# CANDU-SCWR Coupling Toolset (DONJON5 - CATHENA3)

## Technical presentation

Ulysse Le Tennier

Génie énergétique et nucléaire
Polytechnique Montréal

May, $15^{th}$ 2021

# Summary

# CANDU-SCWR

The canadian proposition for SCWR reactors (Generation IV)

- Coolant : subcritical light water that crosses the supercritical point along its course
- Several design adjustments proposed since 2015
- Coupled calculations between CATHENA3 and DONJON5 for the CANDU-SCWR safety analysis never performed before

# Neutronic thermalhydraulic coupling necessity

The transition from subcritical to supercritical happens in the fuel channels.

The location of the density drop in each channel must be accurately located.

Figure: Axial power for BOC, MOC and EOC when coupling takes place

# Summary

# Summary

# Geometry : hydraulic network

- Only 84 fuel channels declared
- Each channel accounts for 4 (symmetries)
- 20 nodes for flow tube and fuel channel

Figure: CATHENA hydraulic network

# Geometry : thermal models

- Four thermal models
- 0.8 MPa total pressure drop
- An arbitrary share of power between inner (48.1%) and outer rods (51.9%)

Figure: CATHENA thermal models

# Coupling marks

Two constraints on the CATHENA input :

- Coupling marks to be replaced
- Consistent set of outputs

Figure: CATHENA input coupling marks

# Input generation

Seven python functions are provided :

- One aggregating function
- Six group functions

The aggregating function relies on the group functions

The group functions declare the groups required by CATHENA
They cope with the coupling constraints

# Equilibrium calculation

Before coupling, an initial burnup distribution which is independent of the cycle is needed.

Two solutions :

- Provide the good distribution (6720 values)
- Perform enough consecutive cycles to reach equilibrium

Equilibrium functions find the burnup distribution and create DONJON inputs required by the coupling program

# Coupling directories architecture

The coupling program requires a strict directories architecture

- Stores
- Executables

The subdirectories CA, DJ and PY contain templates, data or procedures. They are referred as stores. Detailed information is available in IGE-379

```
EXEC_DIRECTORY
   ├── CA
   ├── DJ
   ├── PY
   ├── DISTRIBUTION
   ├── donjon.exe
   ├── cat3_6_1_1-b01.exe
   └── rdonjon5.bat
```

Figure: Directories architecture

# Coupling general principles

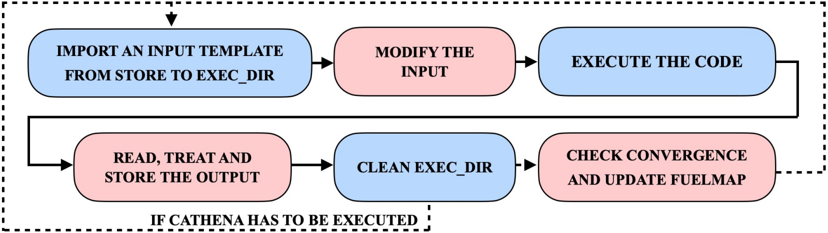


Figure: Coupling program's way to operate

# Coupling data structures

| | Cpl. thermalhydraulic parameters | Non-cpl. thermalhydraulic parameters | Power/Burnup distribution | Critical boron | Execution variables |
|---|---|---|---|---|---|
| numpy array | X (two latest solutions) | X (two latest solutions) | | | |
| FUELMAP | | | X | | |
| HISTORY | | | | X | X |
| .txt file (DISTRIBUTION) | X (latest cycle's solutions) | X (latest cycle's solutions) | | | |
| Python output string | X | X | | | |

Figure: Data dispatching among data structures

Cpl thermalhydraulic parameters :
coolant densities, temperature, fuel temperature and mass flow

# HISTORY structure

DONJON uses specific variables to manage reloading, to burn fuel... Those are stored in HISTORY

- Python main program reads Step variable
- Passes it to DONJON input
- DONJON reads variables in HISTORY at Step

  - Few variables to pass from python to DONJON
  - One fundamental variable, Step
  - All the variables are saved in a file, not in python variables

# Restart coupling

To restart the coupling after the program stopped :

- Correct potential errors
- Clean the execution directory from inputs/outputs
- Retrieve the latest value of Step in HISTORY (DJ dir.)
- Call the main program at the latest Step value

When stopped, python output strings are lost

The latest cycle's solution is available in DISTRIBUTION dir.

# Data treatment principles

Data treatment :

- Recovers 3D data
- Unfolds data
- Builds a 3D matrix consistent with the physical geometry
- Plot heatmaps (if necessary)

It is assumed that, from this basis, additional data treatment programs are easy to create.

# Data treatment functions

Two files are provided TreatMaps.py and TreatMapMFlw.py :

TreatMaps.py contains one function to read power distribution (FMAP), one to read python output strings other than mass flow and one to plot heatmaps

TreatMapMFlw.py contains a function to read mass flow distributions and one to plot heatmaps

# Summary

# IGE-379 overview

IGE-379 is available on GitHub (click), it introduces :

- Core geometries
- General principles to use the coupling program
- Data treatment principles
- CATHENA input generation functions

# IGE-379 most valuable informations

This manual should be used :

- To get familiar with the core geometries
- To get familiar with the main program algorithm
- To have insights on convergence feature
- To modify CATHENA input

CATHENA and DONJON user's manual are cited. They are a valuable support to modify the input of each code.

# CSCT-D5C3 User's Manual overview

CSCT-D5C3 User's Manual is available at GitHub (click), for each function, it provides :

- A general and technical description
- Input and output
- Websites for eventual support (seaborn heatmaps)

The way the functions are chained is not explicitly described

# CSCT-D5C3 User's Manual most valuable informations

This manual should be used :

- To understand technical aspects of the main program
- To modify accurately the coupling program
- To modify CATHENA input generation functions

# Summary

# Databases

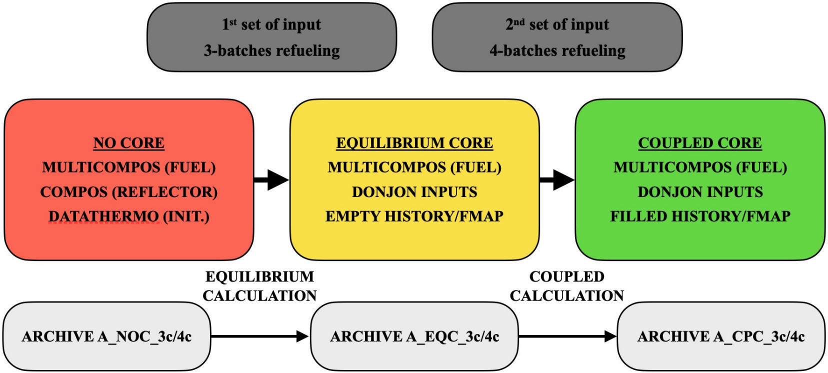


Figure: Different archives available, if asked, at letennier.u@gmail.com