A MULTIPHYSICS APPROACH TO LFR ANALYSIS

Marco Sumini, Sandro Manservisi, Marco Lanconelli

University of Bologna, DIN, Lab. of Montecuccolino, Italy and CIRTEN

Technical Meeting on the Benefits and Challenges of Fast Reactors of the SMR Type, Milan, Sept. 26th, 2019
Key-words

- Open-source
- Multiphysics, coupling different aspects
- Wrapping unifying environment
- Allowing the management of large amount of data
Introduction of multiphysics approach

Development of computational platform that integrates different computational tools, into the common wrapping framework given by the **SALOMÉ platform**:

- Lattice code, **DRAGON**
- Full core simulation code, **DONJON**
- 3D-porous thermal-hydraulic CFD code, **FEMus**
Targets of multiphysics approach

- Investigate the coupling between neutronic code (at lattice level and full core model) and a thermal-hydraulic CFD code
- Extended use of Hyerarchical Data Format structures (HDF5)
- Implementation of multiphysics approach into preliminary model of a Fast Small Modular Reactor (SMR)
- Choosing as case study a lead cooled reactor conceptual design, ALFRED (that can be qualified as SMR), characterized by small axial and large radial core dimensions

Perspectives

A coupled neutronics and thermal hydraulics analysis to better supporting the fast core behavior (in both fields) at design level
The neutronic code are the open-source reactor physics codes Dragon and Donjon Version 5 developed at Ecole Polytechnique de Montréal (Canada), designed around methods to solving transport equation of neutron and, in Version 5, explicitly modified for Salomé compliance. Both codes are able to tackle hexagonal geometries and fast cores from Version 4, f.i. used in ASTRID modeling benchmarks (2013)

**Lattice code, DRAGON**
Simulate the neutronic behavior of the fuel assembly in the reactor core and create the proper nuclear database, called MULTICOMPO (that inherited also the role of the conventional SAPHYB data files)

**Full core simulation, DONJON**
Interpolate information contained in MULTICOMPO, simulate the comprehensive neutronic behavior of core and compute power and flux distribution
Nuclear Database - MULTICOMPO

MULTICOMPO

- **data structure** which contain nuclear properties of assembly’s materials.
- possibility to recovery data concerned parameters as *nuclide’s temperature and density or depletion calculation*.
- necessity to create an optimal nuclear database for a correctly use of full core simulation code.

1. Isotopic cross-section library: for this specific example on ALFRED JEFF 3.1.2 SHEM DRAGLIB with 315 energy groups (as reworked as group structure by Santamarina, Hebert and Hfayed).
2. Lattice calculation
   - Burn-up calculation
   - Temperature and density modification
   - Homogenization and collapse to 33 energy groups
3. Saving macroscopic cross sections in the MULTICOMPO structure
FEMus is an open-source library, developed at the University of Bologna, based on the LIBMESH library with parallel MPI-PETSC (i.e. with improved scalability) and multigrid solvers, recently extended to hexagonal geometries.

For details or if you are willing to contribute, see: https://github.com/FemusPlatform/femus

or ask to: sandro.manservisi@unibo.it
The CFD solvers have the function to solve energy and temperature equations starting from a given neutronic power density distribution.

Assumption for solution of energy equation (only) is porous medium.

Reactor channel (closed core)
The main purpose

is to obtain the solution of a multi-physics and multi-scale three-dimensional problem inside a simplified and more comprehensive framework.

The Salomé project

facilitates the coupling of scientific mesh-based codes thanks to its architecture and suite of tools that provide several data interfaces and exchange across the different codes.

The integration of a code on the Salomè platform is obtained by generating an interface with functions available in the MEDMem library that allows a data transfer from the platform to the code and then from the code to the interface.
Main SALOME’ modules used in wrapping

**MED (Modelisation et Echanges de Données) module**
provide a library for **storing** and **recovering** computer data in a suitable format

**GEOM and MESH modules**
have the main function to **draw** and **create meshes** in MED format, respectively.

**YACS module**
is a tool to **supervise execution** of complex interconnected scientific applications as object structure available during execution of DONJON code.
Computational platform

Open source codes
Computational platform
Proprietary codes
Summary flowchart

FEMUS

Full Core Simulation Donjon

Lattice Code Dragon

Approach Multiphysics Numerical Platform Loop

Evaluation new fuel, coolant temperature and coolant density (MED module)

Solution of thermohydraulic problem

Solver of Temperature Equation

Creation of hexagonal mesh (MESH module)

Send Neutronic Power Solution (VACS module)

Salomé Platform

NCR: Interpolation Cross Section

MACINI: TRIVAT: Perform a numerical discretization of the reactor geometry

FLUD: Compute the numerical solution to an eigenvalue problem

FLPOW: Compute powers and normalized fluxes

GEO: Assembly geometry

SYBILT: Tracking

USS: Perform self-shielding calculation

ASM: FLU: Solve the linear system of multigroup collision probability

EDI: Condensation to 33 groups

EVO: Perform burnup calculation

COMPO: Creation of neutronic data structure

Libraries Jeff 3.1.2 315 groups

LIB:
## The ALFRED Lead Cooled SMR concept model

### Data and Modeling

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Unit</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thermal Power</td>
<td>MW</td>
<td>300</td>
</tr>
<tr>
<td>Total height vessel</td>
<td>mm</td>
<td>3500</td>
</tr>
<tr>
<td>Inner vessel inner/outer radius</td>
<td>mm</td>
<td>1475/1525</td>
</tr>
<tr>
<td>Lattice pitch (hexagonal)</td>
<td>mm</td>
<td>13.86</td>
</tr>
<tr>
<td>Wrapper thickness</td>
<td>mm</td>
<td>4.0</td>
</tr>
<tr>
<td>Distance between to wrappers</td>
<td>mm</td>
<td>5.0</td>
</tr>
<tr>
<td>Pins per FA</td>
<td>-</td>
<td>126</td>
</tr>
<tr>
<td>Total FA</td>
<td>-</td>
<td>134</td>
</tr>
<tr>
<td>Inner/Outer FA</td>
<td>-</td>
<td>56/78</td>
</tr>
<tr>
<td>Inner/Outer enrichment</td>
<td>%</td>
<td>20.5/26.2</td>
</tr>
<tr>
<td>Control/Safety Rods</td>
<td>-</td>
<td>12/4</td>
</tr>
<tr>
<td>In-pile section</td>
<td>-</td>
<td>1</td>
</tr>
<tr>
<td>Average Core Flow</td>
<td>m/s</td>
<td>1.28</td>
</tr>
<tr>
<td>Coolant Inlet/Outlet Temperature</td>
<td>°C</td>
<td>400/520</td>
</tr>
</tbody>
</table>

### Sketch of the reactor core

![Sketch of the reactor core](image)

### Reactor core fuel map

![Reactor core fuel map](image)
## Fuel pin and assembly

### Data and Modeling

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Unit</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pellet hollow diameter</td>
<td>mm</td>
<td>2.0</td>
</tr>
<tr>
<td>Pellet radius</td>
<td>mm</td>
<td>4.5</td>
</tr>
<tr>
<td>Gap thickness</td>
<td>mm</td>
<td>0.15</td>
</tr>
<tr>
<td>Clad thickness</td>
<td>mm</td>
<td>0.6</td>
</tr>
<tr>
<td>Pin diameter</td>
<td>mm</td>
<td>10.5</td>
</tr>
<tr>
<td>Bottom plug length</td>
<td>mm</td>
<td>50</td>
</tr>
<tr>
<td>Gas plenum height</td>
<td>mm</td>
<td>550</td>
</tr>
<tr>
<td>Bottom plug length</td>
<td>mm</td>
<td>10</td>
</tr>
<tr>
<td>Active height</td>
<td>mm</td>
<td>600</td>
</tr>
<tr>
<td>Upper insulator height</td>
<td>mm</td>
<td>10</td>
</tr>
<tr>
<td>Spring length</td>
<td>mm</td>
<td>120</td>
</tr>
<tr>
<td>Upper plug length</td>
<td>mm</td>
<td>50</td>
</tr>
</tbody>
</table>

**Fuel Assembly**
Finite element geometry for neutronic and CFD codes

CFD: each assembly element is divided in such a way that each sub-element has quadrangular surfaces. This coarse mesh is then refined by using midpoint refinement algorithm several times. Each hexagonal element has constant properties but several field points

CFD Solver Mesh  Neutron transport equation Mesh
Comparison with reference ERANOS (+ECCO cell code) and MCNPX calculations for ALFRED (both performed by ENEA using the JEF3.1 libraries), k-eff

<table>
<thead>
<tr>
<th>Time (days)</th>
<th>DONJON</th>
<th>ERANOS</th>
<th>MCNPX</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1.0924</td>
<td>1.0804</td>
<td>1.0525</td>
</tr>
<tr>
<td>365</td>
<td>1.0645</td>
<td>1.0510</td>
<td>1.0229</td>
</tr>
<tr>
<td>730 (BOC)</td>
<td>1.0338</td>
<td>1.0247</td>
<td>0.9964</td>
</tr>
<tr>
<td>1095 (EOC)</td>
<td>1.0094</td>
<td>0.9988</td>
<td>0.9703</td>
</tr>
</tbody>
</table>
Preliminary results of the multiphysics approach

Flux distribution

Power distribution

Temperature field
Preliminary results of the multiphysics approach

Power generation distribution, vertical section

Wrapping power distribution with temperature field on reactor section
Conclusions

- The solution is obtained after several iterations between the thermo-hydraulic and the neutron code with temperature and density feedback corrections.
- The results obtained in this simple case study, open a perspective of an extensive similar approach to other models: SMRs look as ideal candidates due to their size and compactness (mainly in terms of core dimensions in the FR case).
- The whole procedure is able to give to the user an extreme flexibility, also with respect possible implementation of modules thanks to the complete open-source approach.
- This can be useful in that special case that is represented by Fast SMR conceptual design: our feeling is that we are still (apart from a few cases that are in a licensing or near licensing path) in a phase that requires an open and diffuse contribution in terms of ideas and innovation.