantee the correct behavior of the `TRIGAHeatFluxTemperature` is verifying the energy and mass balancing in a simulation. This test was performed in a geometry modeling a sub-channel of the TRIGA IPR-R1 reactor showed in Figure 3.

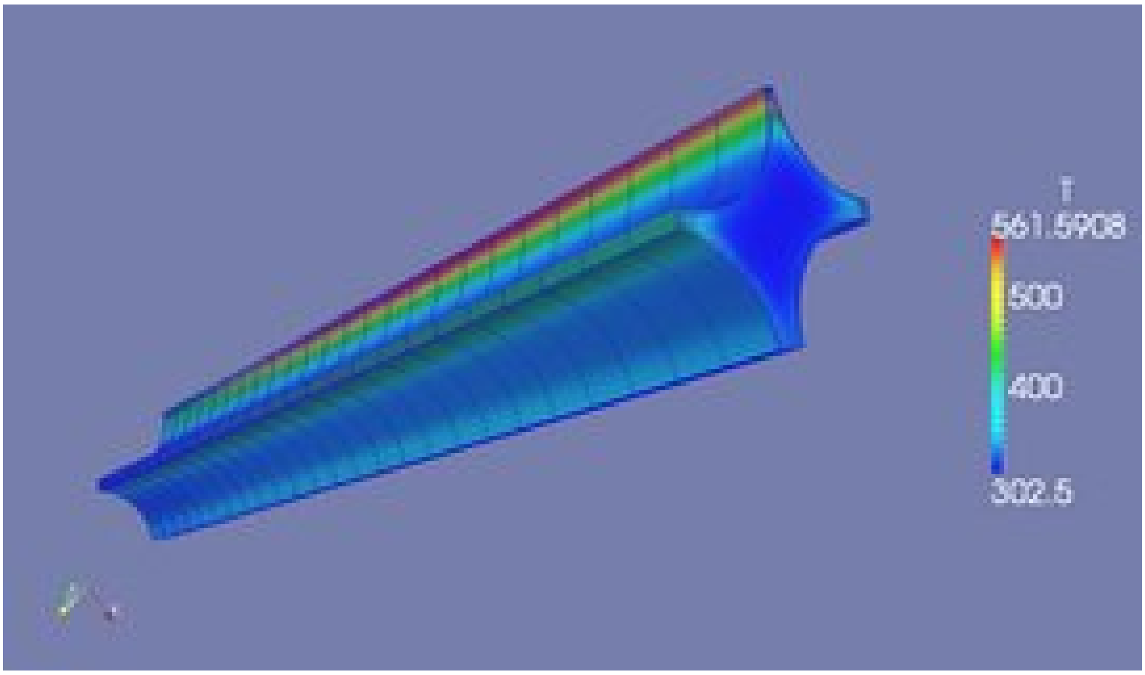


Figure 3 – Modelled subchannel of TRIGA IPR-R1 reactor.

The variation of the mass flow averaged value of temperature of the water flow entering and leaving the subchannel was used to verify if the energy added to the system. In order to be able to apply it, the system must have reached the steady-state. In short, the system must respect the conservation of energy [6]. This is verified by means of the equation:

$$T_{outlet} = \frac{q}{vA\rho Cp} - T_{inlet} \qquad (2)$$

Were the temperature at the outlet [T_{outlet}] must be equal to the heat transferred [q] over fluid velocity [v] times flow area [A], times fluid density [ρ] times fluid specific heat [Cp] minus the temperature at the inlet [T_{inlet}].

The results for the simulation shown in Figure 3 are presented in Table 1.

Table 1 – Results for the energy conservation of the modelled system.

Theoretical values	
Ro	997.2
Cp	4178
Velocity	0.09
T inlet	302.5
Power	4000
theoretical DT	18.968

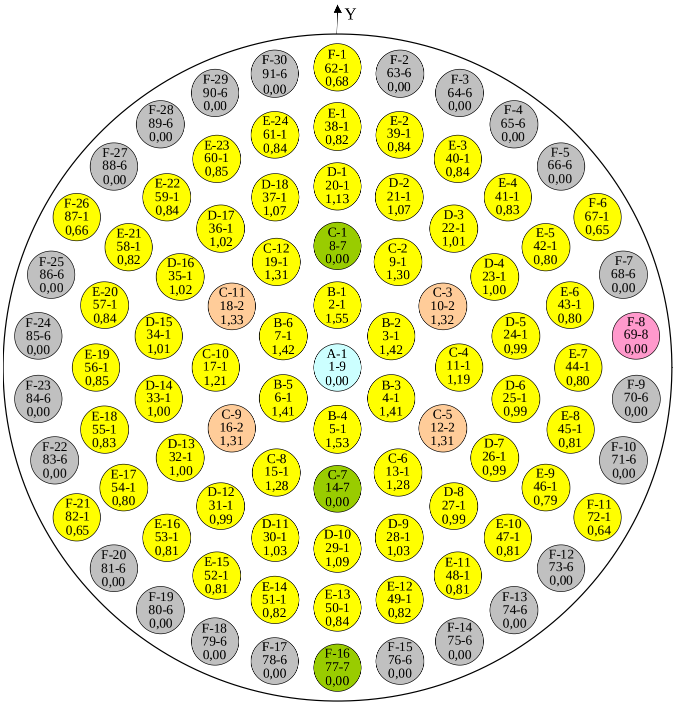
Simulated data	
Mean T outlet	320.597
A	0.0005624
DT	18.097
m	0.05
final m	0.05
Q	3816.31

Difference	
ΔQ%	4.59
ΔT%	4.59

The difference from theoretical calculations to the simulation for the heat transferred is about 4.59%. This loss is due to the oversimplification in the power profile implementation for the fuel extremities.

3. Results

The new boundary condition was tested using the `buoyantBoussinesqPimpleFoam` solver. This is a transient solver for buoyant, turbulent flow of incompressible fluids. The mesh represents the core of IPR-R1 TRIGA reactor, with differences among graphite rods, control rods, fuel and neutron source. The scheme of reactor core is presented in Figure 4.



Kind of element	Description
🟡	Aluminum fuel rod
🟠	Stainless steel fuel rod
⚪	Graphite rod
🟢	Control rod
🟣	Neutron source rod
🔵	Central tube

Source: Veloso 2005

Figure 4 – Element distribution of TRIGA IPR-R1 reactor.

The values in each rod shown in Figure 4 are, in top-bottom order: annulus-position identification, core number-type of fuel (1 aluminum ï 2 stainless steel) and the radial power factor. The radial power factor was obtained from former calculations [2] and is not considered in the test simulations.

In order to verify the natural convection, the inlet velocity for water in the core was set initially to zero and 250kW of power was defined for the whole core.

It is important to remark that there is no simulation of conjugate heat transfer. Due to this fact, a surface heat flux was defined at the fuel elements, neglecting heat conduction effects. This allows the temperature to reach values which are not the same as the real reactor. Moreover, due to the characteristics of the solver, there is no water boiling. The solver is limited to one phase.

The left side of Figure 5 shows the set of fuel rods and the power profile due to the heat flux provided by the `TRIGAHeatFluxTemperature` and in the right side, an upper view from the mesh used in the core simulation. The thin fluid connections separating fuel rods can be seen in yellow and green, showing the highest fluid temperatures in these areas.

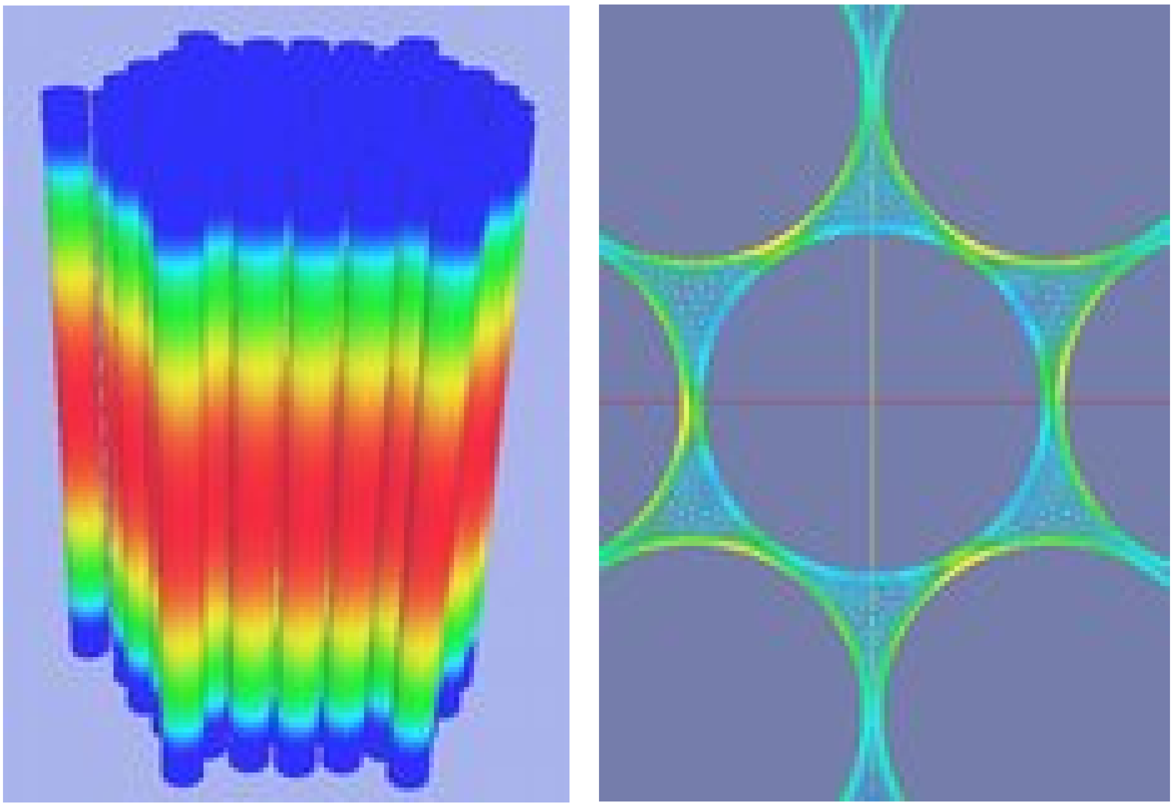


Figure 5 – Simulation power profile (Left side) and top view of the used mesh (right side). The colors show, from blue to red, the the incresing flux from fuel extremities to center.

A screenshot from the full core simulation is show in Figure 6. In this image, the simulation is still is course and the steady-state was not yet reached. The temperatures are in Kelvin.

For all simulations, the mass-energy balance was applied to assure the calculations are mathematically correct.

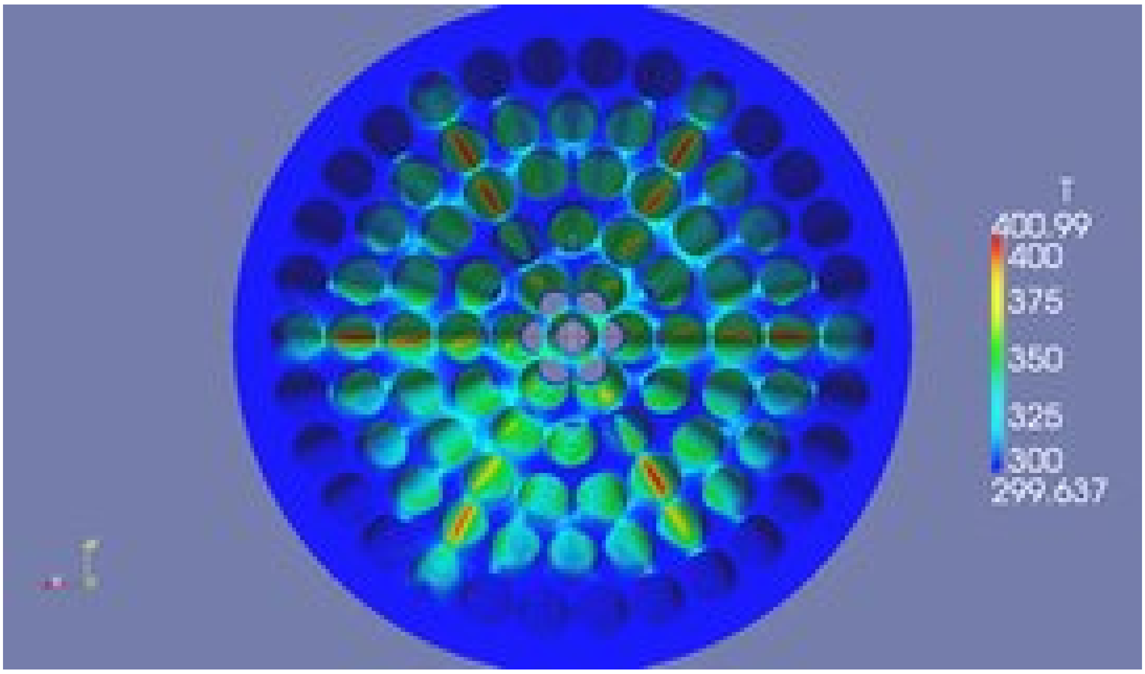


Figure 6 – Temperature distribution of water flowing through TRIGA IPR-R1 core.

4. Conclusion

The new `TRIGAHeatFluxTemperature` boundary condition is fully functional and provides a heat flux in sinusoidal shape like the TRIGA fuel with only 4.6% error to the nominal power. Simulations in a sub-channel configuration and the full core were carried. However, since there is no conjugate heat transfer implementation and sub-cooled boiling, the water in the simulated core can reach much higher temperatures than in the IPR-R1 TRIGA.

5. Future work

A simulation with a different solver (namely chtMultiRegion) is envisaged in order to fully simulate the TRIGA IPR-R1 core and sub-channels. The simulation will consider the material compositions in the fuel: cladding (Al 1100-F), fuel (U-$ZrH_{1,0}$) and graphite. The results of this simulation will be compared to the experimental data collected during reactorȇs operation in different powers.

This implementation is also a first approach to boundary conditions implementation in OpenFOAM. A next step is to implement more detailed heat flux considering a simplified neutronics modeling of the fuel.

6. Aknowledgments

The authors would like to thank AIEA for the grant to attend the Conference and the Department of Nuclear Engineering and Engineering School of UFMG for paying RRFM 2013 conference registration fees.

7. References

[1] JASAK, H., JEMCOV, A. and TUKOVIC, Z. **OpenFOAM: A C++ Library for Complex Physics Simulations** International Workshop on Coupled Methods in Numerical Dynamics, 2007.

[2] **OpenFOAM C++ Documentation.** *openfoam.org***.** n.p. 2013. Mars 21, 2013.

[3] VELOSO, M. A. **Avaliação Termo-Hidráulica do reator TRIGA IPR-R1 a 250kW.** CDTN/CNEN, Belo Horizonte, 2005 (NI-EC3-05/05).

[4] MESQUITA, A. Z. **Investigação experimental da distribuição de temperaturas no reator nuclear de pesquisa TRIGA IPR-R1**. Tese de doutorado apresentada à Faculdade de Engenharia Química da Universidade Estadual de Campinasò, Campinas, SP, 2005.

[5] JANOSY, J. S. **Simulation and Simulators for Nuclear Power Generation**, Nuclear Power - System Simulations and Operation, Dr. Pavel Tsvetkov (Ed.), ISBN: 978-953-307-506-8, InTech, DOI: 10.5772/18219. 2011. Available from: http://www.intechopen.com/books/nuclear-power-system-simulations-and-operation/simulation-and-simulators-for-nuclear-power-generation

[6] WARK, K. Jr. and RICHARDS, D. E. **Thermodynamics**. Sixth Edition. WCB/McGraw-Hill, 1999.