Multiphysics Modelling to Optimize Design and Safety of Advanced Nuclear Reactors

Webinar Series on Nuclear Technology Breakthroughs for the 21st Century

17 March 2021 14:30-16:00 CET (GMT +01:00)

Hello everyone and welcome to today's IAEA Webinar.
We start at 14:30 CET (GMT +01:00)

Multiphysics Modelling to Optimize Design and Safety of Advanced Nuclear Reactors

Webinar Series on Nuclear Technology Breakthroughs for the 21st Century

17 March 2021 14:30-16:00 CET (GMT +01:00)

PLEASE NOTE..

This is an **interactive Webinar** and we would love to hear from you, so if you have any question, please type them at any time into the **CHAT panel** and we will try to address as many of your questions during our Q&A session.

Please also use the **CHAT panel** to let us know if you're experiencing any technical problems and our technical team will help you out.

Participants will be muted for the duration of the session.

Please note that the IAEA is **not offering certifications** for the Webinars in this series.

Multiphysics Modelling to Optimize Design and Safety of Advanced Nuclear Reactors

Webinar Series on Nuclear Technology Breakthroughs for the 21st Century

17 March 2021 14:30-16:00 CET (GMT +01:00)

Moderators:

Chirayu Batra
Nuclear Power Project Officer,
SMR Team, International
Atomic Energy Agency

Michal Zeman
Intern, Fast Reactors
Team, International
Atomic Energy Agency

Multiphysics Modelling to Optimize Design and Safety of Advanced Nuclear Reactors

Webinar Series on Nuclear Technology Breakthroughs for the 21st Century

17 March 2021 14:30-16:00 CET (GMT +01:00)

Speakers:

Christophe Demaziere
Professor, Chalmers
University of Technology,
Sweden

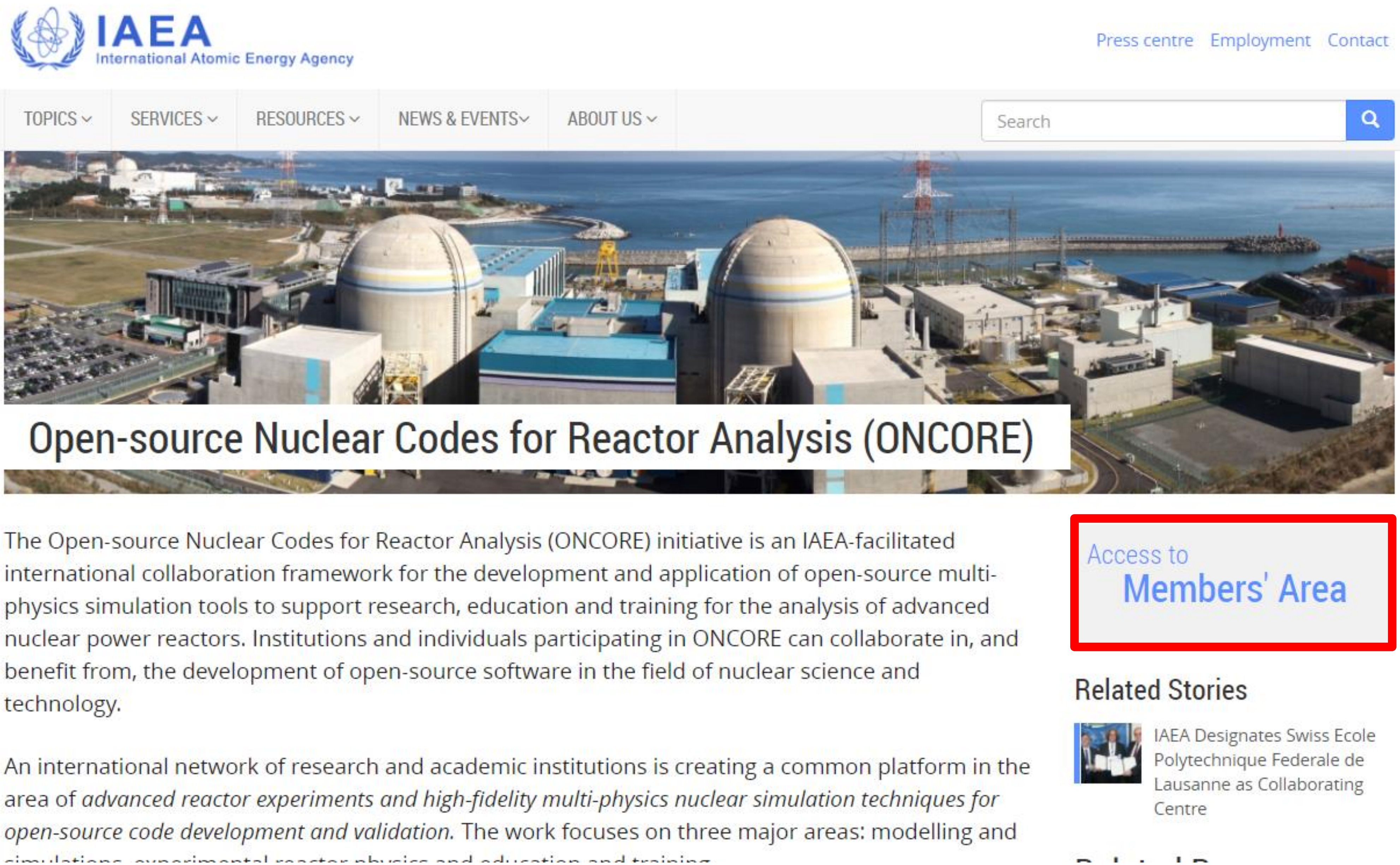
Carlo Fiorina
Scientist, École polytechnique
fédérale de Lausanne (EPFL),
Switzerland

Kathryn D. Huff
Assistant Professor,
University of Illinois at
Urbana-Champaign, USA

ONCORE

Open-source Nuclear Codes for Reactor Analysis

The ONCORE initiative is an IAEA-facilitated international collaboration framework for the development and application of open-source multi-physics simulation tools to support research, education and training for the analysis of advanced nuclear power reactors. Institutions and individuals participating in ONCORE can collaborate in, and benefit from, the development of open-source software in the field of nuclear science and technology.



Open-source Nuclear Codes for Reactor Analysis (ONCORE)

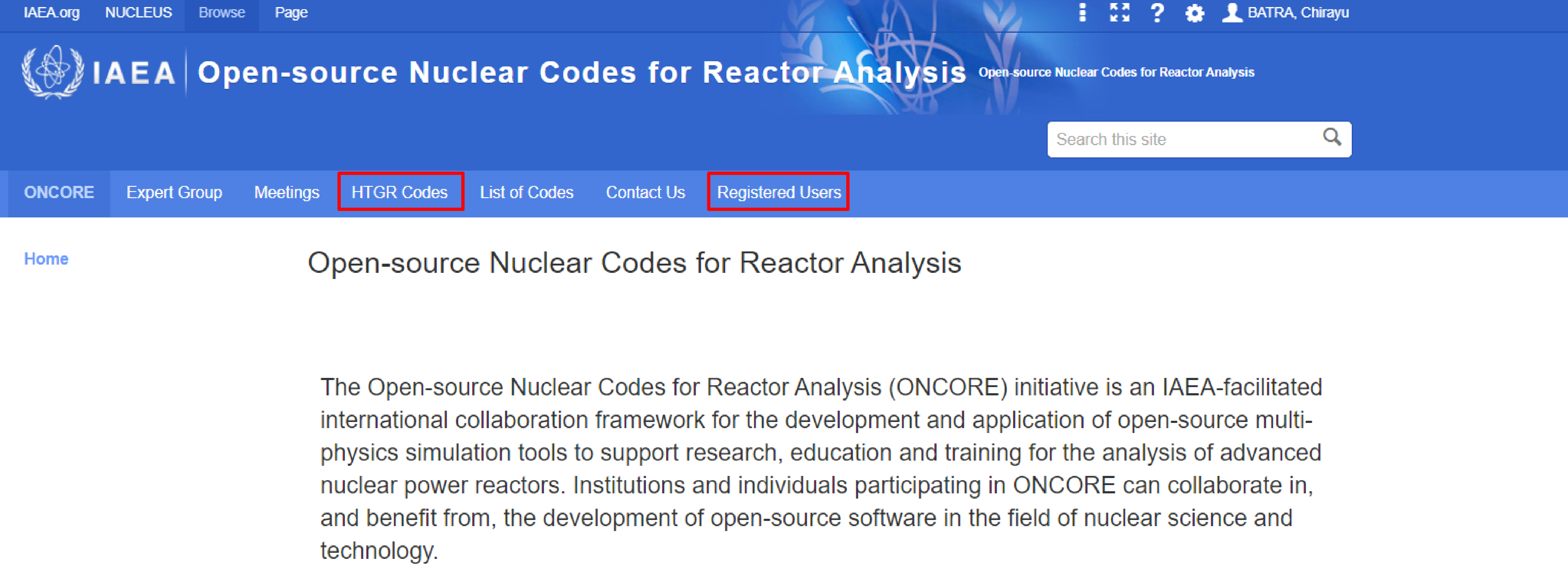
The Open-source Nuclear Codes for Reactor Analysis (ONCORE) initiative is an IAEA-facilitated international collaboration framework for the development and application of open-source multi-physics simulation tools to support research, education and training for the analysis of advanced nuclear power reactors. Institutions and individuals participating in ONCORE can collaborate in, and benefit from, the development of open-source software in the field of nuclear science and technology.

An international network of research and academic institutions is creating a common platform in the area of *advanced reactor experiments and high-fidelity multi-physics nuclear simulation techniques for open-source code development and validation.* The work focuses on three major areas: modelling and

Related Stories

IAEA Designates Swiss Ecole Polytechnique Federale de Lausanne as Collaborating Centre

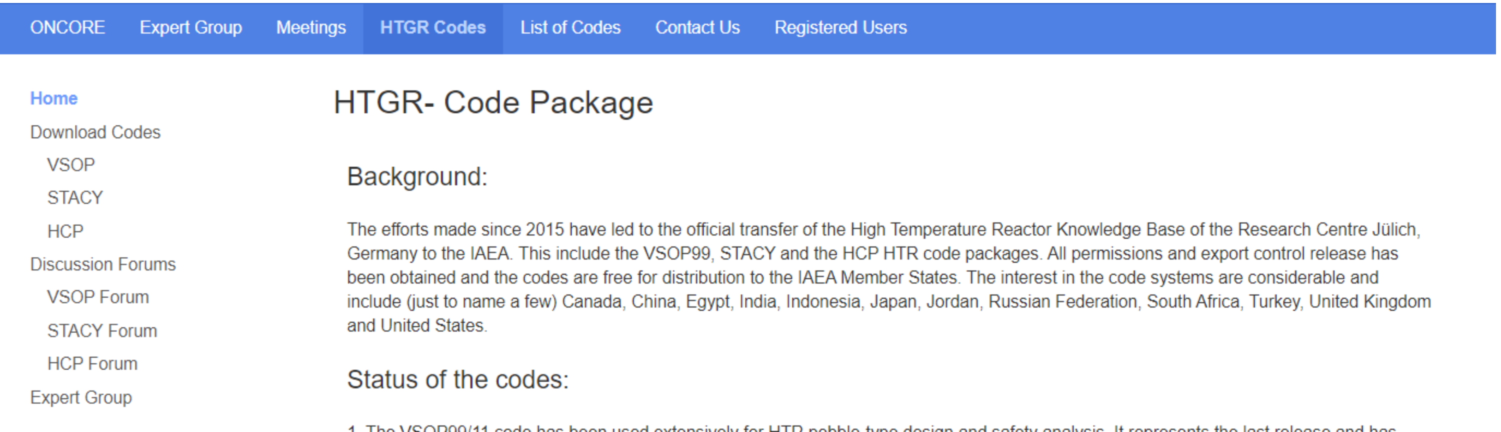
https://www.iaea.org/topics/nuclear-power-reactors/open-source-nuclear-code-for-reactor-analysis-oncore



Open-source Nuclear Codes for Reactor Analysis

The Open-source Nuclear Codes for Reactor Analysis (ONCORE) initiative is an IAEA-facilitated international collaboration framework for the development and application of open-source multi-physics simulation tools to support research, education and training for the analysis of advanced nuclear power reactors. Institutions and individuals participating in ONCORE can collaborate in, and benefit from, the development of open-source software in the field of nuclear science and technology.

https://nucleus.iaea.org/sites/oncore



HTGR- Code Package

Background:

The efforts made since 2015 have led to the official transfer of the High Temperature Reactor Knowledge Base of the Research Centre Jülich, Germany to the IAEA. This include the VSOP99, STACY and the HCP HTR code packages. All permissions and export control release has been obtained and the codes are free for distribution to the IAEA Member States. The interest in the code systems are considerable and include (just to name a few) Canada, China, Egypt, India, Indonesia, Japan, Jordan, Russian Federation, South Africa, Turkey, United Kingdom and United States.

Status of the codes:

1. The VSOP99/11 code has been used extensively for HTR pebble type design and safety analysis. It represents the last release and has

- Collaborative development
- Code submission (guidelines)
- Expert group to support
- Training courses
- Useful resources

IAEA Technical Meeting
on
Development and Application of Open-source Modelling and Simulation Tools for Nuclear Reactors

27-29 October 2021

VIC (M6), Vienna, Austria

https://www.iaea.org/topics/nuclear-power-reactors/open-source-nuclear-code-for-reactor-analysis-oncore

https://nucleus.iaea.org/sites/oncore

ONCORE@iaea.org

Prof. Christophe Demazière

Christophe Demazière is Full Professor at Chalmers University of Technology in Gothenburg, Sweden, leading the DREAM research group (Deterministic Reactor Modelling). DREAM is a cross-disciplinary group having expertise in neutron transport, fluid dynamics, heat transfer, and numerical methods. The aim of the group is to develop beyond state-of-the-art techniques for modelling nuclear reactors, thus contributing to improved simulations tools and enhanced safety.

Multi-physics modelling: from segregated to integrated coupling strategies

Prof. Christophe Demazière
demaz@chalmers.se

in

@ChrisDemaz

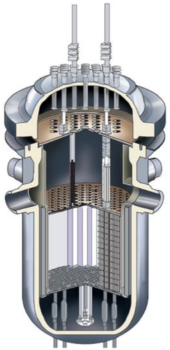
DREAM

TASK FORCE ON
DETERMINISTIC REACTOR MODELLING

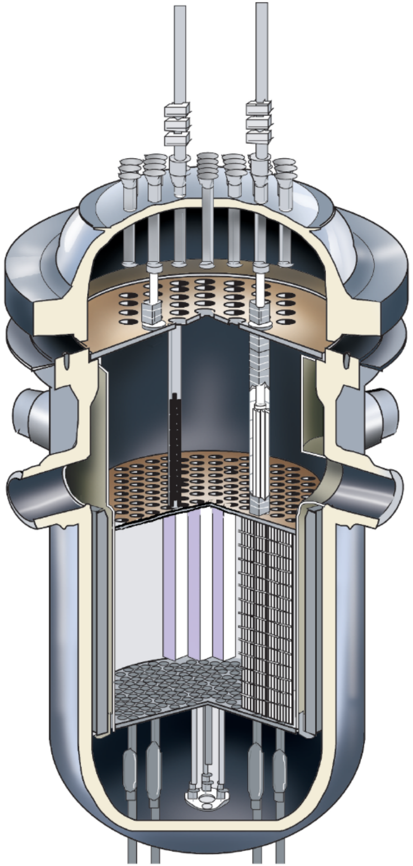
Introduction

- Nuclear reactors = **large** and **complex** systems

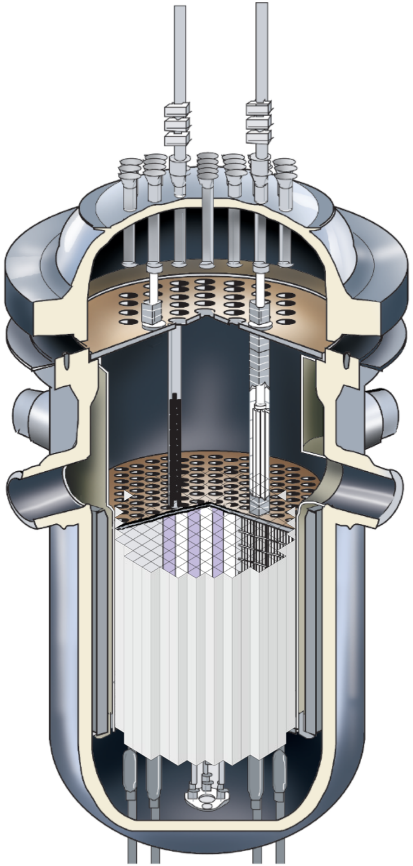
Introduction



Introduction

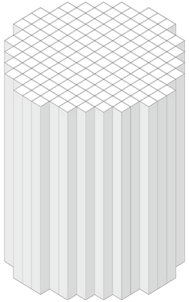


Introduction



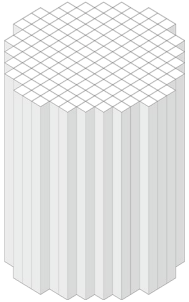
Picture courtesy of Westinghouse Electric Sweden AB

Introduction

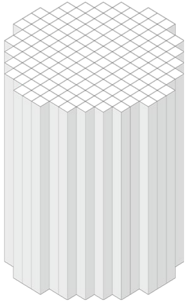


reactor core

Introduction

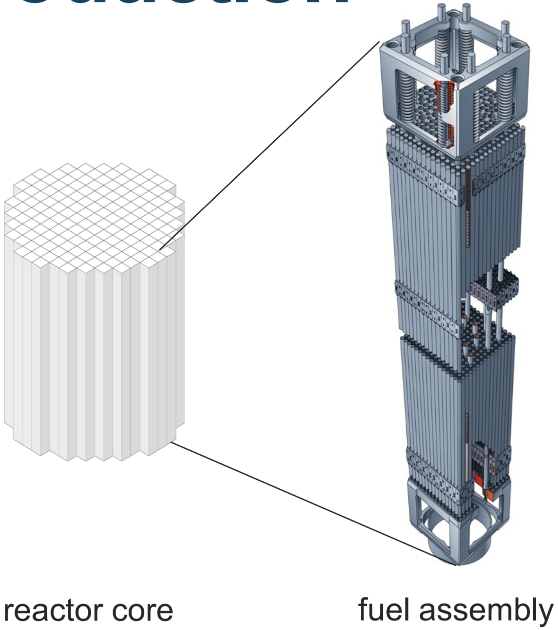


Introduction



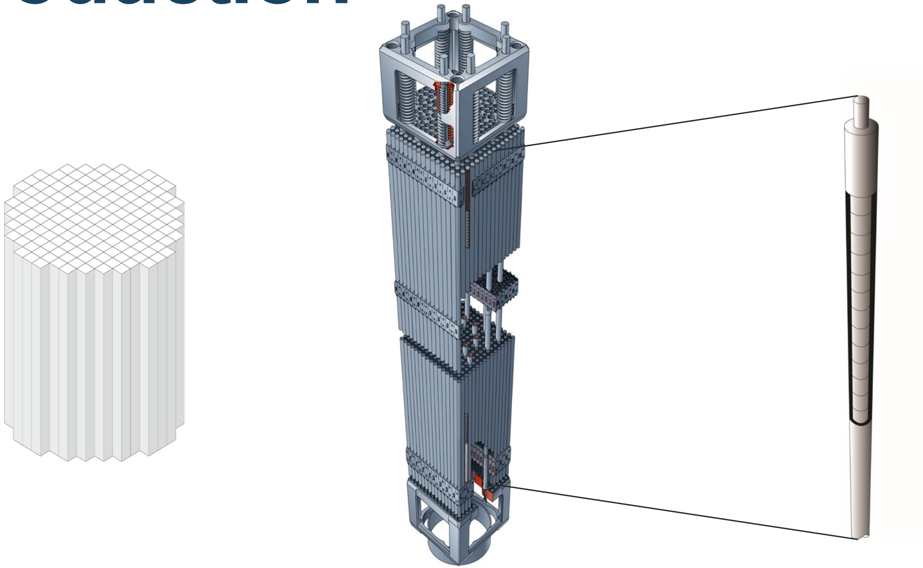
reactor core

Introduction



Picture courtesy of
Westinghouse Electric Sweden AB

Introduction



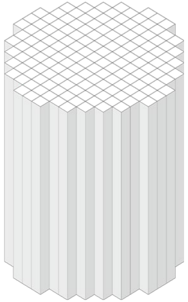
reactor core

fuel assembly

fuel pin

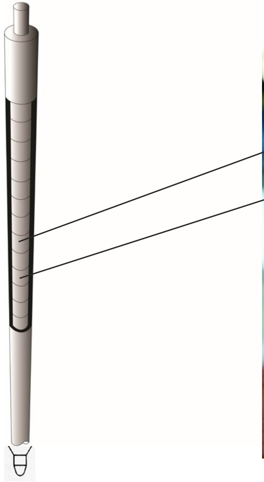
Picture courtesy of the Swedish Academic Initiative for Radiation Sciences and Nuclear Technology

Introduction

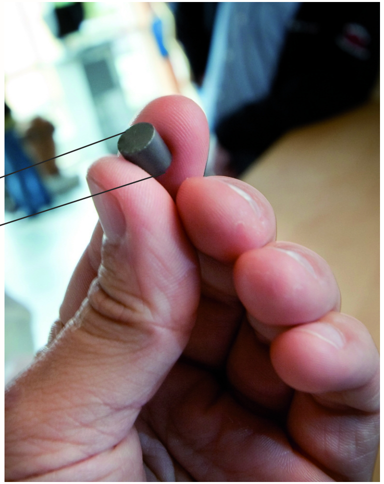


reactor core

fuel assembly



fuel pin



fuel pellet

Picture courtesy of
Analysgruppen - Energiföretagen

Introduction

- **Interplay** between **different fields of physics**:
  - Neutron transport
  - Fluid dynamics
  - Heat transfer
  - Fuel behaviour
  - Structural mechanics
  - Coolant and radiation chemistry
  - Radionuclide transport
  - Etc.

Introduction

- Modelling of such systems has typically been made focusing on **one physics at a time** (with frozen boundary conditions from the other physics)
- “**Less conservative**” estimates rely on **more faithful modelling**

➢Necessary for **optimized reactor design** and **safety**

Multiphysics coupling strategies

- Multi-physics problem generically written as (before time discretization):

$$\frac{d\mathbf{u}}{dt}\left(t\right)=\mathbf{F}\left(\mathbf{u},t\right)$$

- In case of two physics φ_1 and φ_2, problem solved as:

$$\frac{d}{dt}\begin{bmatrix}\mathbf{u}_{\varphi_1}\left(t\right)\\ \mathbf{u}_{\varphi_2}\left(t\right)\end{bmatrix}=\mathbf{F}\left(\begin{bmatrix}\mathbf{u}_{\varphi_1}\left(t\right)\\ \mathbf{u}_{\varphi_2}\left(t\right)\end{bmatrix},t\right)$$

or

$$\frac{d}{dt}\begin{bmatrix}\mathbf{u}_{\varphi_1}\left(t\right)\\ \mathbf{u}_{\varphi_2}\left(t\right)\end{bmatrix}=\begin{bmatrix}\mathbf{F}_{\varphi_1}\left(\begin{bmatrix}\mathbf{u}_{\varphi_1}\left(t\right)\\ \mathbf{u}_{\varphi_2}\left(t\right)\end{bmatrix},t\right)\\ \mathbf{F}_{\varphi_2}\left(\begin{bmatrix}\mathbf{u}_{\varphi_1}\left(t\right)\\ \mathbf{u}_{\varphi_2}\left(t\right)\end{bmatrix},t\right)\end{bmatrix}$$

monolithic approach

segregated approach

Multiphysics coupling strategies

- **Segregated approaches mostly used** because of the extensive verification and validation of mono-physics solvers
- **Different ways** to implement **segregated approaches**:
  - **Exchange of data** via **input/output files** (+ **scripts**)
  - **Exchange of data** within the **computer memory**
    - Mono-physics solvers **compiled into one executable**
    - Use of a **message passing interface**
- Remark: using one single software can still rely on segregated approaches

Multiphysics coupling strategies

- Multi-physics problem rewritten as:

$$\frac{d}{dt}\begin{bmatrix}\mathbf{u}_{\varphi_1}\left(t\right)\\ \mathbf{u}_{\varphi_2}\left(t\right)\end{bmatrix}=\begin{bmatrix}\mathbf{L}_{\varphi_1}\left(t\right) & 0\\ 0 & \mathbf{L}_{\varphi_2}\left(t\right)\end{bmatrix}\times\begin{bmatrix}\mathbf{u}_{\varphi_1}\left(t\right)\\ \mathbf{u}_{\varphi_2}\left(t\right)\end{bmatrix}+\cdots$$

where dependence on the other physics assumed to be in the non-linearities

Multiphysics coupling strategies

- Multi-physics problem rewritten as:

$$\frac{d}{dt}\begin{bmatrix}\mathbf{u}_{\varphi_1}(t)\\ \mathbf{u}_{\varphi_2}(t)\end{bmatrix} = \begin{bmatrix}\mathbf{L}_{\varphi_1}(t) & 0\\ 0 & \mathbf{L}_{\varphi_2}(t)\end{bmatrix}\times\begin{bmatrix}\mathbf{u}_{\varphi_1}(t)\\ \mathbf{u}_{\varphi_2}(t)\end{bmatrix} + \begin{bmatrix}\mathbf{N}_{\varphi_1}\left(\begin{bmatrix}\mathbf{u}_{\varphi_1}(t)\\ \color{red}{\mathbf{u}_{\varphi_2}(t)}\end{bmatrix}, t\right)\\ \\ \end{bmatrix}$$

where dependence on the other physics assumed to be in the non-linearities

Multiphysics coupling strategies

- Multi-physics problem rewritten as:

$$\frac{d}{dt}\begin{bmatrix}\mathbf{u}_{\varphi_1}(t)\\ \mathbf{u}_{\varphi_2}(t)\end{bmatrix}=\begin{bmatrix}\mathbf{L}_{\varphi_1}(t) & 0\\ 0 & \mathbf{L}_{\varphi_2}(t)\end{bmatrix}\times\begin{bmatrix}\mathbf{u}_{\varphi_1}(t)\\ \mathbf{u}_{\varphi_2}(t)\end{bmatrix}+\begin{bmatrix}\mathbf{N}_{\varphi_1}\left(\begin{bmatrix}\mathbf{u}_{\varphi_1}(t)\\ \color{red}{\mathbf{u}_{\varphi_2}(t)}\end{bmatrix},t\right)\\ \mathbf{N}_{\varphi_2}\left(\begin{bmatrix}\color{red}{\mathbf{u}_{\varphi_1}(t)}\\ \mathbf{u}_{\varphi_2}(t)\end{bmatrix},t\right)\end{bmatrix}$$

where dependence on the other physics assumed to be in the non-linearities

Multiphysics coupling strategies

- **Segregated** or **operator splitting** strategies:
  use of each of the mono-physics solvers in their non-altered forms and on some exchange of information/data between the solvers

➢Three basic approaches:

- Non-linearities from the other mono-physics solver evaluated at the previous time step

$$\mathbf{N}_{\varphi_k}\left(\begin{bmatrix}\mathbf{u}_{\varphi_k}\left(t+\Delta t\right)\\ \mathbf{u}_{\varphi_{l\neq k}}\left(t+\Delta t\right)\end{bmatrix}, t+\Delta t\right) \quad \text{replaced by} \quad \mathbf{N}_{\varphi_k}\left(\begin{bmatrix}\mathbf{u}_{\varphi_k}\left(t+\Delta t\right)\\ \mathbf{u}_{\varphi_{l\neq k}}\left(t\right)\end{bmatrix}, t+\Delta t\right)$$

➢**Non-linear inconsistencies** introduced

Multiphysics coupling strategies

- **Segregated** or **operator splitting** strategies:
  use of each of the mono-physics solvers in their non-altered forms and on some exchange of information/data between the solvers

➢Three basic approaches:

- φ_1 first solved using the non-linearities from the other mono-physics solver φ_2 evaluated at the previous time step

$$\mathbf{N}_{\varphi_1}\left(\begin{bmatrix}\mathbf{u}_{\varphi_1}\left(t+\Delta t\right)\\ \mathbf{u}_{\varphi_2}\left(t+\Delta t\right)\end{bmatrix}, t+\Delta t\right) \text{ replaced by } \mathbf{N}_{\varphi_1}\left(\begin{bmatrix}\mathbf{u}_{\varphi_1}\left(t+\Delta t\right)\\ \mathbf{u}_{\varphi_2}\left(t\right)\end{bmatrix}, t+\Delta t\right)$$

  φ_2 then solved using the solution $\mathbf{u}^*_{\varphi_1}\left(t+\Delta t\right)$ evaluated above at the current time step

$$\mathbf{N}_{\varphi_2}\left(\begin{bmatrix}\mathbf{u}_{\varphi_1}\left(t+\Delta t\right)\\ \mathbf{u}_{\varphi_2}\left(t+\Delta t\right)\end{bmatrix}, t+\Delta t\right) \text{ replaced by } \mathbf{N}_{\varphi_2}\left(\begin{bmatrix}\mathbf{u}^*_{\varphi_1}\left(t+\Delta t\right)\\ \mathbf{u}_{\varphi_2}\left(t+\Delta t\right)\end{bmatrix}, t+\Delta t\right)$$

  ➢**Non-linear inconsistencies** introduced

Multiphysics coupling strategies

- **Segregated** or **operator splitting** strategies:
  use of each of the mono-physics solvers in their non-altered forms and on some exchange of information/data between the solvers

➢Three basic approaches:

  - Successive updates of the solution vector as:

$$\begin{bmatrix} \mathbf{u}^{1}_{\varphi_1}\left(t+\Delta t\right) \\ \mathbf{u}_{\varphi_2}\left(t\right) \end{bmatrix}$$

Multiphysics coupling strategies

- Segregated or operator splitting strategies:
  use of each of the mono-physics solvers in their non-altered forms and on some exchange of information/data between the solvers

➢Three basic approaches:

  - Successive updates of the solution vector as:

$$\begin{bmatrix} \mathbf{u}_{\varphi_1}^{1}\left(t+\Delta t\right) \\ \mathbf{u}_{\varphi_2}\left(t\right) \end{bmatrix} \rightarrow \begin{bmatrix} \mathbf{u}_{\varphi_1}^{1}\left(t+\Delta t\right) \\ \mathbf{u}_{\varphi_2}^{1}\left(t+\Delta t\right) \end{bmatrix}$$

Multiphysics coupling strategies

- Segregated or operator splitting strategies:
  use of each of the mono-physics solvers in their non-altered forms and on some exchange of information/data between the solvers

➢Three basic approaches:

  - Successive updates of the solution vector as:

$$\begin{bmatrix} {\color{red}\mathbf{u}_{\varphi_1}^{1}\left(t+\Delta t\right)} \\ \mathbf{u}_{\varphi_2}\left(t\right) \end{bmatrix} \rightarrow \begin{bmatrix} \mathbf{u}_{\varphi_1}^{1}\left(t+\Delta t\right) \\ {\color{red}\mathbf{u}_{\varphi_2}^{1}\left(t+\Delta t\right)} \end{bmatrix} \rightarrow \begin{bmatrix} {\color{red}\mathbf{u}_{\varphi_1}^{2}\left(t+\Delta t\right)} \\ \mathbf{u}_{\varphi_2}^{1}\left(t+\Delta t\right) \end{bmatrix}$$

Multiphysics coupling strategies

- Segregated or operator splitting strategies:
  use of each of the mono-physics solvers in their non-altered forms and on some exchange of information/data between the solvers

➢ Three basic approaches:

  - Successive updates of the solution vector as:

$$\begin{bmatrix} \mathbf{u}^{1}_{\varphi_1}\left(t+\Delta t\right) \\ \mathbf{u}_{\varphi_2}\left(t\right) \end{bmatrix} \rightarrow \begin{bmatrix} \mathbf{u}^{1}_{\varphi_1}\left(t+\Delta t\right) \\ \mathbf{u}^{1}_{\varphi_2}\left(t+\Delta t\right) \end{bmatrix} \rightarrow \begin{bmatrix} \mathbf{u}^{2}_{\varphi_1}\left(t+\Delta t\right) \\ \mathbf{u}^{1}_{\varphi_2}\left(t+\Delta t\right) \end{bmatrix} \rightarrow \begin{bmatrix} \mathbf{u}^{2}_{\varphi_1}\left(t+\Delta t\right) \\ \mathbf{u}^{2}_{\varphi_2}\left(t+\Delta t\right) \end{bmatrix}$$

Multiphysics coupling strategies

- Segregated or operator splitting strategies:
  use of each of the mono-physics solvers in their non-altered forms and on some exchange of information/data between the solvers

➢Three basic approaches:

  - Successive updates of the solution vector as:

$$\begin{bmatrix} \mathbf{u}^{1}_{\varphi_1}\left(t+\Delta t\right) \\ \mathbf{u}_{\varphi_2}\left(t\right) \end{bmatrix} \rightarrow \begin{bmatrix} \mathbf{u}^{1}_{\varphi_1}\left(t+\Delta t\right) \\ \mathbf{u}^{1}_{\varphi_2}\left(t+\Delta t\right) \end{bmatrix} \rightarrow \begin{bmatrix} \mathbf{u}^{2}_{\varphi_1}\left(t+\Delta t\right) \\ \mathbf{u}^{1}_{\varphi_2}\left(t+\Delta t\right) \end{bmatrix} \rightarrow \begin{bmatrix} \mathbf{u}^{2}_{\varphi_1}\left(t+\Delta t\right) \\ \mathbf{u}^{2}_{\varphi_2}\left(t+\Delta t\right) \end{bmatrix} \rightarrow \ldots \rightarrow \begin{bmatrix} \mathbf{u}^{n}_{\varphi_1}\left(t+\Delta t\right) \\ \mathbf{u}^{n-1}_{\varphi_2}\left(t+\Delta t\right) \end{bmatrix}$$

Multiphysics coupling strategies

- Segregated or operator splitting strategies:
  use of each of the mono-physics solvers in their non-altered forms and on some exchange of information/data between the solvers

➢Three basic approaches:

  - **Successive updates** of the solution vector as:

$$\begin{bmatrix} \mathbf{u}^{1}_{\varphi_1}(t+\Delta t) \\ \mathbf{u}_{\varphi_2}(t) \end{bmatrix} \rightarrow \begin{bmatrix} \mathbf{u}^{1}_{\varphi_1}(t+\Delta t) \\ \mathbf{u}^{1}_{\varphi_2}(t+\Delta t) \end{bmatrix} \rightarrow \begin{bmatrix} \mathbf{u}^{2}_{\varphi_1}(t+\Delta t) \\ \mathbf{u}^{1}_{\varphi_2}(t+\Delta t) \end{bmatrix} \rightarrow \begin{bmatrix} \mathbf{u}^{2}_{\varphi_1}(t+\Delta t) \\ \mathbf{u}^{2}_{\varphi_2}(t+\Delta t) \end{bmatrix} \rightarrow \ldots \rightarrow \begin{bmatrix} \mathbf{u}^{n}_{\varphi_1}(t+\Delta t) \\ \mathbf{u}^{n-1}_{\varphi_2}(t+\Delta t) \end{bmatrix} \rightarrow \begin{bmatrix} \mathbf{u}^{n}_{\varphi_1}(t+\Delta t) \\ \mathbf{u}^{n}_{\varphi_2}(t+\Delta t) \end{bmatrix}$$

  ➢**Non-linear inconsistencies resolved**

  ➢**Convergence** usually slow/difficult

Multiphysics coupling strategies

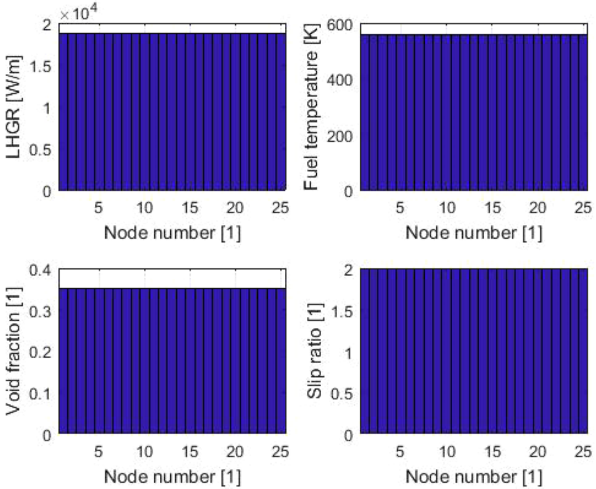
- **Monolithic** approaches:
  Entire multi-physics problem rewritten as “**one**” problem:

$$\mathbf{H}\left(\mathbf{u}\left(t+\Delta t\right)\right)=\mathbf{0}$$

➢Due to the different time scales and characteristic lengths of each physics, the problem is often **ill-conditioned**: need to **pre-condition** the problem

Multiphysics coupling strategies

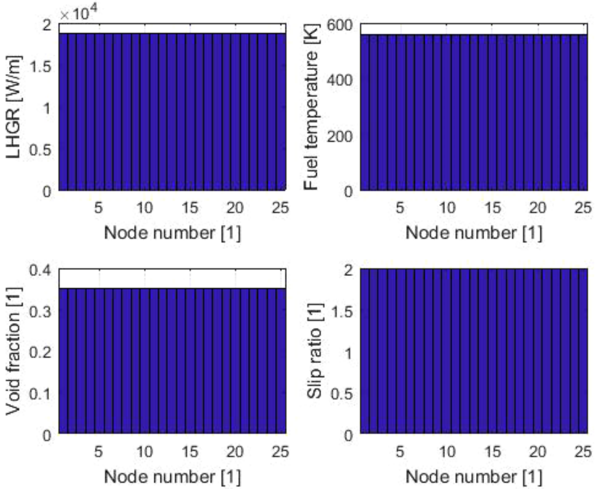
- Example of a 1-dimensional heterogeneous model of a Boiling Water Reactor in steady-state conditions:



Segregated approach

Multiphysics coupling strategies

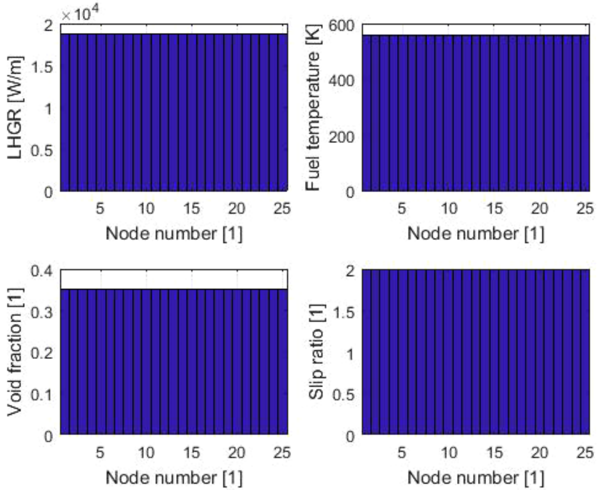
- Example of a 1-dimensional heterogeneous model of a Boiling Water Reactor in steady-state conditions:



Segregated approach

Multiphysics coupling strategies

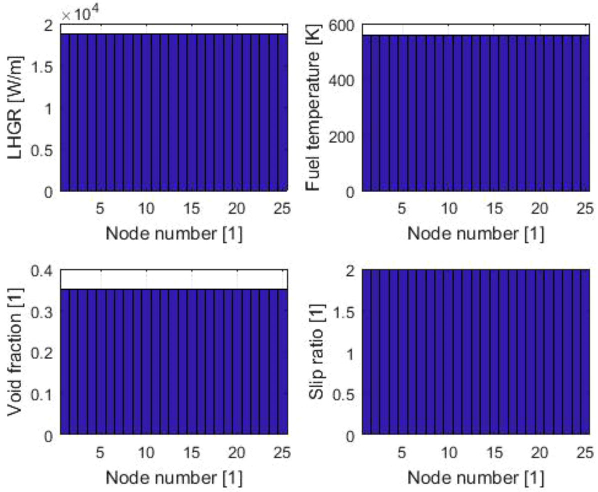
- Example of a 1-dimensional heterogeneous model of a Boiling Water Reactor in steady-state conditions:



Segregated approach with damping

Multiphysics coupling strategies

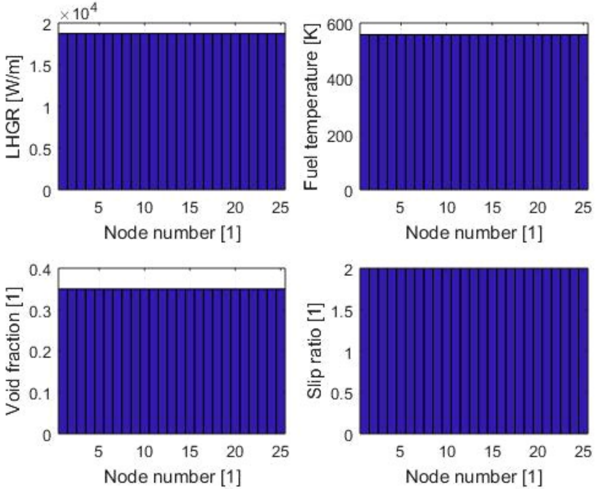
- Example of a 1-dimensional heterogeneous model of a Boiling Water Reactor in steady-state conditions:



Segregated approach with damping

Multiphysics coupling strategies

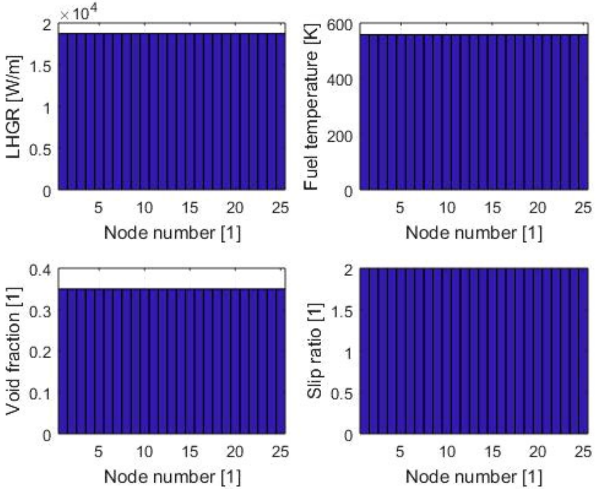
- Example of a 1-dimensional heterogeneous model of a Boiling Water Reactor in steady-state conditions:



Monolithic approach (Jacobian Free Newton Krylov method)

Multiphysics coupling strategies

- Example of a 1-dimensional heterogeneous model of a Boiling Water Reactor in steady-state conditions:



Monolithic approach (Jacobian Free Newton Krylov method)

Conclusions and outlook

- **Segregated** approaches:
  - **Pros**: Extensive V&V + codes tuned to a specific purpose
  - **Cons**: Reaching convergence might be challenging + codes tuned to a specific purpose
- **Monolithic** approaches:
  - **Pros**: Better control of convergence
  - **Cons**: Robustness of the methods might be challenging

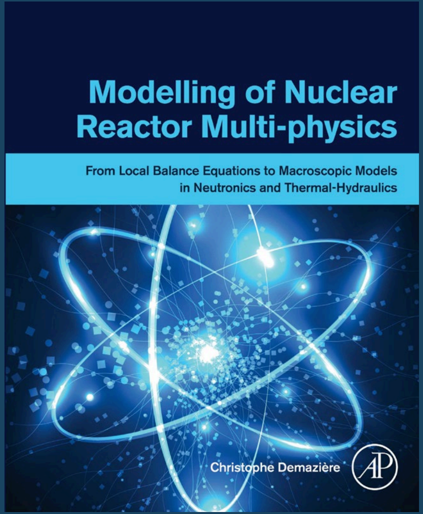
➢Irrespective of the methods used, **multi-physics coupling** best achieved if access to **source code**

CHALMERS
UNIVERSITY OF TECHNOLOGY

Multiphysics modelling: from segregated to integrated coupling strategies

Prof. Christophe Demazière
demaz@chalmers.se

Learn more about reactor modelling:
(Elsevier/Academic Press book)


Modelling of Nuclear Reactor Multi-physics

From Local Balance Equations to Macroscopic Models in Neutronics and Thermal-Hydraulics

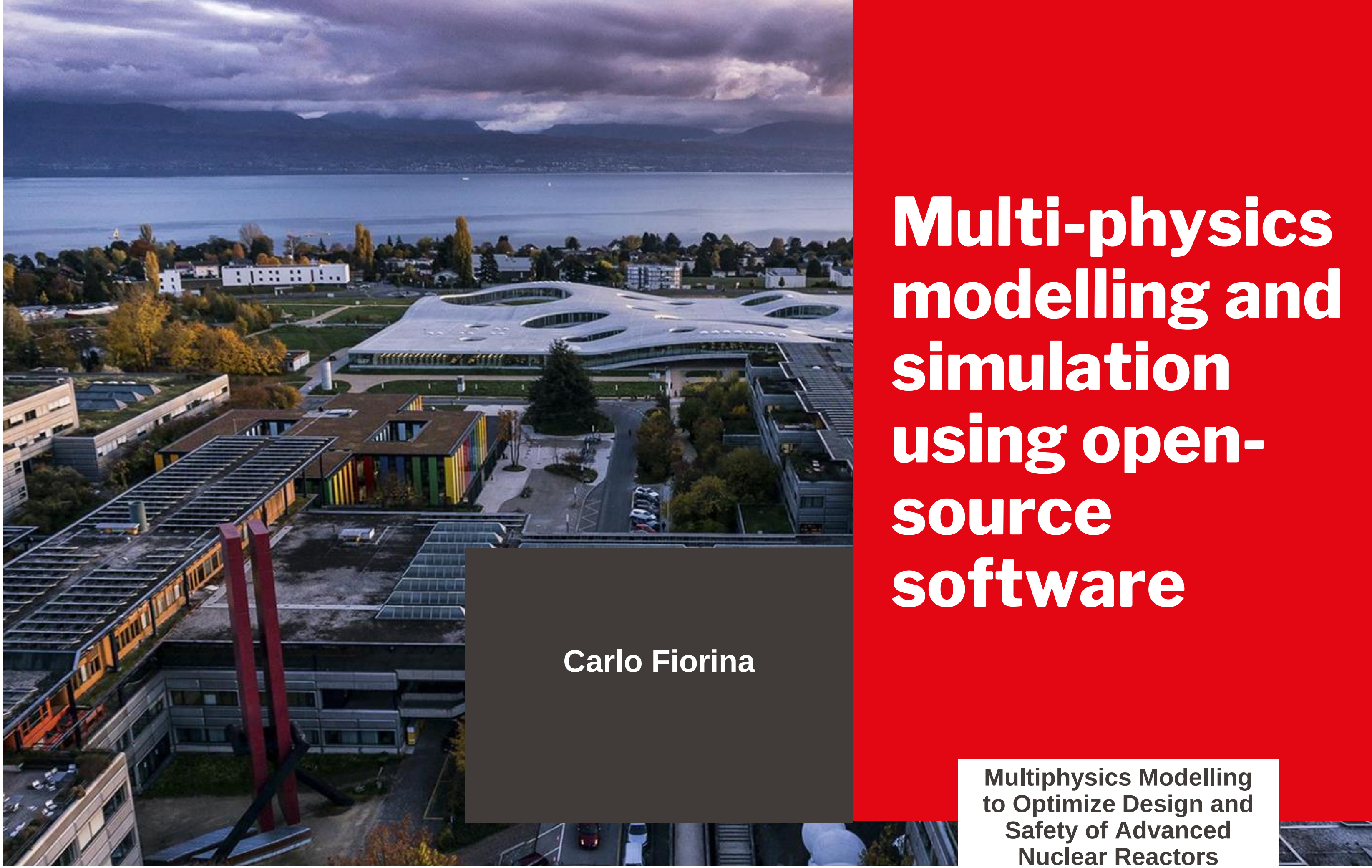
Christophe Demazière

DREAM

TASK FORCE ON
DETERMINISTIC REACTOR MODELLING

Dr. Carlo Fiorina

Carlo Fiorina is a Scientist and Program Manager for Computational Science at the Laboratory for Reactor Physics and System Behaviour at the EPFL, Switzerland. His research activities are focused on the development and application of advanced modelling algorithms and tools for the high-fidelity simulation of advanced nuclear reactors. He leads since 2014 the development of the GeN-Foam open-source solver for the multi-physics analyses of nuclear reactors, as well as the EPFL activities on the development of the OFFBEAT open-source solver for multi-dimensional fuel performance analysis. He is also chair of the OpenFOAM Nuclear Technical Committee and a main contributor to the ONCORE initiative.



Multi-physics modelling and simulation using open-source software

Carlo Fiorina

Multiphysics Modelling to Optimize Design and Safety of Advanced Nuclear Reactors
17.03.2020

École
polytechnique
fédérale
de Lausanne

Motivations for open-source software

- Cost free
- Allows experimenting
- Preserve and valorize R&D work
- Multiply R&D throughput
    - stimulate synergies
    - avoid work duplicates
    - involve a broad community
- Useful for
    - Research institutes: cost free, preserve R&D work, allows experimenting
    - Regulators: “independent” and transparent tool
    - E&T institutions: license, avoid black-box approaches, improve understanding
    - Industry: incorporate acquired know-how
- Often encouraged in publicly-funded projects (e.g., Euratom)

Drawbacks/challenges of open-source software

- Documentation
- Quality assurance (multiple contributors, continuity of work)

In nuclear:

- Lack of open-access data
- Fragmented community

The ONCORE initiative

Several challenges can be addressed by an initiative like ONCORE

- Promote collaboration and facilitate communication (connect the community)
- Provide guidelines for code contribution (documentation, QA)
- Provide development best practices (QA)
- Preserve knowledge
    - Incl. compiling a list of open-source codes

A first important outcome: list of available codes

- https://nucleus.iaea.org/sites/oncore/SitePages/List%20of%20Codes.aspx
- A vibrant community with an impressive R&D output
- ~35 codes already identified so far:
    - OpenMC
    - Raven
    - Dragon
    - MOOSE
    - Salome platform (Code_Saturne, Code_Aster)
    - TrioCFD
    - …
    - **Several OpenFOAM-based tools**

An example: OpenFOAM

The Open Source CFD Toolbox

- What is OpenFOAM?
  - Distributed as CFD toolbox
  - ~10k to 20k estimated users worldwide
  - OpenFOAM = Open Field Operation And Manipulation
  - Essentially a large, well organized, HPC-scalable, C++ library for the finite-volume discretization and solution of PDEs, and including several functionalities like ODE solvers, projection algorithms, and mesh search algorithms
  - Object-oriented, with a high-level “fail-safe” API

$$\frac{1}{v_i}\frac{\partial \varphi_i}{\partial t} - \Delta(D_i \varphi_i) = S$$

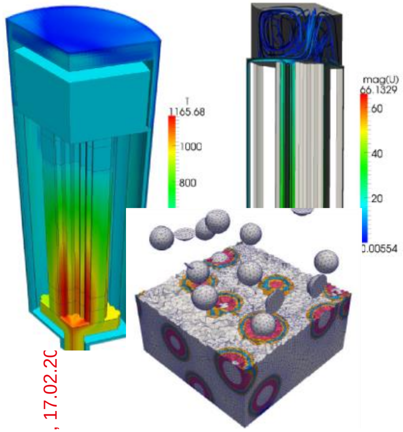
```
fvm::ddt(IV,flux_i])- fvm::laplacian(D,flux_i])= S
```

Use of OpenFOAM for multi-physics

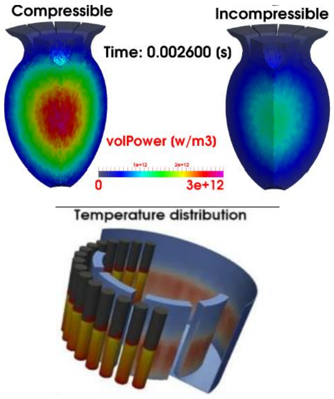
2000-2010
First activities

2010-2015
First widespread use

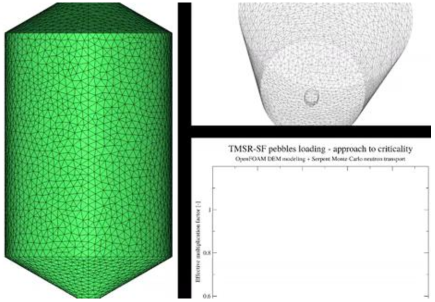
2015-2021
First coordinated and persistent developments



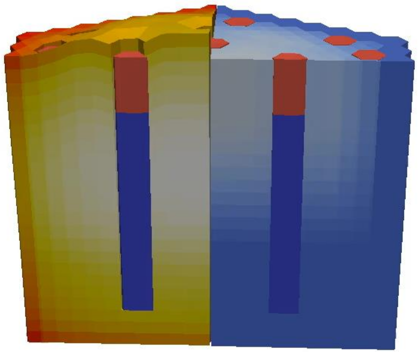
PBMRs and HTRs



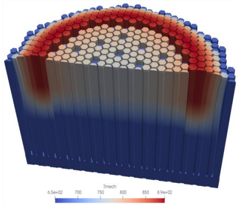
MSRs



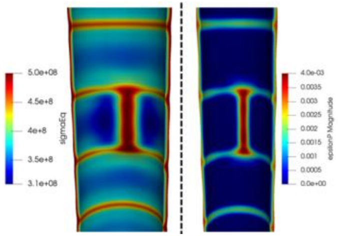
FHR



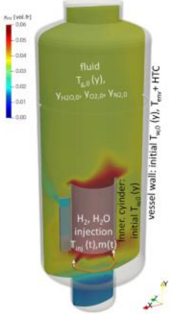
SFRs



GeN-Foam



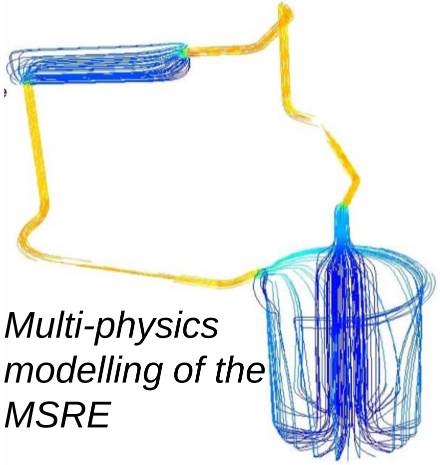
OFFBEAT



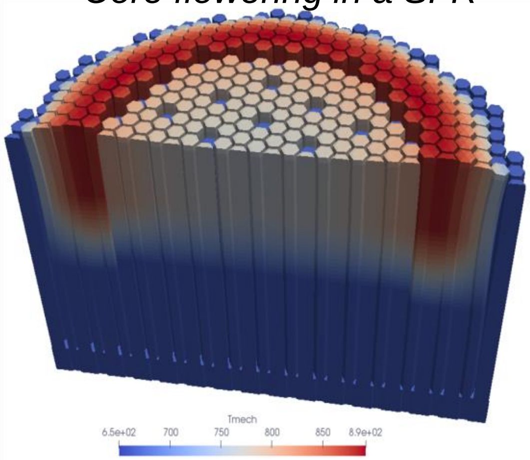
containmentFoam

The GeN-Foam multi-physics solver

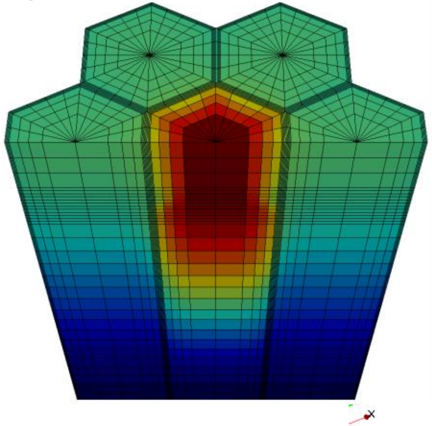
- General solver for reactor analysis, built upon previous efforts on HTRs ad MSRs
- Solves for: neutronics (point kinetics, diffusion, SP3, SN), single- and two-phase thermal-hydraulics (fine or coarse mesh), core deformations
- Developed to complement legacy codes with more flexibility (unstructured meshes, parallel scalability, implicit coupling, code tailoring)
- Beta version available on gitlab: https://gitlab.com/foam-for-nuclear/GeN-Foam/-/tree/master/



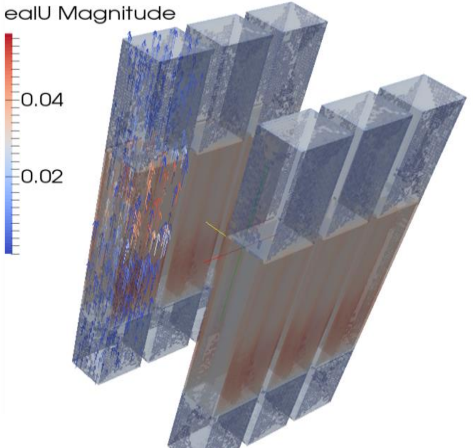
Multi-physics modelling of the MSRE



Core flowering in a SFR



Assembly windows in a SFR



The Argonaut reactor

Perspectives on open-source software

- Same “OpenFOAM” trends observed in all open-source community
    - Growing activities
    - Coordinated efforts
- Cross-platform interaction
- High-level coordination (ONCORE)

From scattered efforts to consistent platforms available to the community

Thank you

Carlo Fiorina

Dr. Kathryn D. Huff

Dr. Kathryn D. Huff is an Assistant Professor in the Department of Nuclear, Plasma, and Radiological Engineering at the University of Illinois at Urbana-Champaign where she leads the Advanced Reactors and Fuel Cycles Research Group. She is additionally a Blue Waters Assistant Professor with the National Center for Supercomputing Applications. She was previously a Postdoctoral Fellow in both the Nuclear Science and Security Consortium and the Berkeley Institute for Data Science at the University of California - Berkeley. She received her PhD in Nuclear Engineering from the University of Wisconsin-Madison in 2013 and her undergraduate degree in Physics from the University of Chicago. Her current research focuses on modeling and simulation of advanced nuclear reactors and fuel cycles. She is an active member of the American Nuclear Society, vice-chair of the Nuclear Nonproliferation and Policy Division, a past chair of the Fuel Cycle and Waste Management Division, and recipient of both the Young Member Excellence and Mary Jane Oestmann Professional Women's Achievement awards. Through leadership within Software Carpentry, SciPy, the Hacker Within, and the Journal of Open Source Software she also advocates for best practices in open, reproducible scientific computing.

Multiphysics Modeling Using the MOOSE Framework

Kathryn Huff
Advanced Reactors and Fuel Cycles Group
University of Illinois at Urbana-Champaign

IAEA Workshop on Multiphysics Modelling to Optimize Design and Safety of Advanced Nuclear Reactors

March 17, 2021

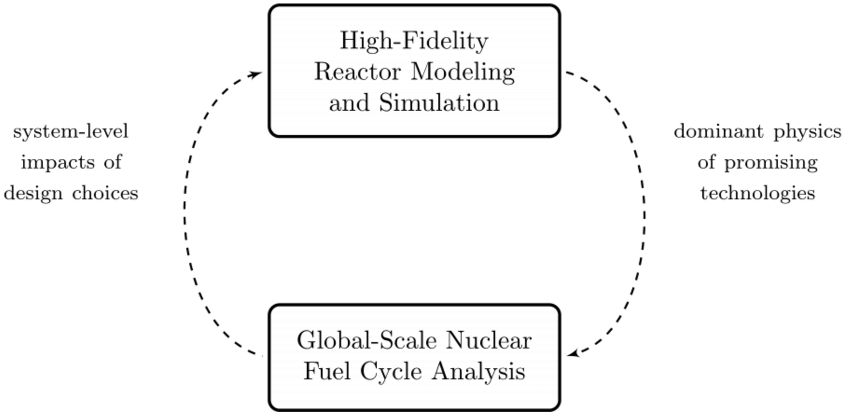
Outline

1. Different Problems, Different Solutions

2. MOOSE Framework

3. Moltres (a MOOSE Application)

Insights at Disparate Scales



High-Fidelity Reactor Modeling and Simulation

dominant physics of promising technologies

Global-Scale Nuclear Fuel Cycle Analysis

system-level impacts of design choices

Challenges in Liquid-Fueled Reactor Simulation

1. Contemporary burnup codes cannot treat fuel movement.
2. Neutron precursor locations drift before neutron emission.
3. Operational and safety parameters change during reactor operation.
4. Neutronics and thermal hydraulics are tightly interdependent.

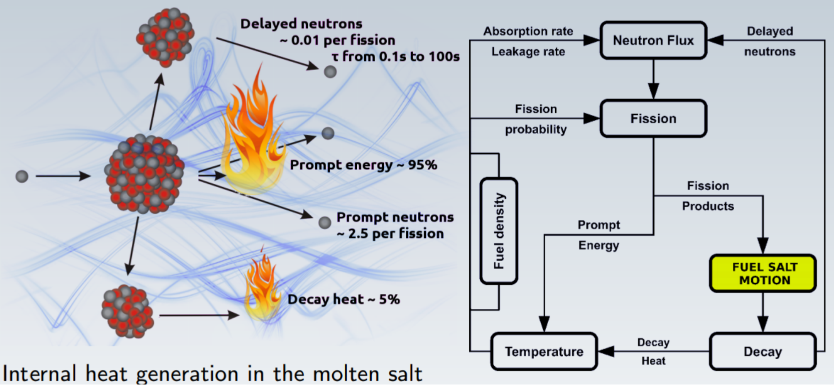


Figure: Challenges in simulating MSRs (Image courtesy of Manuele Aufiero, 2012).

Outline

1. Different Problems, Different Solutions
2. MOOSE Framework
3. Moltres (a MOOSE Application)

MOOSE Framework

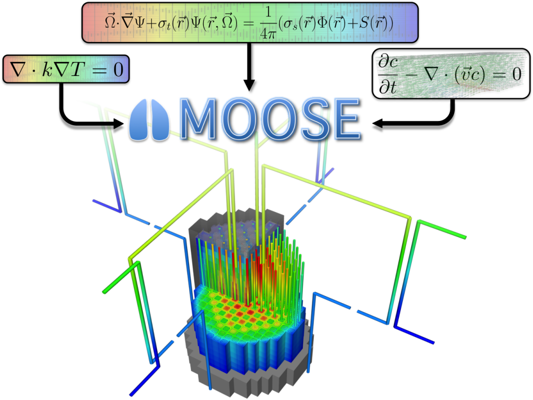


Figure: Multi-physics Object-Oriented Simulation Environment (MOOSE) [1, 5, 4].

- Developed by Idaho National Laboratory [1, 5, 4]
- Framework is truly open source (LGPL)
- Accessible docs and tutorials at `https://mooseframework.inl.gov`
- Source code at `https://github.com/idaholab/moose`

MOOSE Apps & Kernels

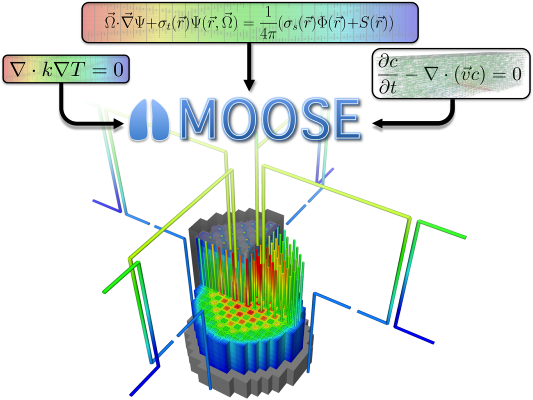


Figure: Multi-physics Object-Oriented Simulation Environment (MOOSE).

Lots of Open Kernels

- Chemical reactions
- Contact
- Fluid Properties
- Functional Expansion Tools
- Geochemistry
- Heat Conduction
- Level Set
- Navier-Stokes
- Peridynamics
- Phase Field
- Porous Flow
- Ray Tracing
- Reconstructed Discontinous Galerkin
- Tensor Mechanics
- ...

MOOSE Apps & Kernels

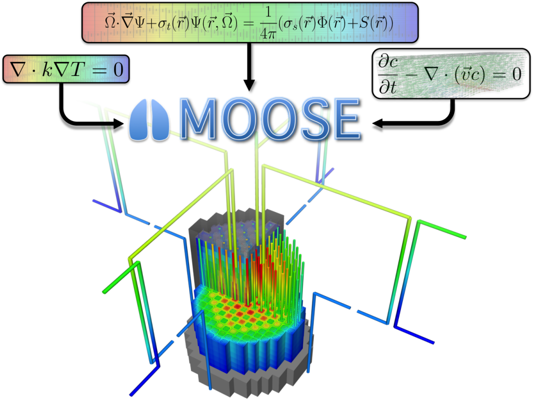


Figure: Multi-physics Object-Oriented Simulation Environment (MOOSE).

Lots of Open Apps

- Moltres (MSRs) [?]
- Squirrel (Utilities)
- Mastodon (structural dynamics, seismology)
- pika (microstructure)
- falcon
- blackbear
- crane (plasma chemistry)
- WhALE (fluid-structure mechanics)

MOOSE Apps & Kernels

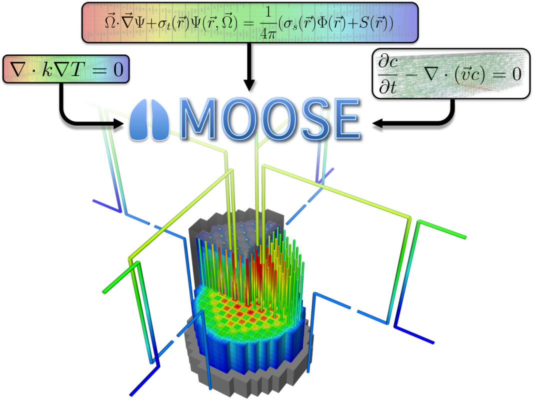


Figure: Multi-physics Object-Oriented Simulation Environment (MOOSE).

Lots of Restricted Apps

- BISON
- Marmot
- RattleSNake
- Pronghorn
- RELAP-7
- . . .

How Does it Work?

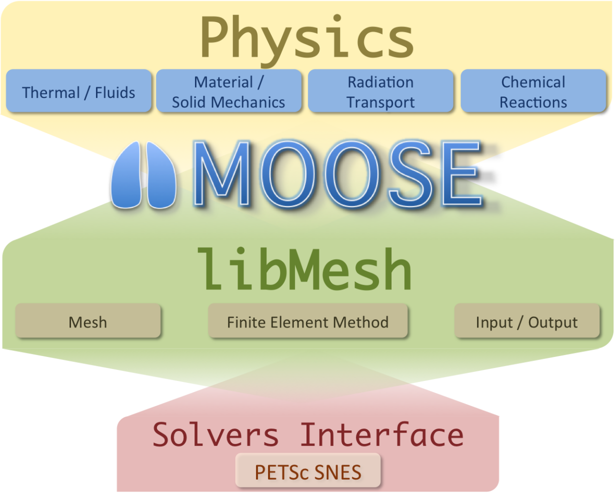


Figure: Shamelessly copied from the MOOSE Team Workshop slides.

How Does it Work?

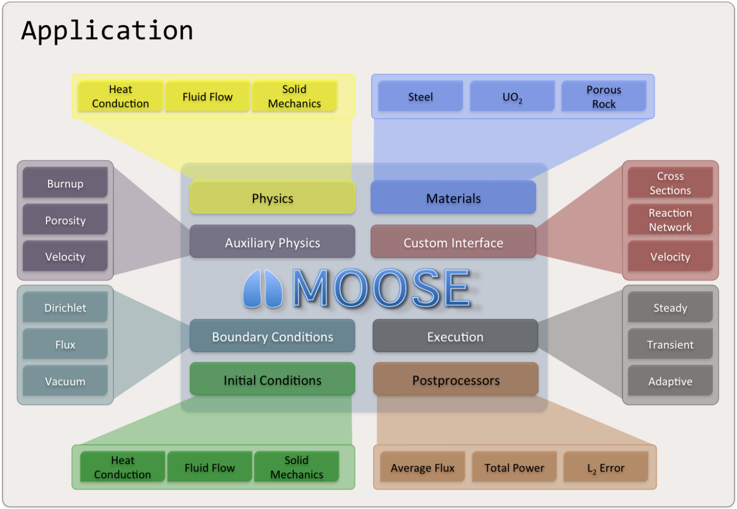
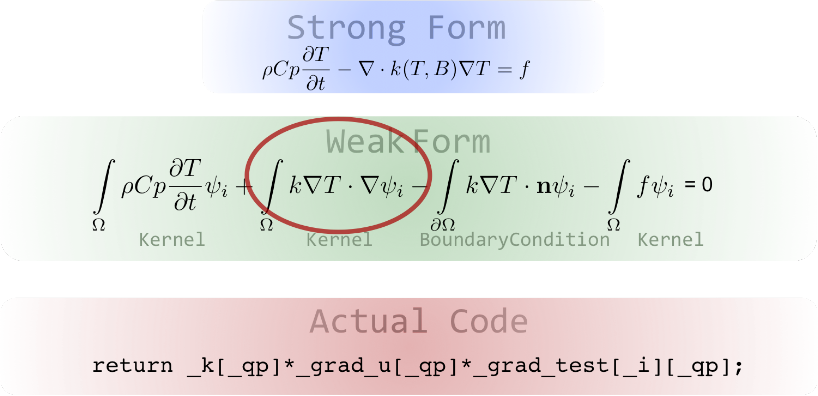


Figure: Shamelessly copied from the MOOSE Team Workshop slides.

How Does it Work?



Strong Form

$$\rho C p \frac{\partial T}{\partial t} - \nabla \cdot k(T, B) \nabla T = f$$

Weak Form

$$\underbrace{\int_{\Omega} \rho C p \frac{\partial T}{\partial t} \psi_i}_{\text{Kernel}} + \boxed{\underbrace{\int_{\Omega} k \nabla T \cdot \nabla \psi_i}_{\text{Kernel}}} - \underbrace{\int_{\partial\Omega} k \nabla T \cdot \mathbf{n} \psi_i}_{\text{BoundaryCondition}} - \underbrace{\int_{\Omega} f \psi_i}_{\text{Kernel}} = 0$$

Actual Code

```
return _k[_qp]*_grad_u[_qp]*_grad_test[_i][_qp];
```

Figure: Shamelessly copied from the MOOSE Team Workshop slides.

MOOSE: Key Features

- MOOSE Framework is truly open source (LGPL)
- Developed initially for nuclear applications
- Signficant long-term support from US DOE
- Continuous integration support (CIVET)
- Intuitive parallel multiscale solves
- Easy developer onboarding

- Object Oriented, C++
- Interfaces with libMesh to discretize simulation volume into finite elements
- Residuals and Jacobians handed off to PetSc which handles solution of resulting non-linear system of algebraic equations
- Fully-coupled, fully-implicit multiphysics solver
- Automatically parallel (largest runs >100,000 CPU cores!)
- Built-in adaptive meshing & timestepping

Pros and Cons

Pros (+)

- LGPL means the Framework is open, but apps can be restricted
- Vast array of available apps and kernels
- Many solver and preconditioning options
- Finite Element Modeling
- Full coupling is optional
- Generates gorgeous visualizations

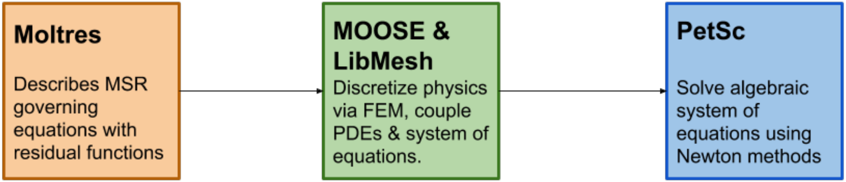
Cons (-)

- LGPL means the Framework is open, but apps can be restricted
- Vast array of available apps and kernels
- Many solver and preconditioning options
- Finite Element Modeling
- Full coupling is optional
- Generates gorgeous visualizations

Outline

1. Different Problems, Different Solutions
2. MOOSE Framework
3. Moltres (a MOOSE Application)

Moltres: Coupling in MOOSE



Moltres

Describes MSR governing equations with residual functions

MOOSE & LibMesh

Discretize physics via FEM, couple PDEs & system of equations.

PetSc

Solve algebraic system of equations using Newton methods

Moltres: Basics

- Developed in ARFC group
- Fluid-fuelled, molten salt reactors
- Multi-group diffusion (arbitrary groups)
- Advective movement of delayed neutron precursors
- Navier-Stokes thermal hydraulics
- 3D unstructured
- 2D axisymmetric
- 3D structured
- Initial developer: Alexander Lindsay [3]

Acquiring Moltres

```
git clone https://github.com/arfc/moltres
cd moltres
git submodule init
git submodule update
```

Diffusion in Moltres

$$\frac{1}{v_g}\frac{\partial \phi_g}{\partial t} - \nabla \cdot D_g \nabla \phi_g + \Sigma_g^r \phi_g = \tag{1}$$

$$\sum_{g \neq g'}^{G} \Sigma^s_{g' \to g} \phi_{g'} + \chi_g^p \sum_{g'=1}^{G} (1-\beta)\nu \Sigma^f_{g'} \phi_{g'} + \chi_g^d \sum_{i}^{I} \lambda_i C_i \tag{2}$$

v_g = speed of neutrons in group g
ϕ_g = flux of neutrons in group g
t = time
D_g = Diffusion coefficient for neutrons in group g
Σ_g^r = macroscopic cross-section for removal of neutrons from group g
$\Sigma^s_{g' \to g}$ = macroscopic cross-section of scattering from g' to g
χ_g^p = prompt fission spectrum, neutrons in group g

G = number of discrete groups, g
ν = neutrons produced per fission
Σ_g^f = macroscopic fission cross section due to neutrons in group g
χ_g^d = delayed neutrons in group g
I = delayed neutron precursor groups
β = delayed neutron fraction
λ_i = average decay constant of delayed neutron precursors in group i
C_i = concentration of delayed neutron precursors in precursor group i

.

Moltres Delayed Neutrons

$$\frac{\partial C_i}{\partial t} = \sum_{g'=1}^{G} \beta_i \nu \Sigma_{g'}^{f} \phi_{g'} - \lambda_i C_i - \frac{\partial}{\partial z} u C_i \tag{3}$$

$G =$ number of discrete groups, g

$I =$ delayed neutron precursor groups

$C_i =$ concentration of delayed neutron precursors in precursor group i

$.u =$ vertical fluid velocity

$\lambda_i =$ average decay constant of delayed neutron precursors in group i

$\beta =$ fraction of delayed neutron precursors in group i

Moltres Fuel Temperature

$$\rho_f c_{p,f} \frac{\partial T_f}{\partial t} + \nabla \cdot (\rho_f c_{p,f} \vec{u} \cdot T_f - k_f \nabla T_f) = Q_f \tag{4}$$

$$\rho_f = \text{density of fuel salt} \tag{5}$$

$$c_{p,f} = \text{specific heat capacity of fuel salt} \tag{6}$$

$$T_f = \text{temperature of fuel salt} \tag{7}$$

$$\vec{u} = \text{velocity of fuel salt} \tag{8}$$

$$k_f = \text{thermal conductivity of fuel salt} \tag{9}$$

$$Q_f = \text{source term} = \sum_{g=1}^{G} \epsilon_{f,g} \Sigma_{f,g} \phi_g \tag{10}$$

Moltres Moderator Temperature

$$\rho_g c_{p,g} \frac{\partial T_g}{\partial t} + \nabla \cdot (-k_g \nabla T_g) = Q_g \tag{11}$$

$$\tag{12}$$

$$\rho_g = \text{density of graphite moderator} \tag{13}$$

$$c_{p,g} = \text{specific heat capacity of graphite moderator} \tag{14}$$

$$T_g = \text{temperature of graphite moderator} \tag{15}$$

$$k_g = \text{thermal conductivity of graphite moderator} \tag{16}$$

$$Q_g = \text{source term in graphite moderator} \tag{17}$$

$$\tag{18}$$

Moltres MSRE Simulation

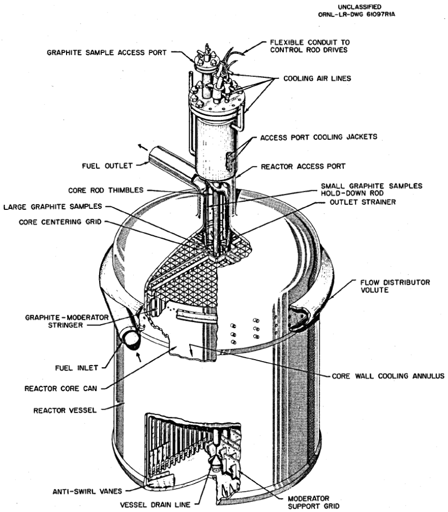
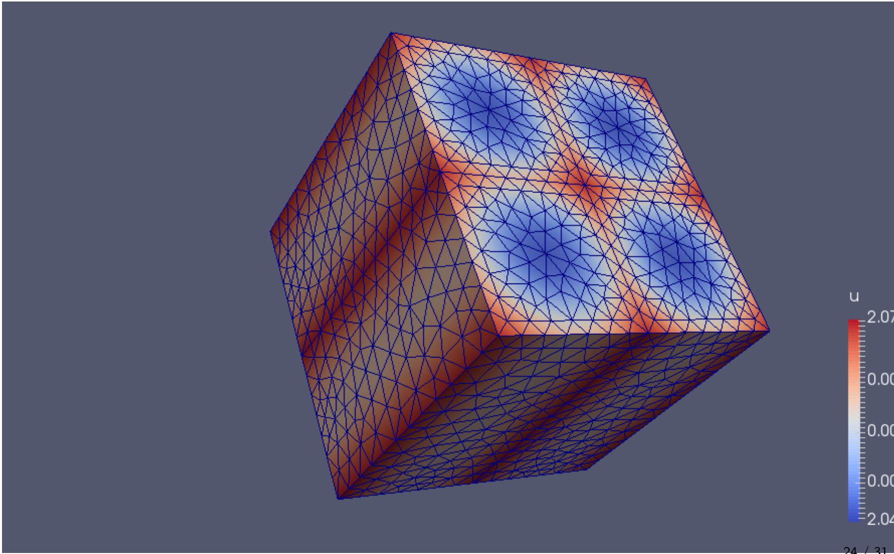
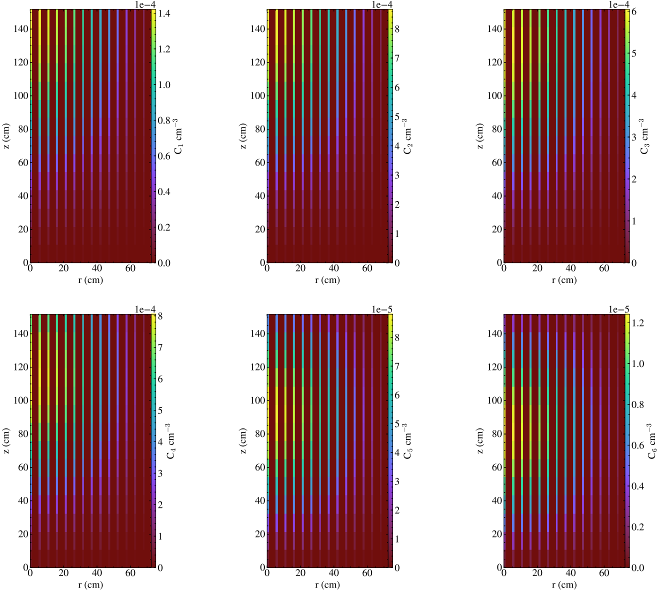


Fig. 6. MSRE Reactor Vessel.

Mesh Generation for MOOSE Apps like Moltres



Moltres Precursor Drift



Moltres: More Complex Mesh

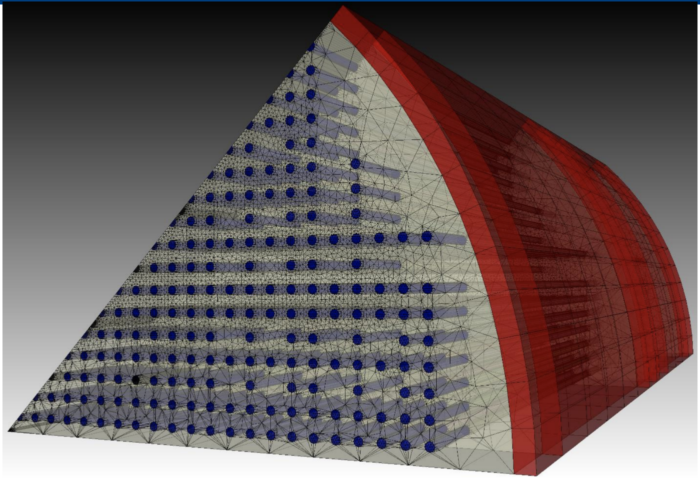


Figure: TAP Mesh generated by Alvin Lee [2]. Red = Reactor Vessel Wall, Light Yellow = Fuel Salt, Dark Gray = Control Rods, Blue = Fuel Salt radially co-located with the Moderator Rods.

Moltres: Multiphysics simulation (3D)

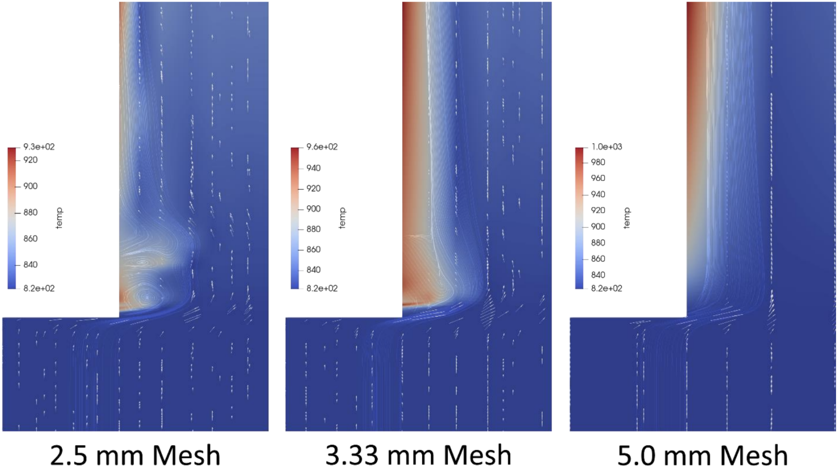


Figure: Meshing study by Alvin Lee [2] regarding KH instabilities and resolution of MSR fuel salt vortices.

What now?

https://github.com/idaholab/moose
https://github.com/arfc/moltres
https://mooseframework.inl.gov/

Acknowledgements

- This research is part of the Blue Waters sustained-petascale computing project, which is supported by the National Science Foundation (awards OCI-0725070 and ACI-1238993) and the state of Illinois.
- Kathryn Huff is additionally supported by the NRC Faculty Development Program, the NNSA (awards DE-NA0002576 and DE-NA0002534), and the International Institute for Carbon Neutral Energy Research (WPI-I2CNER).
- The authors would like to thank members of Advanced Reactors and Fuel Cycles research group (ARFC) at the University of Illinois at Urbana Champaign who provided valuable code reviews and proofreading.
- This work is derived from the work of Alex Lindsay (Idaho National Laboratory), Gavin Ridley (University of Tennessee-Knoxville), Andrei Rykhlevskii (Argonne National Laboratory), Sun Myung Park (UIUC), and Alvin Lee (UIUC).

References I

[1] Derek R. Gaston, Cody J. Permann, John W. Peterson, Andrew E. Slaughter, David Andrš, Yaqi Wang, Michael P. Short, Danielle M. Perez, Michael R. Tonks, Javier Ortensi, Ling Zou, and Richard C. Martineau.
Physics-based multiscale coupling for full core nuclear reactor simulation.
Annals of Nuclear Energy, 84:45–54, October 2015.

[2] Alvin J. H. Lee.
Neutronics and Thermal-Hydraulics Analysis of TransAtomic Power Molten Salt Reactor (TAP MSR) Core Under Load Following Operations.
Master's thesis, University of Illinois at Urbana-Champaign, Urbana, IL, December 2020.

[3] Alexander Lindsay, Gavin Ridley, Andrei Rykhlevskii, and Kathryn Huff.
Introduction to Moltres: An application for simulation of Molten Salt Reactors.
Annals of Nuclear Energy, 114:530–540, April 2018.

[4] Alexander Lindsay, Roy Stogner, Derek Gaston, Daniel Schwen, Christopher Matthews, Wen Jiang, Larry K Aagesen, Robert Carlsen, Fande Kong, Andrew Slaughter, et al.
Automatic differentiation in metaphysicl and its applications in moose.
Nuclear Technology, pages 1–18, 2021.

References II

[5] Cody J. Permann, Derek R. Gaston, David Andrš, Robert W. Carlsen, Fande Kong, Alexander D. Lindsay, Jason M. Miller, John W. Peterson, Andrew E. Slaughter, Roy H. Stogner, and Richard C. Martineau.

MOOSE: Enabling massively parallel multiphysics simulation.

SoftwareX, 11:100430, 2020.

Q&A Session

Christophe Demaziere
Professor, Chalmers University of Technology, Sweden

Carlo Fiorina
Scientist, École polytechnique fédérale de Lausanne (EPFL), Switzerland

Kathryn D. Huff
Assistant Professor, University of Illinois at Urbana-Champaign, USA

Webinar Series on Nuclear Technology Breakthroughs for the 21st Century

IAEA
International Atomic Energy Agency

www.iaea.org/nptd-webinars