

© 2023 Roberto E. Fairhurst-Agosta

ENHANCED METHOD FOR THE SUPPORT OF EXPERIMENT SAFETY ANALYSIS

BY

ROBERTO E. FAIRHURST-AGOSTA

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Nuclear, Plasma, and Radiological Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2023

Urbana, Illinois

Doctoral Committee:

Professor Tomasz Kozlowski, Director of Research, Chair
Professor Rizwan Uddin
Research Scientist Madicken Munk
Professor Paul Fischer

Abstract

Nuclear research reactors enable a wide range of applications and support the development of multiple fields, including energy production, isotopic production, space discovery, among others. Research reactor safety analyses demonstrate the contribution of the operational procedures to the prevention of accidents, focusing mainly on the reactor core. These analyses rely on the assumption of heat being locally deposited in the core after shutdown. This work, however, is concerned with research reactor experiments, and it investigates more detailed methods targeting the specific reactor regions hosting the experiments.

Additionally, research reactor experiments require the development of individual experiment safety analyses, which due to the lack of a streamlined workflow becomes effort and time-consuming as well as error-prone. This thesis supports the development of safety analyses for research reactor experiments with the objective of streamlining the calculation procedure as well as accelerating it. The methodology presented here calculates the delayed heating in experiments, generates the volumetric heat source term for thermal-fluids calculations, solves the experiment temperature evolution during a channel draining event, and determines if the event leads to a radioactive material release. Moreover, this work examines methods to accelerate the calculation, which include the creation of a generic irradiation database and data-based modeling.

This thesis starts with a literature survey summarizing previous work relevant to delayed heating calculations, transport/depletion solvers, and machine learning applied to nuclear engineering. The conclusions of the survey guide the choice of methods for the delayed heating calculation and the choice of machine learning algorithms for the data-based modeling.

The following chapters focus on the calculation workflows. This thesis establishes a delayed heating calculation workflow based on the formal three-step method that builds on top of the MCNP-ORIGEN Activation Automation (MOAA) tool. This workflow is verified and demonstrated with several exercises, which also ease the workflow visualization. The workflow is also demonstrated in two full-size research reactors, for which the heat in the structures and an experiment are calculated.

A simple modification of the delayed heating calculation workflow generates a shutdown dose rate calculation workflow as a value-added. This calculation workflow helps guide the post-irradiation examination (PIE) of experiments and helps prevent the unnecessary exposure of personnel and surrounding equipment. The calculation workflow is demonstrated for a Tristructural Isotropic (TRISO)-fueled experiment which highlights the applicability of this workflow to more complex geometries, such as High-Temperature Gas-Cooled Reactors (HTGRs).

When focusing on the acceleration of the calculations, this thesis proposes the generic irradiation database method. This method relies on the delayed heating calculation workflow to obtain the heat produced in experiments of individual chemical elements and through their combination calculate the heat produced in experiments of arbitrary material composition. This thesis verifies this method by studying a vast range of materials that could be used in a research reactor experiment.

Moreover, this thesis investigates the applicability of data-based modeling to predict the outcome of the most thermally challenging accident in an experiment, a channel draining event. The data preparation process and several machine learning algorithms are described before studying a simplified model. This simplified model allows drawing several useful conclusions before scaling it up, with some minor modifications, to a full-size reactor experiment.

Finally, this thesis wraps up with the conclusions from all the previous calculations. Overall, the results presented in this thesis meet this thesis' objectives and prove that the proposed methodology decreases the computational expense and enables quick calculations supporting research reactor experiments.

A mi familia, for their unconditional and unwavering support.

Acknowledgments

Pursuing a Ph.D. is like a roller coaster of emotions through which we build knowledge and advance the state-of-the-art of a certain field. I learned a lot during these years, wherein I became a better researcher and gained a deeper understanding of how the academic world works, while I increased my knowledge of this engineering field that I love. This Ph.D. thesis is the product of countless cups of coffee and innumerable discussions that can be wrapped up into five and a half years of work, and it would not have been possible without the support of numerous people that I will try to acknowledge here.

First and foremost, I would like to thank my Ph.D. advisor Dr. Tomasz Kozlowski for welcoming me into his research group, the Analysis of Reactor Transients and Stability Group (ARTS). I have always esteemed Tomasz and his broad and deep expertise in the fundamental physics that defines our field. Hopefully, I was able to learn some things from him and the long and fruitful discussions that contributed to my work. Second, I would like to thank my former advisor Dr. Kathryn Huff for allowing me to be part of the Advanced Reactors and Fuel Cycles Group (ARFC) and for her thorough guidance in my academic beginnings during my early years in graduate school. Next, I would like to thank the members of my doctoral committee, Dr. Madicken Munk, Dr. Rizwan Uddin, and Dr. Paul Fischer for the thorough input and guidance they provided in assessing this work. I would like to specially thank Madicken and Paul for their kind words, support, and very enlightening questions and remarks during my preliminary examination.

Additionally, part of my learning experience happened during several summer internships at national laboratories. I am very grateful I was part of the Irradiation Experiment Neutronics Analysis group at the Idaho National Laboratory for two consecutive summers. The seed idea for this work came from discussions with Dr. Joshua Peterson-Droogh, for whom I am very thankful. I also would like to acknowledge my internship mentor Austin Carter, from whom I learned a lot about software development and quality assurance. And also I would like to acknowledge him as the main developer of the MCNP-ORIGEN Activation Automation tool (MOAA), one of the major components of this work. Idaho felt like home thanks to Josh, Austin, and Clinton Wilson, who taught me that there is an important human factor in research. During my last summer internship, I was part of the System Analysis and Methods group at the Argonne National Laboratory, where I was working under Dr. Zhiee J. Ooi's mentorship along with Dr. Ling Zhou and Dr. Rui Hu. I am very thankful for this opportunity, wherein I was able to collaborate with researchers of their caliber and to learn the true meaning of teamwork.

Besides the support from several professors and mentors, this work was possible due to day-to-day motivation from my fellow group-mates Alvin Lee, Silvana Tabares-Burgos, Harun Ardiansyah, and Yifan Mao, also from my former group-mates Sam Dotson, Gwen Chee, and Sun Myung Park, as well as my graduate school friends Jeremy Mettler and Erik Smith. I also want to highlight my family: Roberto, Ana, Leo, Vicky, and Euge, who even though are well over five thousand miles apart, their love, support, and encouragement were always present. Finally, I would like to acknowledge Nicole Veit for her support during

my last semester of graduate school and for giving me the strength, courage, and motivation necessary to finish this work.

This work was conducted in ARTS group at the Nuclear, Plasma, and Radiological Engineering (NPRE) Department at the University of Illinois at Urbana-Champaign (UIUC). This research made use of the Illinois Campus Cluster, a computing resource that is operated by the Illinois Campus Cluster Program (ICCP) in conjunction with the National Center for Supercomputing Applications (NCSA), which is supported by funds from UIUC. This research also made use of the resources of the High Performance Computing Center at the Idaho National Laboratory (INL), which is supported by the Office of Nuclear Energy of the Department of Energy (DOE) and the Nuclear Science User Facilities under Contract No. DE-AC07-05ID14517.

Table of Contents

Chapter 1 Introduction	1
1.1 Motivation	2
1.2 Objectives	4
1.3 Outline	6
Chapter 2 Literature Review	7
2.1 Delayed Heating	7
2.2 Transport/Depletion Solvers Background	10
2.3 Machine Learning	12
Chapter 3 Delayed Heating	15
3.1 MOAA	15
3.1.1 Calculation Workflow	15
3.1.2 ATR cases	17
3.1.3 Quality Assurance and Version Control	18
3.1.4 Simulation Specification	19
3.2 Delayed heating	20
3.2.1 Calculation Workflow	20
3.2.2 Simulation Specification	23
3.3 Results	23
3.3.1 Delayed Gamma in a Uranium Sphere	23
3.3.2 Demonstration Problem	26
3.3.3 ATR experiment	27
3.3.4 ATR results	30
3.3.5 RA-6 structures	30
3.3.6 RA-6 results	35
3.4 Conclusions	35
Chapter 4 Shutdown Dose Rate	38
4.1 Methodology	39
4.2 Verification	40
4.3 AGR-1	43
4.4 The μ HTGR	45
4.5 Conclusions	51
Chapter 5 Generic Irradiation Database	53
5.1 Fast Calculation	53
5.2 Generic Material Irradiation Database	58
5.2.1 Demonstration exercise	58
5.2.2 ATR experiment	62
5.3 Conclusions	65
Chapter 6 Thermal-Fluids and Machine Learning	66
6.1 Thermal-Fluids Solvers	66
6.1.1 MOOSE	66
6.1.2 OpenFOAM	67

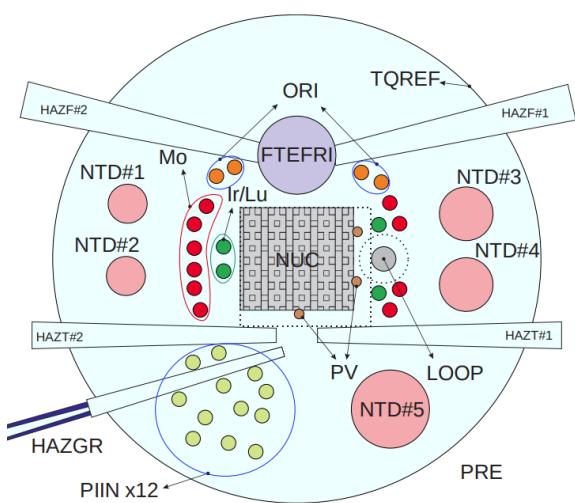
6.2	Machine Learning	68
6.2.1	Support Vector Machine	68
6.2.2	Decision Tree	69
6.2.3	Random Forest	70
6.2.4	K-Nearest Neighbors	70
6.2.5	Logistic Regression	71
6.2.6	Feed-forward Neural Networks	71
6.2.7	Long-Short Term Memory Networks	73
6.2.8	Optimization	74
6.3	Simplified Model	76
6.3.1	Model verification	77
6.3.2	Data Processing and Model Training	79
6.3.3	Strategy Comparison	84
6.3.4	Non-dimensional thermal properties	87
6.3.5	Synthetic Materials	87
6.4	Source Decomposition	90
6.5	ATR experiment	95
6.5.1	Results	100
6.6	Conclusions	105
Chapter 7	Conclusions	107
References		110
Appendix A	MOAA execution	120
A.1	User Input File	120
A.2	ATR case	121
A.3	Executing MOAA	122
A.4	Simulation output	122
Appendix B	Execution of Delayed Heating Calculation Workflow	124
B.1	Batch Size	124
Appendix C	Serpent Gamma Source	127
Appendix D	Experiment Material Properties for Neutronics Calculation	129
Appendix E	Experiment Material Properties for Thermal-Fluids Model	136

Chapter 1

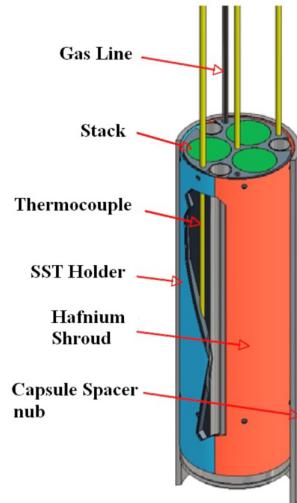
Introduction

Research reactors are nuclear reactors that work as neutron and gamma sources for research and various applications. In contrast to Nuclear Power Plants (NPPs), they are not used to produce heat for electricity generation, and consequently, they are smaller. Their power ratings are smaller too compared to NPPs, typically ranging from zero up to 200 MWth [1]. Smaller sizes and power ratings make them less complex with simpler systems and designs that make them well suited for education and training of reactor operators, radiation protection personnel, students, and researchers.

Research reactors enable a vast range of uses and purposes, including scientific research, material testing, and radioactive material production. An example of scientific research is the utilization of thermal and cold neutron facilities, as in the Argentinian reactor RA-10 shown in Figure 1.1, to study the structure of matter via neutron scattering. Knowing the internal structure of matter allows to understand the macroscopic properties of a material. By performing neutron scattering, biologists understand the repair and decay processes in the bones, physicists create more powerful magnets, and neuron experts study proteins in the brain, just to mention a few areas of research [1].



(a) Scientific research in the Argentinian reactor RA-10 [2].



(b) Material testing of the Advanced Gas Reactor 1 (AGR-1) capsule [3].

Figure 1.1: Examples of research reactor uses.

Material characterization and testing allows to study the effects of radiation on the material properties. Neutrons induce changes in the materials' structure, making them fragile, elastic or hardened, swell, crumble, change their composition, release gas, etc. Moreover, research reactors test advanced reactor fuel which supports their design optimization. The AGR-1 experiment [3], shown in Figure 1.1, was an irradiation test to provide performance data during normal operation of the fuel utilized in HTGRs.

Another frequent use of research reactors is radioactive material production, which encompasses three main applications: neutron activation analysis, neutron transmutation doping, and radioisotope production for medicine. Figure 1.2 shows some examples of these applications. Neutron activation analysis is a technique to measure the elemental composition of water, air, soil, rocks, etc. The samples are irradiated and by studying their characteristic gamma emissions it is possible to identify trace elements in the range of parts per billion. Neutron transmutation doping changes silicon properties, making it conductive, and the electronics industry highly benefits from this application for chip production. Finally, the medical field utilizes radioisotopes for diagnosing and treating health conditions like cancer and cardiovascular diseases. For example, technetium-99m comes from the decay of molybdenum-99, and its primary use is diagnostic imaging [4].

1.1 Motivation

The International Atomic Energy Agency (IAEA) Safety Fundamentals state that the main safety objective in all nuclear installations is to protect people and the environment by establishing an effective defense against radiological hazards [7]. Fulfilling this objective requires the quantification of the risk of radiation release to the environment that may arise from a nuclear reactor. Safety analyses assess the performance of the reactor against a broad range of operating conditions and demonstrate the prevention of radioactive material releases. They can also identify hidden weaknesses and guide design improvements. Although research reactors have a much smaller size and power than NPPs, safety analyses are still required.

Research reactor safety analyses demonstrate to the regulatory body the contribution of the operational procedures to the prevention and mitigation of accidents. However, these analyses focus on the reactor core and disregard a thorough examination of the experiments. This work defines an experiment as any device utilized in any of the aforementioned research reactor applications. The development of experiment safety analyses is paramount to ensure safe experiment activities during any planned or unplanned events. Accordingly, the IAEA recommends that experiment safety analyses include a list of possible hazards and any operational limits, such as maximum temperatures and pressures [8].

Research reactors enable the irradiation of a wide variety of experiments. For example, they may have different combinations of shapes and materials. Therefore, experiments tend to be unique, requiring independent calculations and the development of safety analyses on a case-by-case basis. In consequence, the development of individual analyses often tends to be effort- and time-consuming. For instance, the initial experiment definition, design, and fabrication of a simple static capsule experiment for the Advanced Test Reactor (ATR) takes six months [9]. Nevertheless, using previously analyzed configurations can shorten this process. Another example is the Nuclear Science User Facility (NSUF) of the High Flux Isotope Reactor (HFIR), wherein the experiment proposal review process requires the experiment materials and parameters to fall within the bounds of previously approved experiments [10]. For experiments falling outside these bounds, additional procedures are required, translating to longer processing times. This reinforces the idea that using previous knowledge can expedite the process.

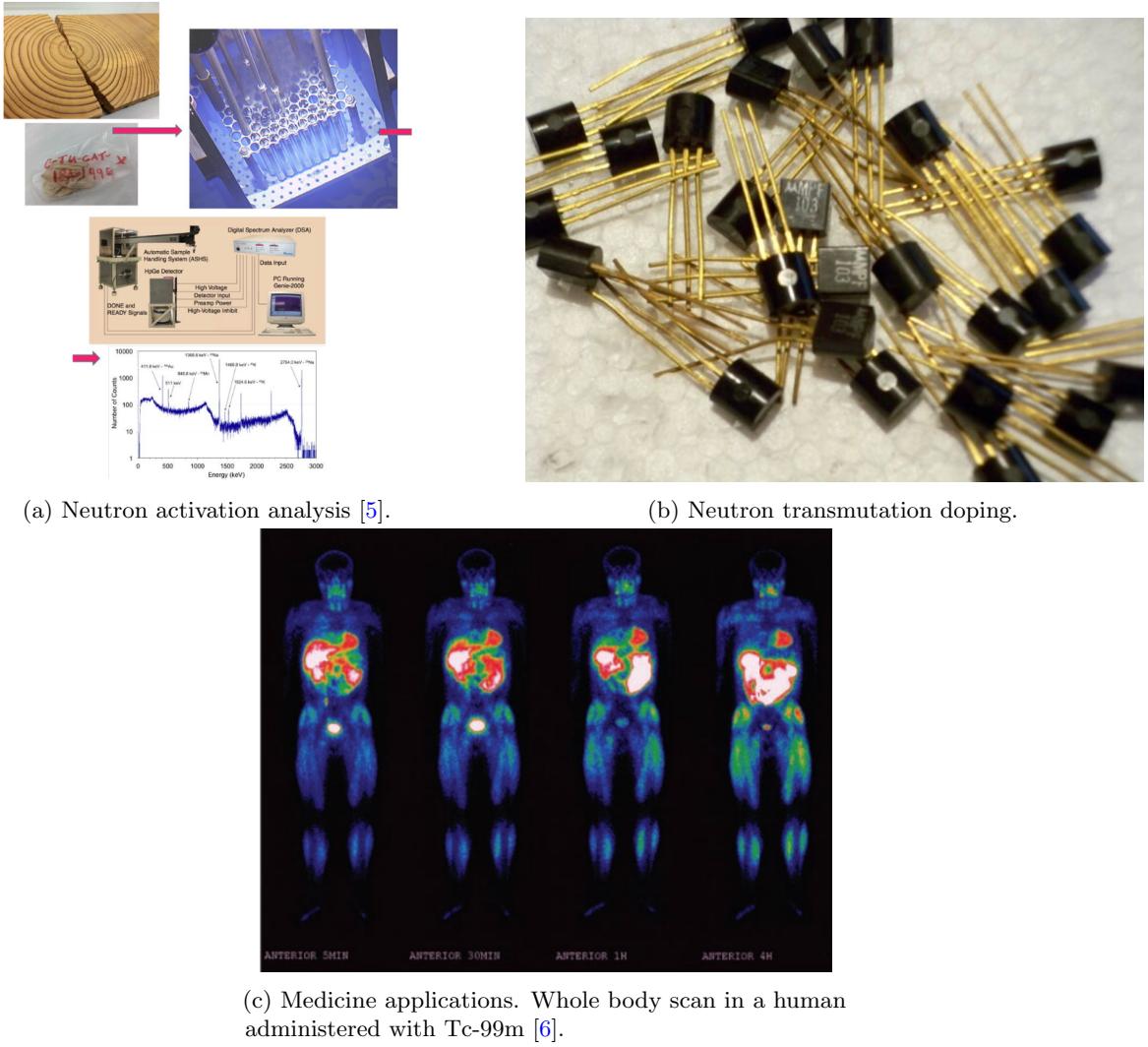


Figure 1.2: Examples of applications of radioactive material production.

The present work supports the development of experiment safety analysis, and, to narrow down the scope of the analysis, it studies the bounding event - i.e., the most thermally challenging scenario that an experiment could undergo within the design basis accidents, a channel draining event. The section on the left of Figure 1.3 shows the standard workflow for the evaluation of the experiment state during the bounding event. The workflow consists of a delayed heating calculation and the computational fluid dynamics (CFD) modeling of the experiment. The delayed heating calculation relies on a three-step process. First, a neutron transport simulation determines the flux distribution. Second, a depletion simulation determines the material composition, decay rates, and photon energy distribution. Third, a photon transport simulation estimates the energy deposition in the experiment. The subsequent run and exchange of information between these steps and the CFD simulation is commonly carried out manually by the lack of a streamlined workflow. In consequence, the development of experiment safety analyses tends to not only be effort- and time-consuming, but also error-prone, as each of these steps introduces an instance for an error to occur.

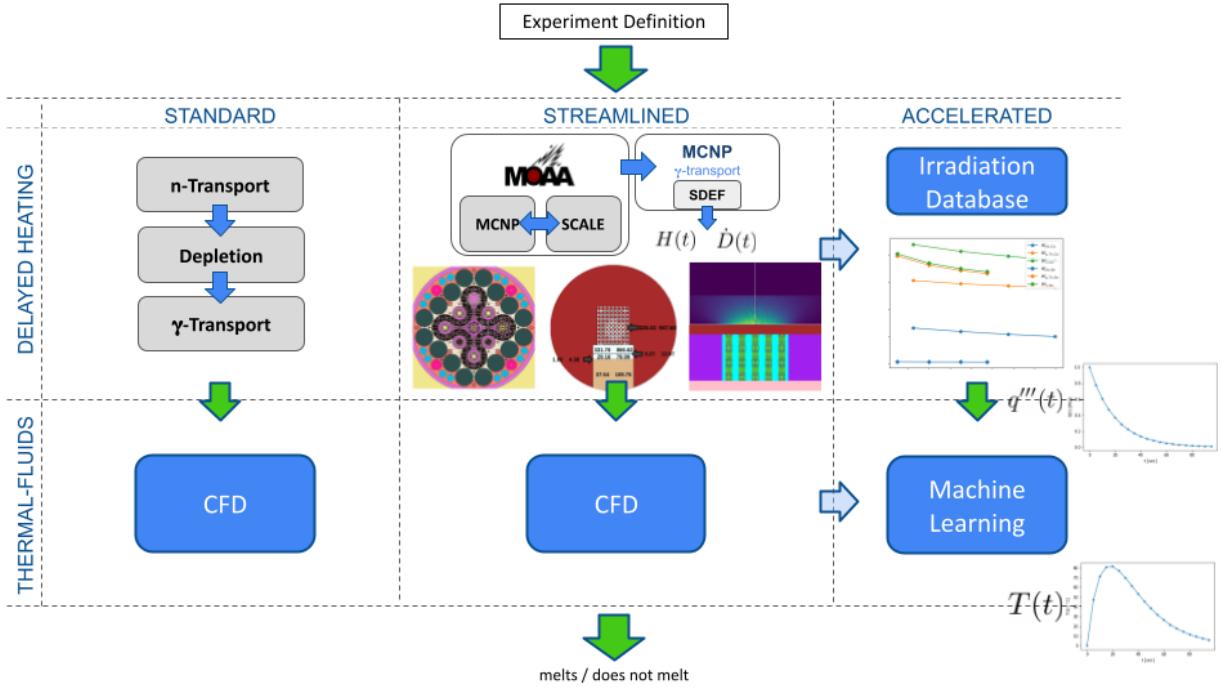


Figure 1.3: Calculation procedure to determine experiment state during a channel draining event.

1.2 Objectives

The objective of this Ph.D. thesis is to reduce the effort and time invested in the development of experiment safety analyses and to lower the chance of errors. This objective can be separated into two parts. The first part streamlines the calculation workflow, and the second part accelerates it by integrating previous knowledge into the process. The section in the center of Figure 1.3 provides a visual representation of the streamlined workflow that fulfills the requirements of the first part of the objective. The streamlined workflow automates the transfer of information within the delayed heating calculation workflow as well as to the CFD simulation. This objective can be organized into the following main subtopics.

Study the capsule's integrity during a channel draining event. Specifically, this work develops

modeling capabilities to quickly predict the outcome of the bounding event on the experiment and if it leads to a radioactive material release. Additionally, the range of applicability of these capabilities should encompass any experiment material composition. The present work considers the melting of the capsule encasing the experiment as the threshold for the radioactive material release. The accident progression considers the following events: the safety system stops the fission reactions in the core, the experiment channel loses all the coolant, air fills up the remaining space, and the resulting natural circulation of air cools down the experiment. The study takes a conservative approach and considers that the channel is drained instantly after shutdown.

Create a CFD model of the experiment. Effective heat removal prevents overheating damage that may lead to a radioactive material release. Quantifying the capsule cooling requirements, and hence ensuring the capsule's integrity, requires the development of a thermal-fluids model to accurately predict the capsule temperature.

Develop delayed heating calculation capabilities. In an operating reactor, there is an equilibrium between the generated and removed heat. After a reactor shutdown, the radioactive materials continue to decay and release energy. This energy is deposited across the reactor, and it is the cause of the delayed heating. The assessment of the energy deposited in the experiment provides the thermal-fluids model with the volumetric heat source to accurately calculate the capsule temperature. The delayed heating calculation workflow not only allows calculating the heating in the reactor experiments but also in reactor structures, a new feature that will be demonstrated.

Shutdown-dose rate calculations. After the experiment has been extracted from the reactor, the ongoing decay exposes workers and equipment in the surroundings to radiation dose. Safety analyses require the quantification of this dose. A value-added of the delayed heating calculation is the shutdown-dose rate, which removes the need for additional workflow for radiation dose rates and will be demonstrated for the post-irradiation examination of an experiment.

The workflow described above streamlines the calculation procedure of experiment safety analyses. The section on the right of Figure 1.3 shows the accelerated workflow that intends to decrease the calculation burden significantly, which fulfills the requirement of the second part of this Ph.D. thesis objective. The first part of the accelerated calculation is the delayed heating in the experiment, which relies on the use of the delayed heating calculation of the streamlined workflow. This step entails the creation of a generic material irradiation database with which it is possible to estimate the delayed heating of experiments of any material composition. The second component of the accelerated calculation is the determination of the bounding event outcome. This part uses the CFD calculation of the streamlined workflow to train and test several machine-learning algorithms and examine their applicability to further lower the calculation burden. This objective can be organized into the following main subtopics.

Generate a delayed heating database. Research reactors fulfill the radioisotope needs from multiple fields. Consequently, the range of materials that form the experiments is vast. For example, the medical field requires a supply of cobalt-60, yttrium-90, molybdenum-99, iodine-131, and iridium-192. Therefore, this work aims to create a model extensible to all possible irradiated materials. The creation of the database involves utilizing the delayed heating calculation scheme to obtain the delayed heating for experiments composed of individual chemical elements. This work creates a method that uses such a database to calculate the delayed heating in experiments of arbitrary composition. This method will be verified for selected materials.

Build a data-based model. The thermal-fluids model allows for the determination of the capsule temperature for an experiment. However, running the thermal-fluids model for each individual experiment is computationally expensive. The data-based model utilizes several machine learning to further lower

the computational burden and quickly predict the capsule temperature. The present work compares their performance and examines their applicability to experiment safety analysis. Ultimately, the data-based model aims to predict if the experiment capsule melts for any experiment material composition, provided its thermal properties are known. Two main approaches were chosen. The first approach relies on classification methods to predict the state of the experiment. The second approach uses deep-learning to predict the temperature evolution of the experiment.

Demonstrate the modeling capability by its application to ATR. The fulfillment of this thesis' objectives is complete with the application, demonstration, and verification of the presented developments on a full-size research reactor. This thesis focuses on the ATR, and this objective consists of the following steps. Create a detailed delayed heating database for an ATR experiment. Tailor the thermal-fluids model to the experiment. Train and test a data-based model specific to the experiment.

1.3 Outline

This thesis describes the motivation, objectives, methodology, results, and conclusions towards the advancement of the state-of-the-art of research reactor experiment analysis. The remainder of this document is organized as follows. Chapter 2 presents a literature review organizing and summarizing previous relevant work to this thesis. Chapter 3 describes the delayed heating calculation workflow and demonstrates it for multiple examples. Chapter 4 focuses on the shutdown dose rate calculation workflow and several demonstration cases. Chapter 5 introduces the generic material irradiation database method, and its verification with selected materials commonly used in nuclear engineering. Chapter 6 describes the utilized machine learning algorithms, the implementation strategies, and the train and test of different exercises. Chapter 7 summarizes the contributions of this thesis to the nuclear engineering field.

Chapter 2

Literature Review

2.1 Delayed Heating

The state of the reactor core during an accident is the main focus of research reactor safety analyses. Accordingly, numerous studies have targeted diverse accident conditions affecting research reactors, such as the loss-of-coolant accident (LOCA) and the loss-of-flow accident (LOFA). Bajs et al. [11] studied the performance of the emergency systems of the International Reactor Innovative and Secure (IRIS) reactor during a LOCA concluding that the reactor core remains at safe temperatures and covered at all times during the event. Kazeminejad et al. [12] investigated the peak fuel and cladding temperatures of a 10-MW IAEA research reactor after a LOFA with flow inversion. Hainoun et al. [13] introduced a benchmark exercise for thermal-hydraulic solvers to study the LOFA in the IEA-R1 research reactor, where the metrics of interest were the coolant and cladding temperatures, reactor power, and flow rates of different positions in the core. Abdelmaksoud et al. [14] examined the possibility of passively cooling by natural convection and radiation the reactor core of a typical research reactor that is completely uncovered after a LOCA. All these studies rely on different versions of the ANSI/ANS standard [15] for estimating the decay heat after shutdown, assuming that most of the energy from radioactive decay is deposited locally in the region of interest: the reactor core.

Additionally, many software packages, such as Serpent [16] and ORIGEN [17], estimate the decay heat by assuming the local deposition of heat, and diverse studies relied on the same assumption. Lee et al. [18] used ORIGEN-2 to compare the decay heat in a HTGR core containing transuranic and uranium fuel. Ilas et al. [19] calculated the decay heat in the HFIR irradiated fuel using ORIGEN-2.2. Hawkes et al. [20] evaluated with ORIGEN-2.2 the thermal behavior of the AGR-2 experiment in the ATR, an experiment fueled with TRISO particles. However, an accurate assessment of the deposited energy across the reactor geometry allows to better determine the heat removal requirements and ensure an effective cool down. Moreover, the regulatory authorities and the industry are interested in more precise calculations for economic reasons and Uncertainty Quantification (UQ). Therefore, instead of assuming the local deposition of heat, this work focuses on more detailed methods targeting the specific reactor regions hosting the experiments.

The main contributions to the nuclear heating in an operating reactor can be summarized as follows [21]:

- Prompt neutron heating: deposition of the recoil energy of nuclei after a neutron collision and energy deposition of resulting charged particles from neutron interactions - e.g., heating caused by protons released in (n, p) interactions.

- Delayed neutron heating: energy deposition of resulting charged particles created with a delay after a neutron interaction - e.g., heating caused by alpha particles resulting from actinide activation and heating caused by beta particles emitted from fission product decay.
- Prompt gamma heating: energy deposition of resulting charged particles of photo-atomic and photo-nuclear interactions. Prompt photons are emitted immediately after a neutron interaction. This includes prompt gamma radiation from the rapid de-excitation of the fission fragments and capture gammas.
- Delayed gamma heating: same energy deposition mechanism as prompt photons. Delayed photons are the result of activation and fission product decays.

After reactor shutdown, the prompt emissions no longer contribute to the heating, and we are left with only the delayed neutron and delayed gamma contributions. These contributions can be further classified by the location in which the emitted particles deposit their energy. The delayed neutron heating is caused by charged particles that deposit their energy locally due to their short range. The delayed gamma heating is caused by photons that deposit their energy globally and must be transported [22]. The following formula calculates the nuclear heating after reactor shutdown

$$H_T = H_{ch} + H_{\gamma,Tr} \quad (2.1)$$

where H_T is the total heat deposited in the region of interest, H_{ch} is the energy deposited by charged particles in the region of interest, $H_{\gamma,Tr}$ is the energy deposited in the region of interest resulting from photon transport.

A common method for the calculation of the delayed gamma contribution is the so-called Rigorous 2-Step (R2S) process [23]. Because the R2S process consists of three steps, this work refers to it as the formal 3-step process. This method consists of the following three steps:

- First, a neutron transport calculation determines the neutron flux spatial and energy distribution during reactor operation.
- Second, an activation calculation estimates the energy distribution and emission probability of the photon decays after reactor shutdown.
- Third, a photon-transport calculation evaluates the delayed gamma heating.

Because the R2S process consists of three steps, this work refers to it as the *formal 3-step process*. Earlier versions of this method accounted for the delayed photons through a fixed-source calculation based on pre-defined gamma emission probabilities. For example, Ambrosek et al. [24] used this method to calculate the delayed gamma heating component in two ATR experiment positions. Lee et al. [25] also used this method to calculate the gamma-radiation dose of multiple Pressurized Water Reactor (PWR) spent fuel elements.

For the formal 3-step process, Mozin [26] utilized MCNP [27] and CINDER [28] to investigate the delayed gamma spectra sensitivity to the actinide content of spent Light Water Reactor (LWR) fuel. Lemaire et al. [21] utilized TRIPOLI-4 [29] and the point-depletion tool PEPIN-2 [30] to study the delayed gamma contribution from fission product decay in the Jules Horowitz Reactor. Noh et al. [31] coupled MCNP and ORIGEN to study the delayed gamma heating by radiation originated in the activated structures of the High flux Advanced Neutron Application ReactOr (HANARO), a 30-MW research reactor of the Korea Atomic

Energy Research Institute (KAERI). Peterson-Droogh and Howard [22] summarized in a paper the modeling techniques for the calculation of the heating rates and the activity of iridium-192 targets in the HFIR at the Oak Ridge National Laboratory (ORNL). Their delayed heating calculation was separated into two methods. The first method was the PIKMT method, described later in this chapter, for obtaining the delayed gamma contribution from fission product decay. The second method calculated the delayed gamma contribution from activation product decay using a coupling of MCNP and ORIGEN.

The dose rate evaluation of fusion reactor components is another common application of the formal 3-step process. Chen et al. [23] calculated the shutdown dose rate of the International Thermonuclear Experimental Reactor (ITER) mid-plane maintenance port coupling MCNP and the FISPACT inventory code [32]. Other studies using this method include [33], [34].

Valenza et al. [35] introduced another method widely used for shutdown dose rate calculations in fusion reactors. The method replaces the prompt gamma spectrum with a decay gamma spectrum, and it is better known as the Direct 1-step (D1S) method. The method requires modifying the nuclear data library so that the delayed gamma radiation replaces the prompt gamma radiation. For example, the delayed gamma rays from Co-60 replace the prompt gamma rays that accompany the Co-59 (n, γ) reaction. Additionally, the method relies on a depletion solver to account for the build-up and the decay of the considered radionuclides. Other fusion reactor studies [36], [37] used this same method. For fission reactors, Lee et al. [38] used this method to calculate the delayed gamma contribution in the HANARO reactor. Although the D1S method is simple, it is most reliable if the generated radionuclides have short half-lives [31].

The PIKMT method has become predominant in the past few years for delayed gamma heating calculations. It was first introduced by Brown et al. [39] and relies on the assumption that the prompt and delayed gamma spectra are similar. The method uses one neutron-photon coupled simulation using the MCNP PIKMT card to evaluate the energy deposited by prompt gamma radiation originated at fuel fission events, and then it scales it to the delayed gamma component.

$$H_{\gamma,d} = Q_{\gamma,d}/Q_{\gamma,p} \times F6:p_{PIKMT} \quad (2.2)$$

where $H_{\gamma,d}$ is the delayed gamma heating from the fission product decay, $Q_{\gamma,p}$ and $Q_{\gamma,d}$ is the energy released from fission carried away by the prompt and delayed gamma components, and $F6:p_{PIKMT}$ is the MCNP $F6$ tally with the PIKMT card.

Ilas et al. [40] used the PIKMT method for calculating the heating rates in the HFIR's reflector during a highly-enriched uranium (HEU) to low-enriched uranium (LEU) conversion study. Davidson et al. [41] calculated the delayed gamma heating in different reactor regions using the PIKMT method for another HEU-to-LEU conversion study in HFIR. Jurbandam [42] calculated the spatial heat distribution in the SAFARI-1 nuclear reactor, relying on the PIKMT method to account for the delayed gamma contributions.

One of the main advantages of the PIKMT method over the formal 3-step process is that it requires only one coupled neutron/photon-transport simulation for obtaining the delayed gamma contribution. However, it only accounts for the gamma radiation emitted from fission product decay. In general, when the component of interest is located close to the fuel, the photons from fission product decay become the major contributor to the delayed heating [21]. In other cases, the photons from activation product decay may have higher contributions. For those cases, only the formal 3-step process is applicable. Additionally, the formal 3-step process obtains the charged particle contribution, which is a contribution neglected by the PIKMT method. Furthermore, the PIKMT method obtains the delayed gamma contribution during reactor operation or immediately after shutdown. Given that this work is concerned with the time evolution of the delayed heating,

the only applicable method here is the formal 3-step process.

2.2 Transport/Depletion Solvers Background

The first two steps in the formal 3-step process are accomplished by coupling a transport and a depletion solver. This strategy determines the evolution of the material composition in the reactor during operation. Depletion solvers ignore the geometry of the system, which underscores the need for a transport solver. The transport solver generates geometry- and material-dependent parameters that the depletion solver relies on. The depletion solver then calculates the material compositions and returns this information to the transport solver, and a cyclical transfer of information occurs, as shown in Figure 2.1.

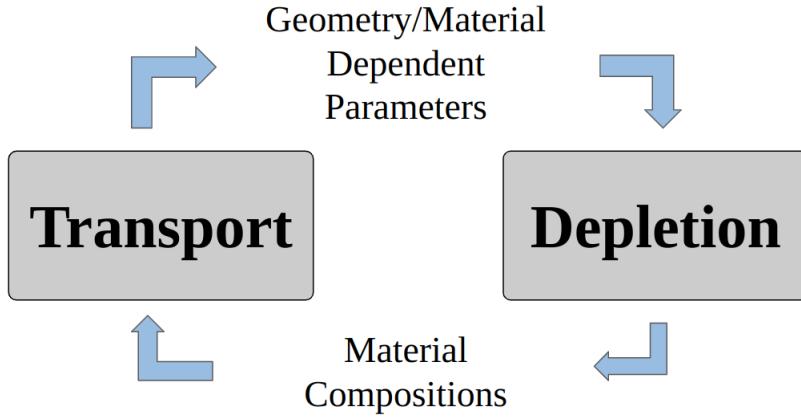


Figure 2.1: Typical workflow of a transport/depletion solver.

There are several software packages available with these capabilities. Table 2.1 summarizes some popular transport/depletion software packages that have been developed over the past 25 years. This list is not meant to be extensive and instead provides the reader with an overview of past developments. The motivation for the development of these tools varies. Some were created to address specific projects, such as MOCUP [43], which was developed to simulate isotope depletion of the ATR and the Advanced Neutron Source, and Monteburns [44], which was created to simulate isotope depletion for the Accelerator Transmutation of Waste (ATW) project at the Los Alamos National Laboratory (LANL).

Additionally, MCNP's lack of depletion capabilities motivated the development of many of these software packages, as well as the addition of depletion subroutines to MCNP itself [28]. The integration of CINDER90 into MCNP provides a self-contained Monte Carlo/depletion capability in a single tool. Serpent [45] is another example of a Monte Carlo solver with self-contained depletion subroutines.

Most of these tools, however, still rely on ORIGEN2, which was deprecated and lost support more than three decades ago. The current version is ORIGEN/SCALE (sometimes called ORIGEN-S or just ORIGEN), which uses the Chebyshev Rational Approximation Method (CRAM) [48] to solve the system of depletion equations. While the open literature describes various depletion algorithms, the three most relevant are CINDER90 [28], MATREX [17], and CRAM. MCNP uses CINDER90, ORIGEN2 uses MATREX, and ORIGEN uses either MATREX or CRAM. The accuracy of the CINDER90 algorithm deteriorates for actinide prediction in higher-burnup cases [28]. Although increasing CINDER90's accuracy is possible by using a sub-step method, the CRAM solver produces more accurate solutions even when using longer burnup intervals

Table 2.1: List of previous Transport/Depletion software packages.

Tool	Reference	Transport	Depletion	Year developed
MOCUP	[43]	MCNP	ORIGEN2	1995
Monteburns	[44]	MCNP	ORIGEN2	1998
MCODE-2.2	[46]	MCNP	ORIGEN2	2008
VESTA	[47]	MCNP	ORIGEN2/Phoenix	2008
MCNP	[27]		MCNP (CINDER90)	2009
Serpent	[45]		Serpent (CRAM)	2010

[49]. The MATREX algorithm is a hybrid method combining the Matrix Exponential method and linear chains [50]. While the matrix exponential and linear chains can be solved accurately when independent of each other, accounting for the effects of short and long-lived nuclides in the hybrid method requires additional approximations that lead to a loss of accuracy [51]. Overall, CRAM is orders of magnitude more accurate than other depletion algorithms currently available [49]–[51]. Furthermore, in most cases, the solver is up to several times faster due to not having sub-stepping requirements [51].

As mentioned in Chapter 1, this work intends to demonstrate the modeling capabilities of ATR. The physical configuration of ATR, shown in Figure 2.2, gives the reactor the unique capability to operate simultaneously at different power levels in the corner lobes. This feature allows for independent testing conditions within the same operating cycle for experiments located in different reactor regions [52].

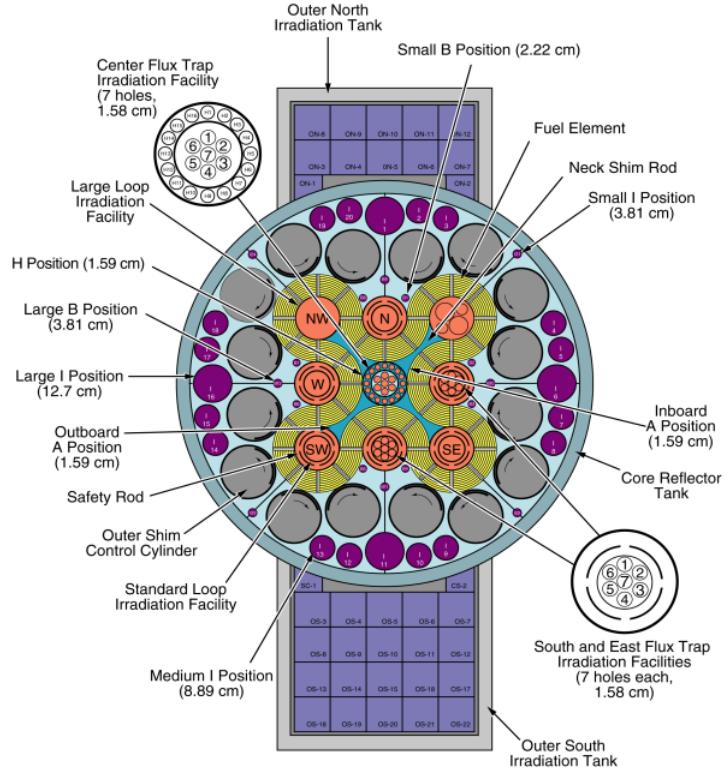


Figure 2.2: ATR cross sectional diagram and irradiation locations. Image reproduced from [52].

The analysis of ATR experiments requires the transport/depletion solver to have a lobe power scaling

capability to properly calculate the neutron flux normalization factor and hence obtain the right depletion rates. While multiple tools have targeted ATR in the past, most of them are no longer maintained. Active maintenance has many desirable features, including the development of new capabilities, bug catching and fixing, code updates for compliance with quality assurance requirements, among others. Finally, continuous energy Monte Carlo depletion calculations have high computational requirements with long computation times.

Overall, a tool relying on ORIGEN (and CRAM) that can model ATR and has low computational requirements can fulfill this thesis' needs. The tool with all the desired capabilities is the MOAA tool [53], which is described in detail in Chapter 3.

2.3 Machine Learning

As stated in Chapter 1, one of the objectives of this Ph.D. thesis is to accelerate the prediction of the bounding event outcome (capsule melts or does not melt). This method relies on multiple machine-learning algorithms. This section of the thesis motivates the choice of the algorithms and describes them briefly.

Machine learning is the branch of artificial intelligence that allows a computer algorithm to derive relationships from data. This feature allows them to solve complex problems, while at the same time decreasing the calculations' computational expense. For example, the high computational expense of physics-based calculations makes them unsuitable for various applications, e.g., real-time reactor control. Even though the high-performance computing (HPC) capabilities have increased considerably in the past few decades, they are still not sufficient. In contrast, machine learning models can emulate the behavior of the input/output relationship of the physics-based model in real-time.

Traditional machine learning techniques have been applied in the nuclear engineering field to some extent. Na et al. [54] applied Support Vector Machine (SVM) and Support Vector Regression (SVR) to NPP safety analysis. During an accident, the NPP operators may have insufficient time to analyze the data and swiftly mitigate it. Their work studied the applicability of SVM and SVR to a LOCA by predicting the break size and its location. The model used multiple reactor signals, including core exit temperature, pressurizer water level, and containment pressure.

NPPs safety systems calculate a number of safety-critical parameters and trip the reactor when they exceed the operating limits. An accurate estimation of the local power density is necessary for the safe operation of an NPP for preventing the fuel rod from melting. Bae et al. [55] applied SVR to NPP online monitoring. Their study accurately estimated the local power density based on different signals of the reactor coolant system, including reactor power, core inlet temperature, and coolant flow rate.

Zeng et al. [56] created a machine learning-based system for autonomous control of small reactors. The system consisted of a prediction module using SVR to estimate the state of some system variables, such as coolant and fuel temperatures, flux, and power history. The prediction module fed into a decision module, which consisted of the control decision making and control system motion. They demonstrated the system prediction module for a reactivity insertion event in a 20 MW_{th} Fluoride-salt-cooled High-temperature Reactor (TFHR).

Gohel et al. [57] developed a framework utilizing SVM and Logistic Regression (LogR) for conducting predictive maintenance of NPPs. Predictive maintenance allows mitigating component failures and to increase the power plant availability. Their framework intended to predict the failure of a power plant infrastructure in future cycles based on sensor real-time data. Due to the lack of real NPP data, the paper demonstrated

the framework by applying it to the degradation of a turbofan engine.

Blevins and Yang [58] discussed the suitability of machine learning for manufacturing in nuclear engineering. Their article presented an overview and examples of applications of multiple machine learning techniques, such as Decision Tree Regression (DTR), Random Forest Regression (RFR), SVM, K-Nearest Neighbors (KNN), and Feedforward Neural Networks (FNNs), to advanced manufacturing. As many of these algorithms have not been yet applied to advanced manufacturing in the nuclear engineering field, they extrapolated these methods from different engineering applications, including semiconductor doping, manufacturing defect detection, and additive manufacturing parameters' optimization, among others.

Additionally, deep-learning, which is a sub-field of machine learning, has also been applied to nuclear engineering problems. Deep-learning algorithms are more complex and abstract than traditional machine learning algorithms. Nevertheless, a higher level of abstraction allows them to automate some pieces of the training process and, hence, eliminate some of the human intervention. In 1992, Guo and Uhrig [59] presented one of the first applications of Artificial Neural Networks (ANNs) to nuclear engineering. They developed a model for online monitoring that was able to predict the thermal performance and gross heat rate of the Sequoyah NPP. The model utilized a hybrid neural network, comprising a clustering stage and an FNN, trained with sensor signals from the plant. Additionally, they applied a sensitivity analysis to the network to determine the variables that affected the heat rate the most. This kind of study allows power plant personnel to determine the cause of anomalies and, hence, provide a controlling basis for them.

Erdogan and Melih [60] developed an optimal loading algorithm for nuclear fuel based on an FNN and a genetic algorithm. Their algorithm searched for a loading pattern that minimized the initial amount of fuel and satisfied the reactor safety requirements. Overall, minimizing the use of fuel reduces the operational cost of the plant.

Andersson et al. [61] developed an application for online monitoring utilizing FNN. The network calculated the control rod positions from the axial shape of the neutron flux. Such an application helps to corroborate the control rod positions during transients. The neural network could guide that control rod calibration process as well, as the electromechanical position indicators can be recalibrated during operation.

Montes et al. [62] applied FNNs to reactor fuel optimization. The network quickly determined the local power peaking factor in a Boiling Water Reactor (BWR) based on the distribution of fissile and burnable poison materials. The power peaking factor is a design parameter, and the ability to quickly estimate it reduces the fuel optimization time. Gomez Fernandez et al. [63] applied FNNs to reactor accident identification. Their data comprised multiple sensor signals from a small LWR and the time evolution corresponding to two transient events, a piecewise-constant power ramp and a loss-of-feedwater (LOFW) accident. The model was able to accurately identify the transients and predict the time evolution of the reactor. A model with such features can guide the NPP operators to identify safety issues and to take prompt actions to mitigate the accident.

Other examples of ANNs applied to nuclear engineering include the use of Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs). For example, Kamuda et al. [64] applied CNNs to automate isotope identification in gamma-ray spectroscopy. Wilson [65] compared the performance of SVR and a CNN at predicting the reactor power shape for diverse shim blade positions. Their goal was to demonstrate that machine learning can provide real-time predictions and contribute to the autonomous control of nuclear reactors.

RNNs were created to handle inputs and outputs correlated as part of a sequence. For example, when predicting the next word of a sentence, the prior words are necessary. While all inputs and outputs in an

FNN are independent of each other, an RNN is a sequential network that allows information to persist and uses it for processing the current input. In the last few years, RNNs have been successfully applied to various tasks involving sequential inputs, such as speech and language recognition/prediction [66].

In practice, RNNs cannot learn long-term dependencies. When training the network, the back-propagated gradients may grow or shrink significantly at each time step. Therefore, the gradients may vanish or explode for sufficiently long time series. This issue is called the vanishing gradient problem, and Long Short-Term Memory Networks (LSTMs) were explicitly designed to avoid it. LSTMs are a special kind of RNN and are able to learn long-term dependencies.

Yang and Kim [67] developed a model using LSTMs for accident diagnosis. The model predicted the evolution of the plant's main variables and identified the accident. Lee et al. [68] created a model using LSTMs for the autonomous operation of safety systems. Radaideh et al. [69] predicted the progression of a LOCA utilizing deep-learning algorithms, including FNNs and LSTMs.

Overall, machine learning has been applied to different areas of nuclear engineering. The numerous applications described here cover many algorithms that can be classified as traditional machine learning or deep-learning. This literature survey helps to understand the state of development of this field and can guide the choice of algorithms for this work. However, machine learning has not been yet utilized in applications similar to this thesis'. Therefore, Chapter 6 compares various algorithms performance and analyzes their suitability to this thesis.

Chapter 3

Delayed Heating

3.1 MOAA

The Irradiation Experiment Neutronics Analysis group at the INL supports the ATR and the Transient Reactor Test Facility (TREAT) by performing neutronics analyses before the irradiation of experiments takes place. The analyses provide the radioisotope source term of the experiments after irradiation. The calculated source terms have multiple applications, which include radiation shielding, inventory and shipping purposes, demonstration of compliance with the safety limits, modeling of radioactive material dispersion, comparison to PIE results for validation, and experiment design and optimization, among others.

This section introduces the MCNP-ORIGEN Activation Automation (MOAA) tool and explains its workflow. The Irradiation Experiment Neutronics Analysis group developed MOAA to streamline the calculation of the experiment source terms. MOAA is a Python package that couples MCNP [27] and ORIGEN [17] by writing MCNP tally cards, reading MCNP tallies, creating SCALE input files, executing SCALE, and standardizing the results post-processing. Streamlining this procedure helps to reduce processing times and avoid potential human errors.

Although MOAA was originally developed to analyze the irradiation of experiments at ATR, it has grown to become a more general tool through further development. Overall, MOAA is a tool that calculates radionuclide mass, activity, and heat in the regions of interest in a nuclear reactor. Similar tools set a path for potential future applications for MOAA, including fuel burnup [3], spent fuel repository design [70], transportation cask design [71], shutdown dose calculation [23], to mention just a few. Additionally, earlier work utilized MOAA for delayed heating calculations [72]–[74].

3.1.1 Calculation Workflow

MOAA performs material irradiation and decay calculations, which can be configured to include a single or multiple irradiation steps [53]. While a single-step calculation uses constant power irradiation, multi-step calculations can handle several core configurations and piecewise constant power irradiation (i.e., different power levels at each time step). Such capability allows for the modeling of evolution in material density, temperature, and geometry, as well as the control rod movement during the irradiation cycle. Additionally, the calculations can be configured to utilize the conventional predictor-corrector scheme, which is a two-step process that combines explicit and implicit calculations to achieve higher accuracy.

Figure 3.1 highlights MOAA’s main calculation workflow that integrates with MCNP and ORIGEN.

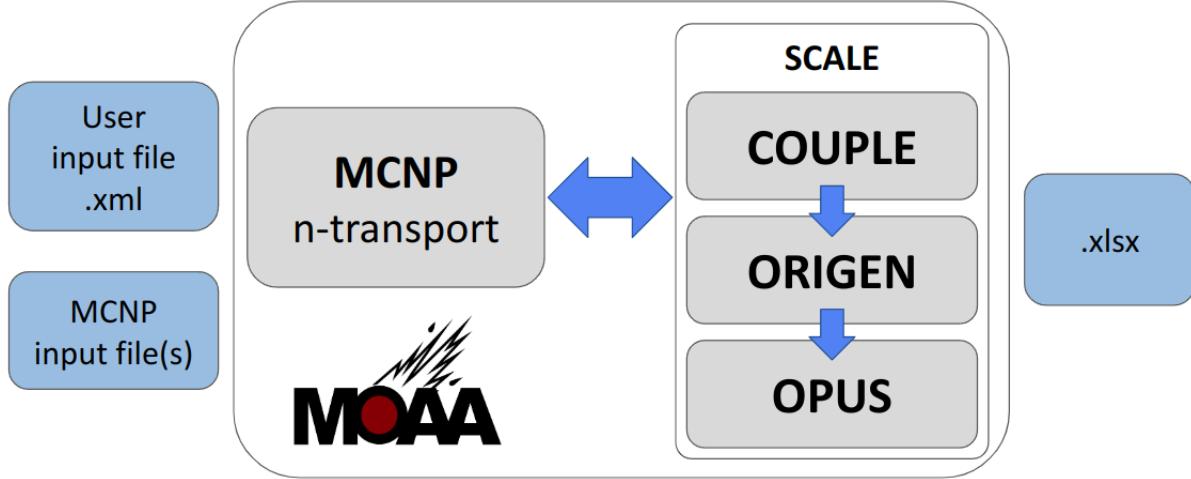


Figure 3.1: Graphical representation of MOAA’s workflow.

MCNP is a general-purpose, continuous-energy, generalized-geometry, Monte Carlo radiation-transport tool that can track neutrons, photons, electrons, and other particles. The main advantage of the Monte Carlo method is its capability to model geometry and interaction physics without significant approximations. MOAA relies on MCNP for obtaining the geometry- and material-dependent parameters of the user-defined system during the irradiation steps - i.e., fluxes and reaction rates.

ORIGEN is a general-purpose point depletion and decay tool to calculate isotopic concentrations, radiation source terms, and decay heat. ORIGEN is integrated into the SCALE code system, which is a modeling and simulation suite for nuclear safety analysis and design. MOAA uses the calculated parameters from the MCNP output files to define the SCALE input files. Besides ORIGEN, MOAA also uses the COUPLE and OPUS modules from SCALE. ORIGEN requires a single volume- and spectrum-weighted cross-section library that MOAA generates using COUPLE. Meanwhile, OPUS provides the ability to extract specific data from the ORIGEN output libraries, perform unit conversions, and generate data for post-calculation analysis.

MOAA calculates the following parameters for each cell of interest

$$\phi(E)^{l,m} = F4_E \quad (3.1)$$

$$\bar{\phi}_T^{l,m} = \frac{\nu \times P[MW]}{Q[MeV/fiss] \times k_{eff} \times 1.6022 \times 10^{-19}} \times \sum_E F4_E \quad (3.2)$$

$$\sigma_{j,k}^{l,m} = \frac{F4_{FM_{j,k}}}{\sum_E F4_E} \quad (3.3)$$

$$k = (n, \gamma), (n, p), (n, \alpha), (n, fission) \quad (3.4)$$

where $\phi(E)$ is the spectrum normalized to unity, $\bar{\phi}_T$ is the power-normalized total neutron flux, and σ is the microscopic one-group cross-section. The index l corresponds to the irradiation step and index m specifies the calculation to be the predictor or corrector case. The index j corresponds to the isotope identification number and index k corresponds to the reaction type. $F4_E$ is the F4 tally split up into energy bins, $F4_{FM_{j,k}}$ is the F4 tally for isotope j reaction type k , ν is the number of neutrons emitted per fission, P is the reactor

power, Q is the released energy per fission, and k_{eff} is the effective multiplication factor. MOAA extracts the parameters ν , k_{eff} , $F4_E$, and $F4_{FM_{j,k}}$ from the MCNP output files, and P and Q are user-defined parameters. A special use case is the ATR simulations in which P has to be calculated. Section A.2 gives further details on its calculation.

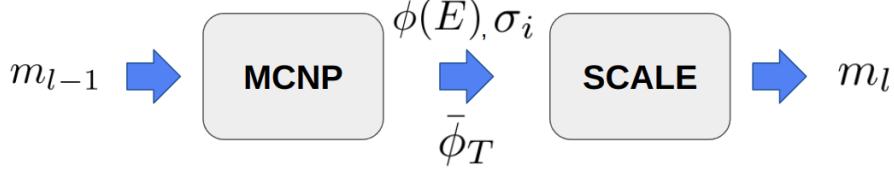


Figure 3.2: Information flow of a MOAA simulation.

For multiple irradiation steps, MOAA calculates the isotopic composition at the end of the step m_l based on the isotopic compositions from the previous step m_{l-1} , as shown in Figure 3.2. For the first irradiation step, the isotopic composition is extracted from the MCNP input file. This method is known as constant extrapolation (CE) or Euler's method, which assumes that the flux and cross-sections remain at their beginning-of-step values throughout the interval [75]. If the predictor/corrector scheme is enabled, the irradiation step requires two irradiation calculations. The first irradiation calculation uses m_{l-1} to obtain $\phi(E)^{l,p}$, $\bar{\phi}_T^{l,p}$, and $\sigma_{j,k}^p$ to define the SCALE input files and calculate an intermediate isotopic composition m_l^p . The second irradiation calculation uses m_l^p to obtain $\phi(E)^{l,c}$, $\bar{\phi}_T^{l,c}$, and $\sigma_{j,k}^c$ and the following averaged parameters to define the SCALE input files and calculate m_l

$$\phi(E) = \frac{1}{2}\phi(E)^{l,p} + \frac{1}{2}\phi(E)^{l,c} \quad (3.5)$$

$$\bar{\phi}_T = \frac{1}{2}\bar{\phi}_T^{l,p} + \frac{1}{2}\bar{\phi}_T^{l,c} \quad (3.6)$$

$$\sigma_{j,k} = \frac{1}{2}\sigma_{j,k}^{l,p} + \frac{1}{2}\sigma_{j,k}^{l,c}. \quad (3.7)$$

This method is known as conventional predictor-corrector or constant extrapolation with linear interpolation (CE/LI) [75]. As the name suggests, the predictor calculation assumes constant values throughout the interval, and the final calculation uses an average of the predictor and corrector flux and cross-sections, which corresponds to a linear interpolation between the beginning and end-of-step values.

The creation of a MOAA application starts with the definition of a user input file (UIF) in *.xml* format, as shown in Figure 3.1. The UIF contains a set of required and optional settings for MOAA. The required settings are a list of irradiation cases, a list of decay times, and a list of cells. Within the irradiation cases, the user must specify an associated MCNP input file, an irradiation time, and an irradiation power. Appendix A displays a minimal MOAA input file. MOAA's workflow can be summarized into Algorithm 1. If the predictor/corrector scheme is enabled, the number of cases is twice the number of irradiation cases.

3.1.2 ATR cases

The definition of an ATR case uses the same flux normalization as Equation 3.2, but requires the specification of the different lobe powers, as well as the calculation of adjusted lobe powers. The adjusted lobe power can be thought of as a “virtual” total reactor power that would produce the same flux level in that lobe. For

Algorithm 1 MOAA's main algorithm.

```
1: parses user input file
2: for case do
3:   if case > 0 then
4:     updates MCNP inputs with case-1 calculated compositions
5:   appends MCNP tallies
6:   executes MCNP
7:   parses MCNP output
8:   if case is corrector then
9:     applies correction
10:  writes COUPLE sequence
11:  writes ORIGEN sequence
12:  writes OPUS sequence
13:  executes SCALE
14:  processes results
```

experiments in lobes northwest (NW), northeast (NE), center (C), southwest (SW), and southeast (SE), the flux normalization uses the adjusted lobe power of the user-specified lobe powers. The flux normalization of experiments in the rest of the lobes uses a weighted power average of the closest lobes. For example, the east lobe (E) power is calculated as an average between the power of lobes NE, C, and SE. The following formula calculates the adjusted lobe powers

$$T_i = \sum_c F7_c \times m_c \quad (3.8)$$

$$f_i = \frac{T_i}{\sum_j T_j} \times \sum_j LP_j \quad (3.9)$$

$$P_i = \frac{LP_i}{f_i} \times \sum_j LP_j \quad (3.10)$$

where $F7_c$ is the fission energy tally, m_c is the mass, and index c represents the cells integrating the lobe. LP_i is the user-defined power for lobe i and P_i is the adjusted lobe power. The required settings within the irradiation cases include an MCNP input file, an irradiation time, an irradiation power divided into five lobe powers: NW, NE, C, SW, SE. Additionally, the user must provide the F7 tally identification number that will be used for adjusting the lobe powers. Appendix A displays a typical UIF for an ATR irradiation experiment.

3.1.3 Quality Assurance and Version Control

Nuclear regulators require the implementation of a quality assurance program for all systems that prevent or mitigate the consequences of postulated accidents in nuclear power plants and reprocessing plants. Software plays a major role in licensing by producing calculations critical to the design and safety [76]. The implementation of a software quality assurance program ensures code reliability.

This section briefly describes the current efforts for MOAA to fulfill the American Society of Mechanical Engineers (ASME) Nuclear Quality Assurance 1 (NQA-1) Standard [77]. According to the NQA-1 documentation, “this Standard reflects the current understanding of the quality assurance requirements necessary to

achieve safe, reliable, and efficient utilization of nuclear energy, and management and processing of radioactive materials.” These requirements target different steps of the software development process, including *software design requirements*, *software design*, *software implementation*, and *software design verification*. These steps entail the documentation of the entire process – from the establishment of the software requirements to the confirmation that the design correctly fulfills them. The collection of these assets holistically qualifies MOAA for performing nuclear reactor physics analysis. While this documentation discusses actions pertaining to the development of MOAA, MCNP and SCALE respective requirements are beyond the scope of this work.

The NQA-1 standard requires reviews and approvals for each stable release, and a software test plan details the verification process that leads to such a release. The verification process comprises design reviews, alternative calculations, and/or testing. Testing is the process of comparing the behavior of the observed code against its expected behavior. The type of output for the different tests may range from numerical verification to error handling testing, covering behavior from the common to the extreme. MOAA uses “dynamic” testing, including white and black box techniques, to attempt to identify defects by executing the software. All tests have the same objective, which is to ensure that the system produces the expected output when executed with the prescribed input. This mechanism helps to provide confidence that the software performs as intended [78].

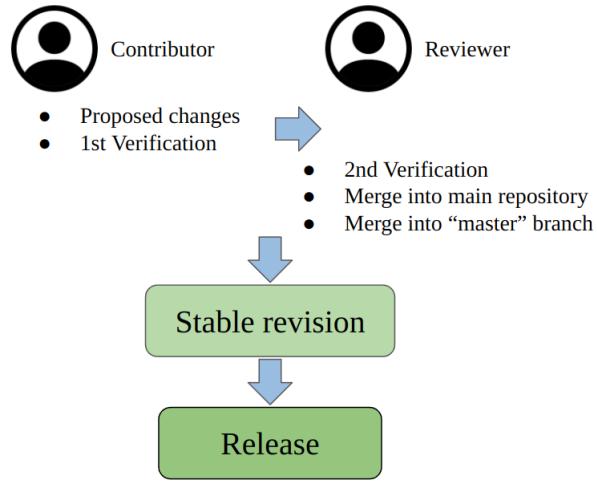


Figure 3.3: Quality assurance process for MOAA.

MOAA leverages the AGILE methodology for software development and tests that all proposed changes are properly integrated before merging them into the main repository hosting the source code, as shown in Figure 3.3. The proposed changes may include new features, which require the addition of new software design requirements, or a bug fix, which means that a software design requirement is not currently met. After testing has been successfully completed (independently) by the contributor and a reviewer, a merge is made into the “master” branch of the repository, creating a new “stable” revision. Each revision is eligible for release and subject to a complete release review. Currently, a stable release of MOAA is under review.

3.1.4 Simulation Specification

To accommodate the MOAA calculations, the MCNP input files must undergo a few minimal modifications. First, the depleted cells sharing the same material definition require the duplication of the original material, as

each MOAA depleted cell requires an independent material definition. The reason for this is to accommodate the depletion of independent cells subject to individual flux levels, divergent depletion rates, and the resulting different material compositions.

Second, while MOAA can handle the depletion of cells defined as repeated structures, it cannot handle the depletion when the same cell is present in different nested structures. For example, if cell 100 is nested in cells 300 and 900 ($100 < 300 < 900$), and cell 100 is also nested in cells 400 and 900 ($100 < 400 < 900$), MOAA can handle only one of the two nested structures in one given simulation. To handle both nested structures, cell 400 would have to be redefined as 300 if the geometry definition allows, and MOAA will be able to handle both occurrences simultaneously.

Third, the volume of some depleted cells has to be calculated and specified within the MCNP input file. This is because MCNP can only calculate the volume of simple geometries only, and the user must specify the cell volumes to avoid errors. For cells defined as repeated structures, the user must specify the volume of the whole material in those cells that is present in the geometry. Otherwise, MCNP will calculate the volume of only one cell.

3.2 Delayed heating

In an operating reactor, an equilibrium exists between the generation and removal of heat. After reactor shutdown, radioactive decay continues to release energy, which is deposited across the reactor and causes the delayed heating. The estimation of delayed heating in experiments and in the reactor structures is paramount in the safety analyses of research reactors, and its accurate determination generates a reliable heat source term for thermal-hydraulics calculations. This in turn allows the determination of heat removal requirements and evaluation of temperature levels in the components of interest [72], [79], [80]. Additionally, detailed calculations help in the optimization process of the irradiation target designs [22].

Overall, these calculations determine whether delayed heating presents a hazard for the components of interest, which could lead to the release of radioactive contaminants into the reactor coolant. An important application example is in spent nuclear fuel where delayed heating calculations are crucial for the design of nuclear facilities, including spent fuel storage pools, spent fuel transportation systems, reprocessing plants, and final storage sites.

This section introduces a delayed heating calculation workflow based on MOAA, presents the motivation for the choice of the method, and showcases its capabilities by presenting two specific applications.

3.2.1 Calculation Workflow

Figure 3.4 summarizes the calculation workflow, which relies on the formal 3-step method for the delayed heating calculation. During reactor operation, the first two steps in the formal 3-step process are accomplished by coupling a transport and a depletion solver. The delayed heating calculation workflow presented here relies on the MOAA tool [53] for carrying out the first two steps. After reactor shutdown, the depletion solver calculates the energy distribution and emission probability of the photon sources. The photon-transport calculation uses this information to evaluate the delayed gamma heating.

The delayed heating calculation can be separated into two parts. The first part calculates the deposition of energy from charged particles H_{ch} . This part includes only the charged particles that originated in the geometry of the component of interest and assumes that all the particles with mass deposit their energy in the region of interest. The following equations calculate H_{ch} [16], [22], [81]:

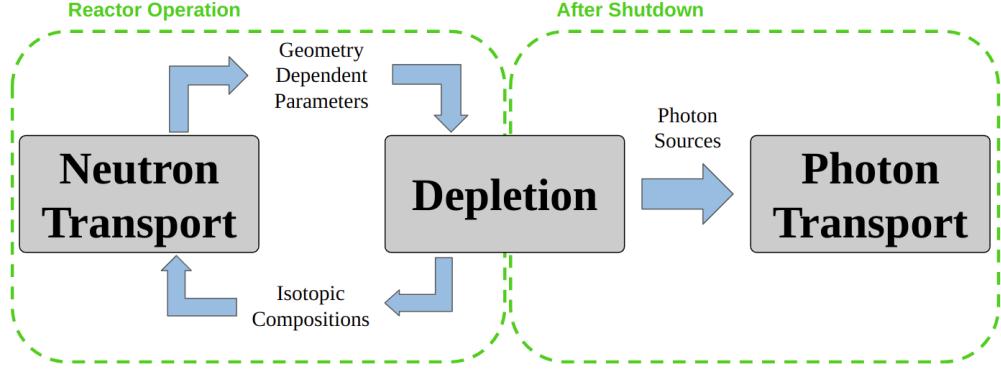


Figure 3.4: Calculation workflow diagram.

$$H_{T,L} = \sum_j N_j \lambda_j \bar{E}_j \quad (3.11)$$

$$H_{\gamma,L} = \sum_j N_j \lambda_j \bar{E}_{EM,j} \quad (3.12)$$

$$\bar{E} = \bar{E}_{EM} + \bar{E}_{HP} + \bar{E}_{LP} \quad (3.13)$$

$$\bar{E}_{EM} = \bar{E}_\gamma + \bar{E}_{X-ray} + \bar{E}_{annih.rad.} + \dots \quad (3.14)$$

$$\bar{E}_{HP} = \bar{E}_\alpha + \bar{E}_{SF} + \bar{E}_p + \dots \quad (3.15)$$

$$\bar{E}_{LP} = \bar{E}_{\beta^-} + \bar{E}_{\beta^+} + \bar{E}_{Ae^-} + \dots \quad (3.16)$$

$$H_{ch}[W] = H_{T,L} - H_{\gamma,L} \quad (3.17)$$

where $H_{T,L}$ is the total decay heat, N_j is the number of nuclei j , λ_j is the decay constant of nucleus j , \bar{E}_j is the total energy available (excluding neutrino energies) per decay of nucleus j , $H_{\gamma,L}$ is the total gamma heating, \bar{E}_{EM} is the electromagnetic component, \bar{E}_{HP} is the heavy particle component, \bar{E}_{LP} is the light particle component, \bar{E}_γ is the average gamma-ray energy, \bar{E}_{x-ray} is the average X-ray energy, $\bar{E}_{annih.rad.}$ is the average annihilation energy, \bar{E}_α is the average α energy, \bar{E}_{SF} is the average spontaneous fission energy, \bar{E}_p is the average proton energy, \bar{E}_{β^-} is the average β^- energy, \bar{E}_{β^+} is the average β^+ energy, and \bar{E}_{Ae^-} is the average Auger-electron energy.

The second part calculates the deposition of energy from photons $H_{\gamma,Tr}$. Because photons deposit their energy globally, photon transport is required, and this includes the photons originating from across the

reactor. The following equations describe the delayed heating

$$H_T[W] = H_{ch} + H_{\gamma,Tr} \quad (3.18)$$

$$H_{\gamma,Tr}[W] = 1.6022 \times 10^{-13} \times *F8[MeV] \times \sum_i S_i \quad (3.19)$$

$$S_i[\gamma/s] = \int_E \varphi_i^\gamma(E) dE \quad (3.20)$$

$$s_i[-] = S_i / \sum_j S_j \quad (3.21)$$

where H_T is the total heat deposited in the region of interest, H_{ch} is the energy deposited in the region of interest by charged particles, $H_{\gamma,Tr}$ is the energy deposited in the region of interest resulting from the photon transport, $*F8$ is the energy deposition calculated by an $*F8$ tally, S_i is the total photon emission rate from the region i , $\varphi_i^\gamma(E)$ the photon source energy distribution from region i , and s_i is the source cell emission probability.

Overall, the calculation scheme follows the formal 3-step process, in which MOAA conducts the first two steps, and multiple MCNP photon-transport simulations conduct the third step, calculating the delayed gamma heating. Figure 3.5 shows the calculation workflow. MOAA calculates $H_{T,L}$ and $H_{\gamma,L}$ in the component of interest assuming those to be locally deposited.

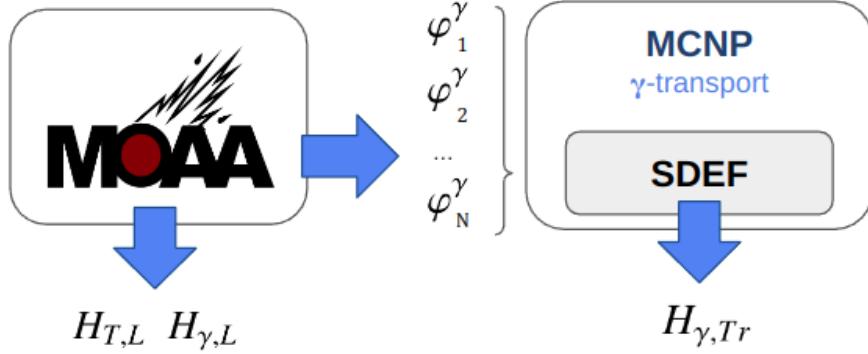


Figure 3.5: Delayed heating calculation scheme relying on MOAA.

MOAA calculates $\varphi_i^\gamma(E)$ for all the cells defined by the user as contributors to the delayed heating. The MCNP photon-transport simulations require the definition of a fixed source for these cells. The source energy distribution is specified with $\varphi_i^\gamma(E)$.

The photon-transport calculation can be performed with multiple simulations, in which one simulation obtains the heat contribution for one source cell, or with one simulation, in which the simulation obtains the heat contribution for all the source cells simultaneously. While the former determines the individual contributions from the different cells, the latter is computationally less expensive. Nonetheless, the latter requires the definition of the cell emission probabilities s_i using Equation 3.21.

The source spatial distribution is assumed uniform in each source cell. For uniformly sampling the birth of a particle in a cell, MCNP uses the enclosing volume rejection method, which requires the user definition of a volume enveloping the source cell. The randomly sampled points in the volume are accepted as source points only if they fall inside the source cell. Finally, the photon transport estimates $*F8$, which allows the

calculation of $H_{\gamma,Tr}$ (Eq. 3.19).

3.2.2 Simulation Specification

The calculation workflow requires the MOAA UIF and at least one MCNP input file defining the problem geometry. However, the user may want to use different geometries for the neutron and photon transport and both MCNP input files should be defined. This feature gives the calculation workflow great flexibility, allowing the user to model more complex scenarios. For example, a study case can be the delayed heating sensitivity to long burnup in a reactor where there is fuel assembly shuffling. Additionally, Chapter 4 studies cases where nuclear fuel is burned in the core, but the focus of the study is the shutdown dose rate for the PIE.

The delayed heating calculation requires two additional scripts: `run_case.py` and `source_gen.py`. The script `run_case.py` is the main workflow driver, and its role is summarized into Algorithm 2.

Algorithm 2 Delayed heating main algorithm.

- 1: initializes delayed heating object
 - 2: reads source information
 - 3: creates neutron transport input files
 - 4: runs MOAA
 - 5: assigns lattice numbers to cells in repeated structures
 - 6: processes photon sources
 - 7: divides sources into batches
 - 8: creates photon transport input files
 - 9: runs MCNP for photon transport
 - 10: processes results
-

The script `source_gen.py` allows for the automation of the photon source geometry definition. This script defines the dimensions of the source where the particle birth is randomly sampled. In the workflow described here, the user can define either a parallelepiped or a cylinder as the source geometry. Appendix B provides further details on the use of the delayed heating calculation workflow.

3.3 Results

This section presents and discusses the results of four exercises. The first exercise is a code-to-code comparison between the workflow presented here and the Serpent Monte Carlo tool [45]. The second exercise demonstrates the calculation workflow for a simple case geometry. The third exercise consists of the delayed heating calculation of an ATR experiment. Finally, the fourth exercise analyzes the delayed heating in various structures in the RA-6 reactor.

3.3.1 Delayed Gamma in a Uranium Sphere

This exercise studies the delayed gamma heating in a sphere of uranium. The purpose of this study is to understand the foundation of the method by comparing the results with Serpent. The Serpent Monte Carlo tool is a three-dimensional continuous-energy neutron transport application developed by the VTT Technical Research Centre of Finland, and it has been in public distribution since 2009. This work uses Serpent 2.1.32 and the cross-section library ENDF/B-VII.0 for the calculations. The choice of nuclear data was based

on their availability. This work uses Serpent because of its built-in burnup calculation routine. Moreover, Serpent is capable of calculating delayed gamma heating in a two-step process. The first step consists of a transport/depletion calculation that saves the radioactive material composition in a binary restart file. The second step is a photon-transport simulation that generates the photon source based on the binary restart file. This second simulation creates a `_gsrc.m` file specifying the gamma source distribution (see Appendix C for further details).

The problem geometry includes a sphere of uranium surrounded by water, and it is shown in Figure 3.6. The study focuses on the last component of equation 3.18, the heat generated by the transported gamma radiation $H_{\gamma,Tr}$. The gamma heating is calculated after a 50 day-burnup at a constant power of 5 MW.

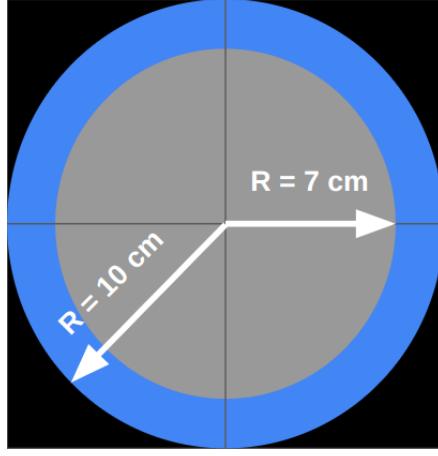


Figure 3.6: Geometry of a sphere of uranium surrounded by water.

The exercise compares the results of the nuclear heating calculation scheme developed here and Serpent. Serpent provides the reference values for calculating the relative difference employing the following equation.

$$\text{Rel. Difference [\%]} = \frac{R_{\text{Serpent}} - R_{\text{MOAA}}}{R_{\text{Serpent}}} \times 100 \quad (3.22)$$

where R is the result of interest, which can be the integral flux, the mass of a nuclide, or gamma source intensity.

Table 3.1 summarizes the main results for different stages of the calculation. The first stage is the burnup calculation. The relative difference for the total neutron flux before burnup is below 1%. After burnup, the relative difference for the masses of U-235 and U-238 is below 1% as well.

Table 3.1: Summary of the main results of the delayed heating calculation. ¹ Before burnup. ² After burnup.

	Units	MOAA/ MCNP	Serpent	Rel. Diff. [%]
¹ Tot. neutron flux	$10^{-3}/\text{cm}^{-2}/\text{source}$	3.9376	3.9284	0.24
² U-235 mass	kg	25.279	25.279	0.01
² U-238 mass	kg	2.835	2.835	0.01
² Photon source intensity	$10^{17}\gamma/\text{s}$	10.16	4.65	118.40
² Tot. gamma flux	$10^{14}/\text{cm}^{-2}/\text{s}$	2.6464	1.7805	48.63
$^2\text{H}_{\gamma,Tr}$	kW	73.969	48.037	53.98
$^2\text{H}_{\gamma,Tr}/\text{source}$	$10^{-14} \cdot \text{J/source}$	7.283	10.332	-29.51

Figure 3.7 displays the gamma source distribution from radioactive decay after burnup. MOAA uses 101 groups to obtain the source energy distribution and Serpent's data are formulated in the form of discrete lines. There are some clear discrepancies around 10 keV between Serpent and MOAA caused by the Bremsstrahlung radiation. While ORIGEN accounts for the photons originated by the deceleration of charged particles, Serpent considers only the photons originated directly from radioactive decays. In opposition, the upper range of the source distribution shows better agreement.

MOAA predicts a photon source intensity more than two times greater than Serpent, as shown in Table 3.1. The origin of this discrepancy is mainly attributed to the consideration of the Bremsstrahlung radiation into the calculations. Additionally, understanding the discrepancies requires some grasp of the algorithms in MOAA and Serpent. In MOAA, COUPLE provides ORIGEN with the cross-section library necessary for the depletion calculation. COUPLE creates by default spectrum-averaged cross-sections based on JEFF-3.0/A and replaces them with the cross-sections generated from the MCNP neutron transport calculation. The cross-section values that are not replaced still use the JEFF spectrum-averaged cross-sections. Monte Carlo solvers, such as MCNP and Serpent, track all necessary reaction rates and follow all the isotopes available in the nuclear data. While MOAA relies primarily on ENDF/B-VII.1, it still depends on JEFF-3.0/A. Consequently, MOAA and Serpent predict different after-burnup compositions, which translate into different photon source intensities.

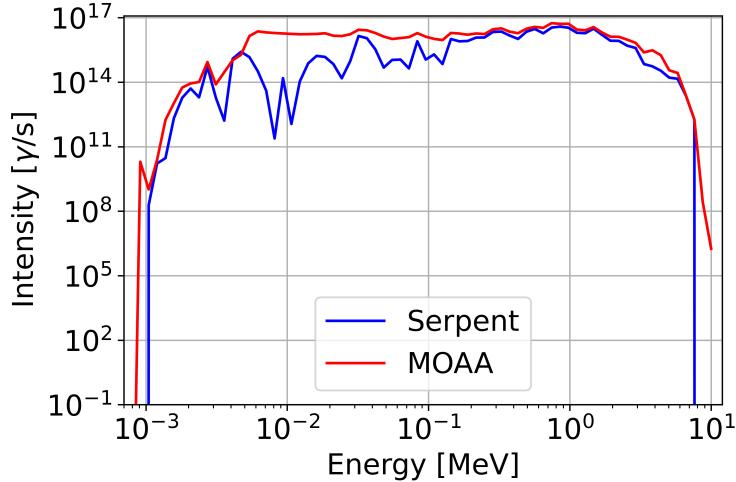


Figure 3.7: MOAA and Serpent photon source distributions.

Figure 3.8 displays the gamma spectra. The largest discrepancy in the spectra is close to 10 keV. MCNP fails to resolve some of the peaks in that area. Additionally, the differences in gamma source distributions are carried over. Table 3.1 shows the results for the total gamma flux and the gamma heating. Overall, the MCNP results are approximately 50% larger than Serpent's.

Calculating the gamma heating per source particle, we can isolate the effects of the source intensity and source energy distribution on the results. $H_{\gamma,Tr}/source$ in Table 3.1 shows the effects of the source energy distribution. The MCNP value is 30% smaller than Serpent's. This last result confirms that the discrepancies in the total gamma flux and gamma heating are caused by the differences in both the source energy distribution and the source intensity.

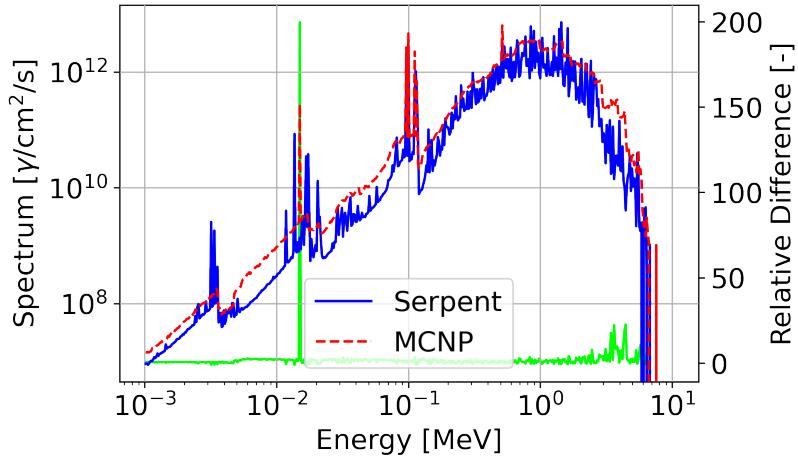


Figure 3.8: MCNP and Serpent photon spectra.

3.3.2 Demonstration Problem

A top view of the exercise geometry and its dimensions are shown in Figure 3.9. The experiment center is located 8.2 cm away from the reactor center. The fuel element and experiment heights are 14.8 and 24 cm, respectively. The reactor encompasses the following materials: 85% enriched uranium with a density of 19 g/cm³ for the fuel elements, light water with a density of 1 g/cm³ for the moderator, and iron with a density of 7.87 g/cm³ for the experiment.

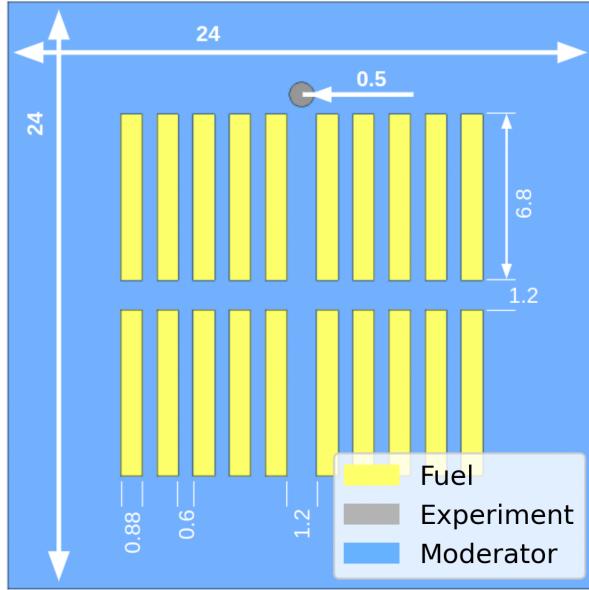


Figure 3.9: Geometry of the demonstration problem.

The experiment is irradiated for a period of 50 days with the reactor operating at a constant power of 5 MWth. After that period, the reactor shuts down, and the heat deposited in the experiment is calculated immediately after shutdown (to avoid the prompt-gamma contributions, this step is taken 1 sec after shutdown). The reactor regions considered to contribute to the experiment heating are the fuel elements and the same

experiment, while the activation of the moderator is neglected.

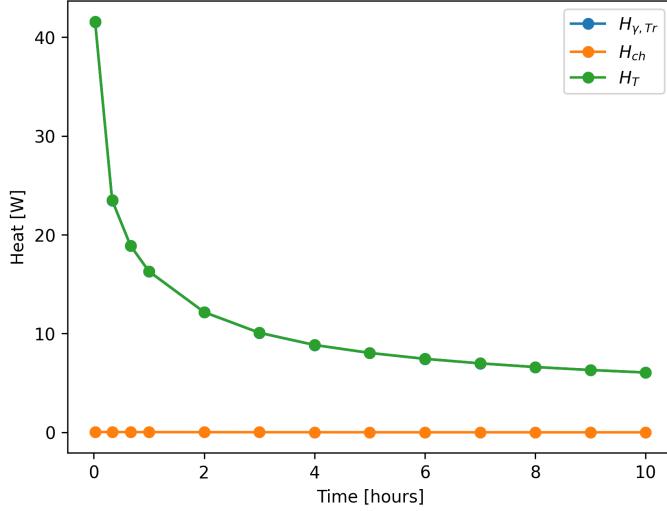


Figure 3.10: Heat production in experiment over time for the demonstration exercise.

Figure 3.10 displays the heat production over time in the experiment. The heat produced in the experiment immediately after shutdown is 41.58 W, which is less than 0.001% of the reactor total power. As expected, the heat deposition has a decaying shape. The curve has a half-life lower than 1 second, and after 4 seconds the heat production becomes smaller than 10 W.

Figure 3.11 shows the heat contribution by source. The fuel elements contribute to the experiment heating via the deposition of the gamma radiation that originates in them. The largest contribution to the heat deposition is originated in the closest fuel elements to the experiment. The experiment self-heating is a combination of the energy deposition of the charged particles and gamma radiation originating in the same experiment. The experiment material is not considerably activated during the irradiation period, and the self-heating contribution is deemed negligible.

3.3.3 ATR experiment

The calculation scheme presented in this work is reactor-agnostic, meaning that it can be tailored to any research reactor. Section 3.3.2 calculated the delayed heating in a simple configuration, which eases the visualization of the calculation scheme. This section demonstrates the calculation scheme for a full-scale research reactor, the ATR.

The ATR is a 250-MWth high flux test reactor located at the Reactor Technology Complex of the INL. The ATR was designed to provide an irradiation test environment for conducting a variety of experiments, to study the effects of radiation on reactor structural and fuel materials, and to produce medical and industrial isotopes [82], [83].

The ATR core contains 40 fuel elements arranged in a serpentine annulus between and around nine flux traps, as shown in Figure 3.12. Each fuel element consists of 19 parallel, curved, aluminum-clad fuel plates forming a 45-degree sector of a right circular cylinder. The fuel arrangement gives the reactor core a clover-leaf configuration, which allows the ATR to be operated at different power levels in the corner lobes, allowing for independent testing conditions within the same operating cycle. Within each fuel plate, the fuel

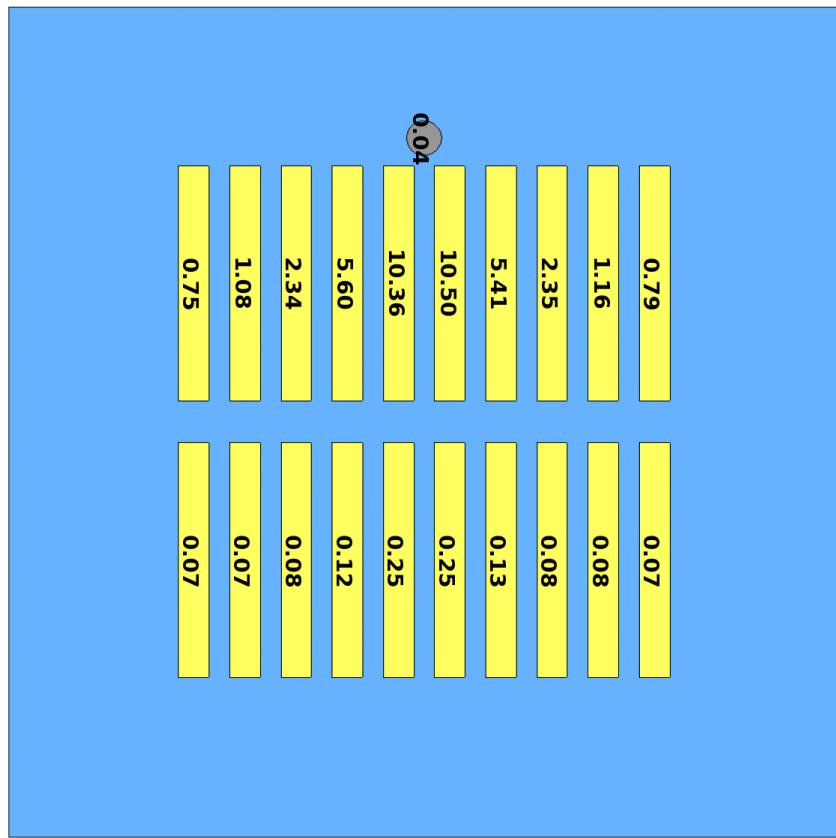


Figure 3.11: Heat contribution by each source immediately after shutdown. Values expressed in W .

meat consists of highly enriched (93 wt%) uranium aluminide (UAl_x) fuel powder dispersed in aluminum. Plates 1 through 4 and 16 through 19 contain natural boron carbide (B_4C) powder as a burnable poison.

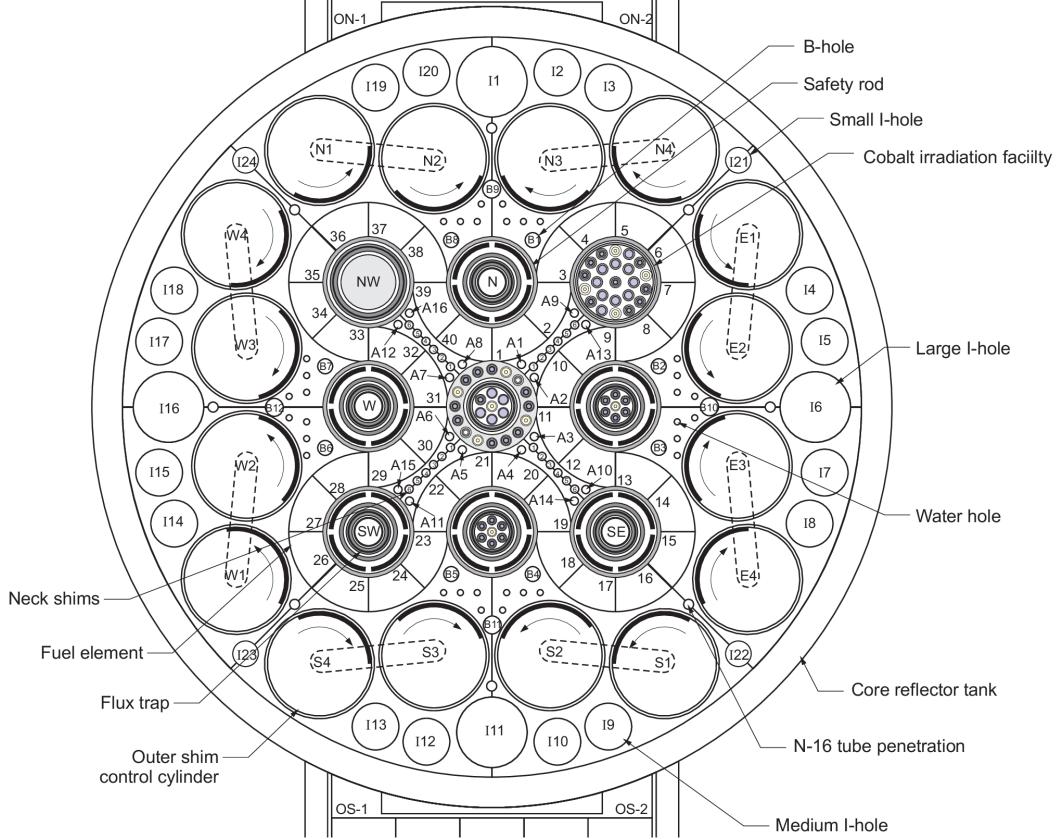


Figure 3.12: Top view of the ATR. Image reproduced from [82].

The reactor has 9 flux trap positions and 68 additional irradiation positions inside the reactor core reflector tank. The center capsule facility is surrounded by the H-hole irradiation facility, which contains 10 cobalt baskets, 4 flux monitor holders, and 2 N-16 flow tubes. The neck shim housing is the structural guide for the 24 neck shim and regulating rods, as well as 8 inner and 8 outer A-holes. The beryllium reflector fills the space between the fuel annulus and the core reflector tank, and it hosts the outer shim control cylinders (OSCC), irradiation holes, and water holes. The irradiation holes include a total of 8 small B-holes, 4 large B-holes, 4 small I-holes, 16 medium I-holes, 4 large I-holes, and 4 round and 4 square N-16 monitor holes.

This work uses an MCNP model representing the ATR critical experiment evaluation designated as Cycle 103A-2 that was performed in 1994. This critical evaluation was part of a nuclear re-qualification program that followed the core internals change-out. The critical configuration was defined by the following parameters: 40 fresh fuel elements, OSCC set to 51.8 degrees, 22 shim rods fully inserted, 2 regulating rods fully withdrawn, and 6 safety rods fully withdrawn.

The ATR critical evaluation was published by the OECD NEA Nuclear Science Committee in the September 2005 Edition of the International Handbook of Evaluated Criticality Safety Benchmark Experiments (ICSBEP Handbook) [82]. The original benchmark model uses the evaluated continuous energy ENDF/B-V cross-section data for 27 °C. This work uses the ENDF/B-VIII.0 cross-section library for the isotopes included in it. The remaining isotopes use the cross-section data of the original model. The model uses 10^4 histories per

generation with 250 and 3750 inactive and active cycles. This work considers the reactor composition at the beginning of the equilibrium cycle (BOC) based on the available ATR model.

This demonstration exercise focuses on the delayed heating of the A1 experiment position. The A1 experiment position is an inner A-hole, and it is hosted by the neck shim housing. The diameter of the hole is 0.625 in. This exercise considers an experiment sample made of aluminum.

Although the ATR maximum power rating is 250 MWth, most contemporary experiments do not require such a level, and the reactor operates at a much lower power. This work considers for the calculations a total operating power of 110 MWth that is equally distributed among the 5 lobes. The irradiation time is 30 days, and the decay is recorded in seven non-uniformly distributed steps up to 12 hours after shutdown.

As the experiment position is in the core region between the fuel assemblies, the calculations assume that the fission products generate a much higher gamma source intensity than the activation products in the neighboring cells. Hence, the contribution of the reactor components surrounding the experiment is neglected. The only considered contributors to the heating are all the fuel plates and the experiment itself.

3.3.4 ATR results

To accommodate the MOAA calculations, the MCNP input file underwent the following modifications. The ATR model defines 19 different material compositions that all fuel elements share for their 19 different fuel plates. The material definitions of each fuel plate were duplicated for each fuel element to allow for their independent definition. Additionally, the volume of each fuel plate was calculated and added into the MCNP input file.

Figure 3.13 presents the experiment heat production over time. $H_{\gamma,Tr}$ was calculated for 41 independent photon-transport simulations, wherein each simulation accounts for the contribution to delayed gamma heating of each fuel assembly and the experiment itself.

The gamma heating immediately after shutdown was 849.2 W, and the heating due to charged particles was 56.32 W, which equals to 6.6% of the total heating in the experiment. The self-gamma heating was 5.8 W, which is equal to 0.73% of the total delayed gamma contribution and 0.68% of the total heating. These results show that the self-heating in aluminum is very low and validate the initial simplification of neglecting the surrounding structure contributions given that both the experiment and the structures are made of aluminum.

Additionally, the heating due to charged particles decays much faster than the gamma heating and becomes negligible one hour after shutdown. The gamma heating decays to 100 W at approximately six hours after shutdown.

Figure 3.14 shows the contribution to the total heat in percentage by each fuel assembly and the experiment immediately after shutdown. The largest contributions to the heating in the sample come from the fuel assemblies 1, 10, 40, 2, 11, 9 with contributions of more than 3%. These assemblies are the closest in distance to the experiment position. Additionally, the sample produced a self-heating of 7.3%, from which 6.6% came from the charged particles, and 0.7% from the gamma particles.

3.3.5 RA-6 structures

The RA-6 is a 3-MWth open pool research reactor located at the Bariloche Atomic Center (CAB), a research and development center of the Argentine National Atomic Energy Commission (CNEA) [82]. It was originally conceived as a teaching tool to support student training at the Balseiro Institute before its uses became

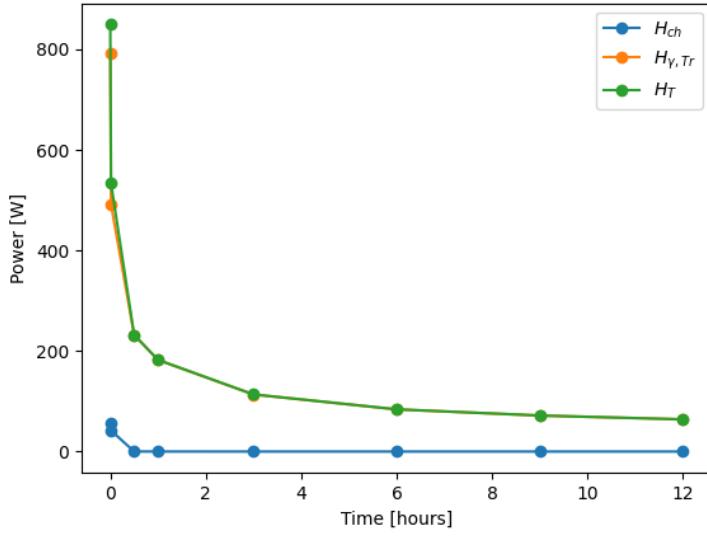


Figure 3.13: Shutdown heating rate over time in the ATR A1 experiment position for an aluminum sample.

diversified. Current applications include neutron activation analysis, neutron radiography, and Boron Neutron Capture Therapy (BNCT), among others.

The reactor core is composed of a fuel element arrangement, as shown in Figure 3.15, that rests on a support grid inside a 2.4-m-diameter, 10.4-m-high stainless-steel tank. The tank is filled with demineralized light water, which acts as a coolant, moderator, and reflector. The fuel elements are Material Test Reactor (MTR) type, composed of rectangular fuel plates with aluminum side plates. The fuel meat consists of 19.7% enriched uranium silicide (U_3Si_2) dispersed in an aluminum matrix. There are two types of fuel elements: Normal Fuel Elements (NFEs) and Control Fuel Elements (CFEs). While an NFE is composed of 17 internal and two external fuel plates, a CFE is composed of 14 internal and 4 control guide plates.

The bottom plane of the core assembly rests 8.9-meters-deep on the support grid, which is a 20 cm-thick slab of 99.5% pure aluminum. Primary and secondary cylindrical holes allow the cooling of the internal and external fuel plates, respectively. The primary holes have a diameter of 6.179 cm and are arranged in a rectangular 8x10 array with 7.7-cm pitch in the x-direction and 8.1-cm pitch in the y-direction. The secondary holes have a diameter of 2.225 cm and are centered between the primary holes.

On a side of the fuel element arrangement, there is a neutron filter that works as a source for BNCT, as shown in Figure 3.16. The neutron filter for the BNCT facility is an 87.6-cm-long, 77.1-cm-wide, 82.35-cm-high aluminum container. The filter is filled along the 87.6-cm dimension with the following materials: 17 cm of aluminum bricks, a 0.15-cm-thick cadmium sheet, 10 cm of aluminum, a 0.15-cm-thick cadmium sheet, and the rest is filled with alumina bricks.

The RA-6 reactor operated for 20 years with spent HEU fuel from a higher power reactor and was converted to LEU fuel in 2008. The RA-6 critical experiment reported in this evaluation corresponds to the 2008 configuration utilized in the startup program. This critical evaluation was published by the OECD NEA Nuclear Science Committee in the September 2010 Edition of the ICSBEP Handbook. This work considers the reactor composition at the BOC based on the available RA-6 model and uses the evaluated continuous energy ENDF66 cross-section data for 27 °C, which is defined in the original benchmark model.

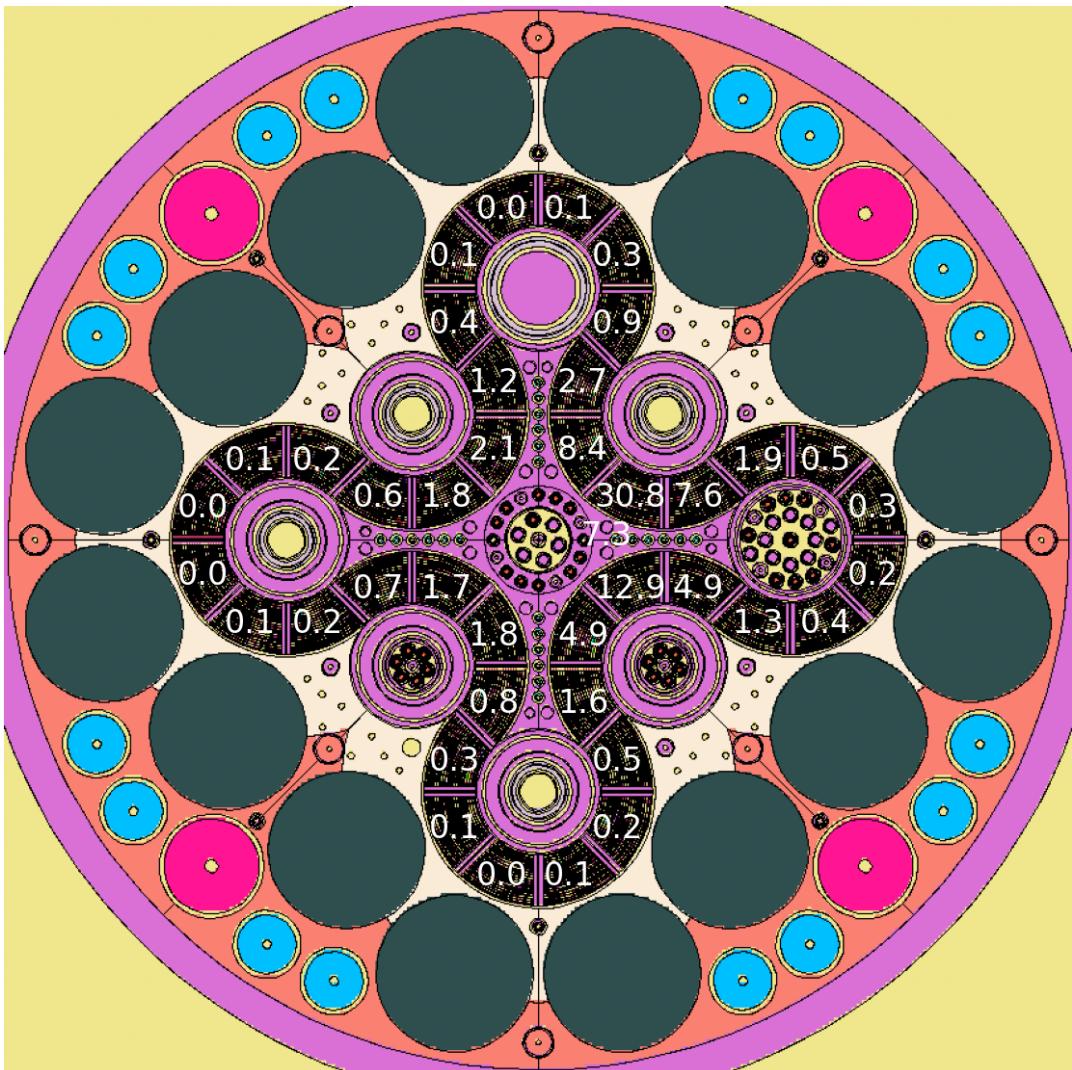


Figure 3.14: Contribution by each source to the A1 experiment heat rate immediately after reactor shutdown, with the values expressed in (%). The Figure is based on the MCNP model geometry, which is rotated 45° clockwise with respect to Figure 3.12.

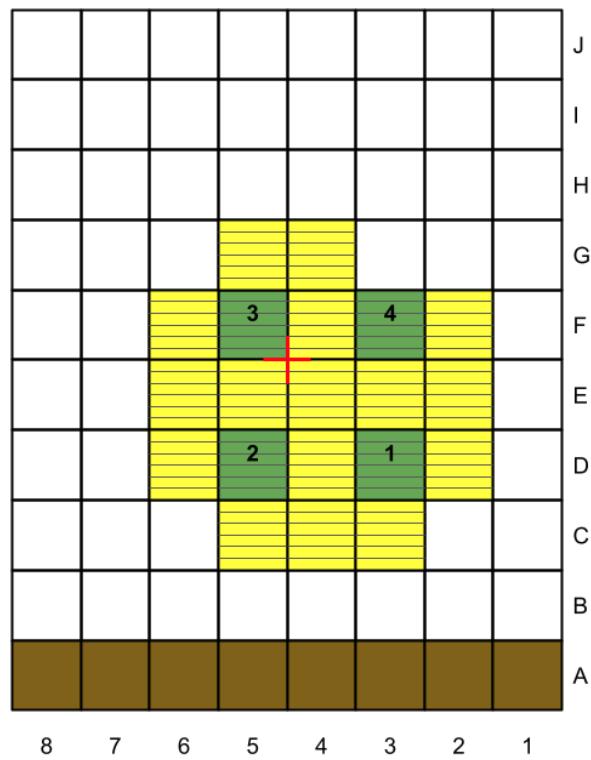


Figure 3.15: Top view of the RA-6 critical configuration. NFEs in yellow, CFEs in green, and BNCT filter in brown. Red cross denotes the center of the reactor pool. Image reproduced from [82].

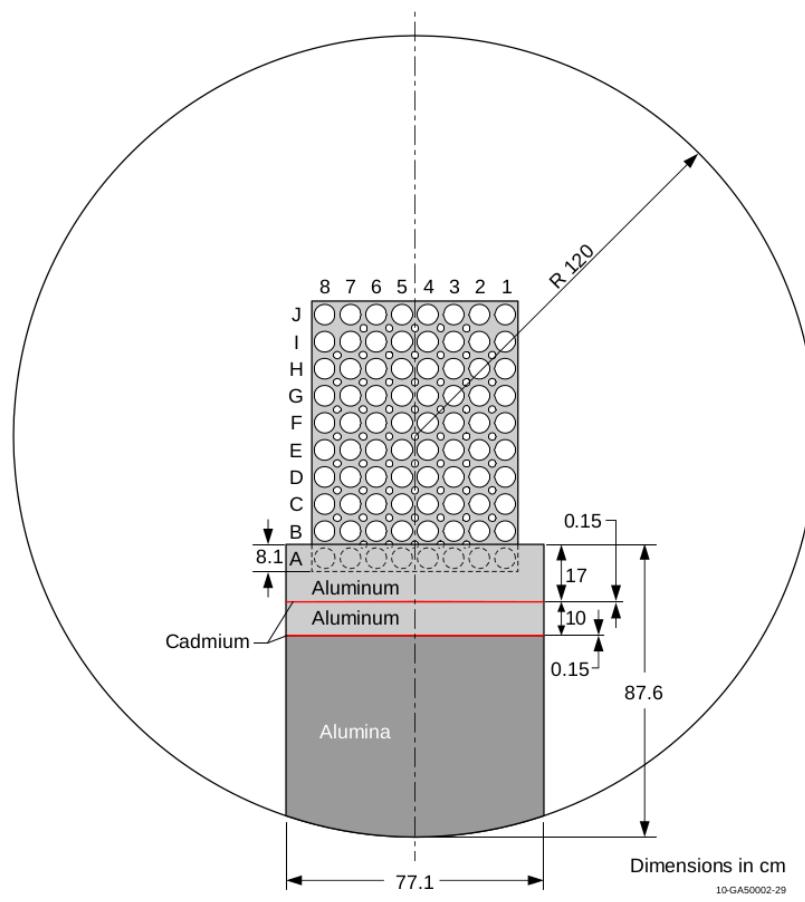


Figure 3.16: Top view of the RA-6 support grid and BNCT filter. Image reproduced from [82].

The simulation uses 10^4 histories per generation with 50 and 4000 inactive and active cycles.

Although not shown in the MCNP model, the reactor core has several surrounding components, such as primary pipes, the stainless-steel tank, and neutron beam extraction tubes. The analysis of delayed heating in these components could be worthwhile but was omitted as the MCNP model excluded them. Hence, this work calculates the delayed heating on the most important structural components in the model, i.e., the support grid and the BNCT filter.

The RA-6 was originally designed to operate at 1 MWth and the 2008 startup program included a reactor power upgrade to 3 MWth. Although the RA-6 maximum power rating is 3 MWth, it normally operates at a lower power. This work considers operating powers of 1 and 3 MWth and compared their respective results. In this work, the reactor operated for 1 year and the depletion was calculated in four 3-month steps. The considered contributors to the heating are the fuel plates, the support grid, and the BNCT filter.

3.3.6 RA-6 results

To accommodate the MOAA calculations, the MCNP input file underwent the following modifications. First, even though the RA-6 model distinguishes between five different types of fuel plates: internal plates in the nuclear fuel elements (NFEs), internal plates with burnable poisons in the NFEs, external plates in the NFEs, fuel plate with burnable poisons in the control fuel elements (CFEs), and fuel plate without burnable poisons in the CFEs, the original MCNP input file defines all of them with the same material definition. Additionally, several layers of the BNCT filter and the support grid are made of pure aluminum, and the original MCNP input file defines the material aluminum only once. The BNCT filter also uses two cadmium layers that share the same material definition in the original MCNP input file. All material definitions of these components were duplicated to allow for their independent definition. Second, the original MCNP input file defines the four CFEs using nested structures with independent cells, but as their definition is identical, a minimal universe renumbering allowed for their simultaneous modeling in MOAA. Third, the volume of each fuel plate type, the volume of the different BNCT filter layers, and the volume of the support grid were calculated and added into the MCNP input file.

Figure 3.17 shows the delayed heating in the support grid and the different BNCT filter layers for operating powers of 1 and 3 MWth, and Table 3.2 shows the deposited heat density in the different structures. The highest deposited heat occurs in the first BNCT layer, while the highest deposited heat density occurs in the second BNCT layer, which is the first cadmium layer in the filter. For the first, third, and fifth layers of the BNCT filter, the deposited heat density decreases due to an increase in the spatial distance from the core. However, the fifth layer has a larger deposited heat than the third layer due to its considerably larger volume. The deposited heat in the cadmium sheets is considerably smaller than for the rest of the BNCT layers due to their smaller volume. Table 3.2 also shows the comparison between the results for 1 and 3 MWth. An increase in operation power of 200% translated into an increase in delayed heating of between 160-190% for the reactor structures.

3.4 Conclusions

Safety analyses in research reactors require the estimation of heat deposited in experiments and reactor structures after reactor shutdown. These analyses establish the heat-source term of those components, and the subsequent thermal-hydraulics calculations determine if an accident could jeopardize their integrity. Additionally, detailed calculations help guide the optimization process of the design of irradiation targets.

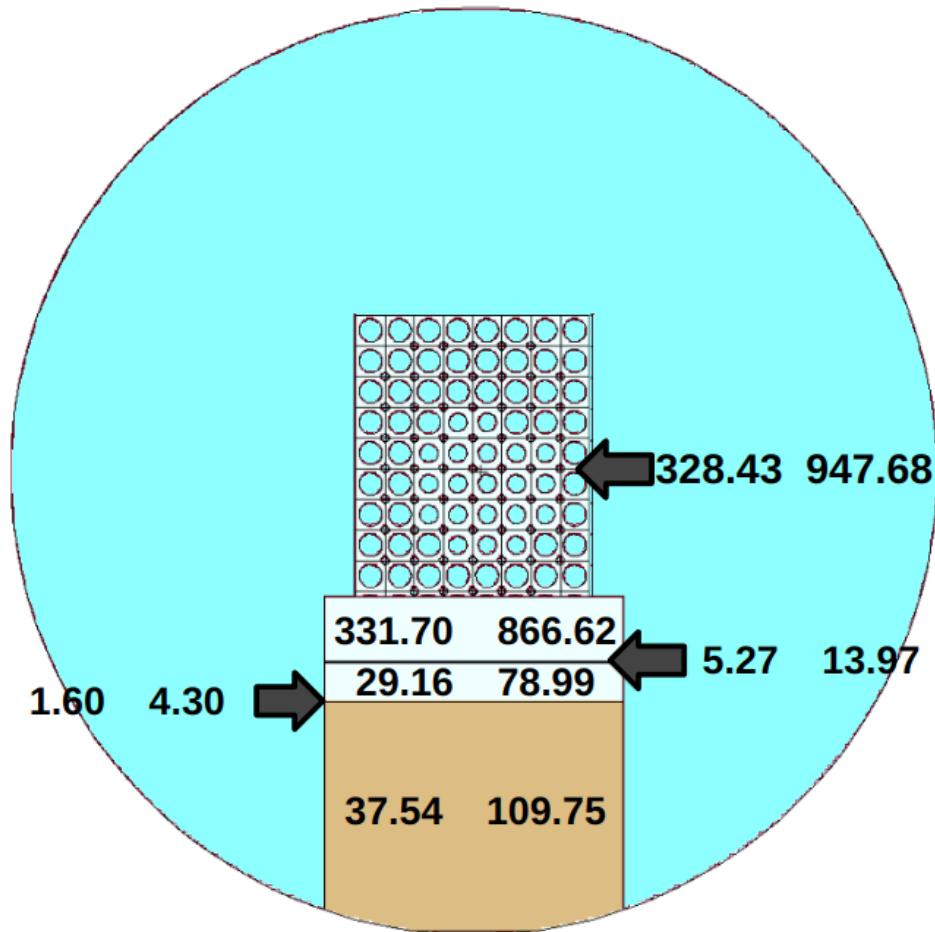


Figure 3.17: Delayed heating in the RA-6 reactor structures. Values expressed in W . Left: values corresponding to 1 MWth. Right: values corresponding to 3 MWth.

Table 3.2: Main results for delayed heating in the RA-6 reactor structures for operating powers of 1 and 3 MWth.

	$H_{T,1MW}/V[W/m^3]$	$H_{T,3MW}/V[W/m^3]$	Rel. Diff. [%]
Support grid	3291.14	9496.55	189 %
BNCT 1st layer	3073.12	8029.02	161 %
BNCT 2nd layer	5533.52	14668.54	165 %
BNCT 3rd layer	459.27	1244.10	171 %
BNCT 4th layer	1680.01	4515.01	169 %
BNCT 5th layer	102.13	298.57	192 %

In general, existing research reactor safety analyses have a strong focus on the reactor core and assume that the energy released after shutdown is deposited locally in the region of interest. Moreover, many software packages rely on the same assumption, which may be inadequate in some cases. On the other hand, an accurate assessment of the deposited energy across the reactor geometry allows for better determination of the heat removal requirements and ensures an effective cool down. Instead of assuming locally deposited heat, this work utilizes detailed methods to target the specific reactor regions.

Chapter 2 discussed various delayed heating calculation methods, and several publications relying on them. The fact that the formal 3-step process accounts for the activation product decay and the time evolution after shutdown makes it better suited for this thesis than other methods. Therefore, this chapter introduced a delayed heating calculation workflow based on the formal 3-step process that relies on MOAA.

MOAA is a Python package that couples MCNP and ORIGEN to streamline the calculation of the experiment source terms for experimental reactors. This chapter described MOAA’s workflow and the delayed heating calculation workflow and several practical aspects of the simulations. The delayed heating calculation relies on MOAA for conducting the first two steps of the process, while the third step is conducted by multiple MCNP photon transport simulations.

Finally, this chapter demonstrated the delayed heating calculation capabilities by presenting four exercises. The first exercise consisted of a code-to-code comparison between the workflow presented here and the Serpent Monte Carlo tool. The purpose of this exercise was to understand the foundation of the method by comparing the results with Serpent. The second exercise demonstrated the calculation workflow for a simple case geometry. This exercise allows to better visualize the method and current capabilities. The third and fourth exercises focus on full-size reactor exercises. These exercises include the delayed heating in an ATR experiment and the RA-6 structures. For the ATR experiment, the results showed the time evolution of the delayed heating up to 12 hours after shutdown, as well as the contribution to the heating from the different sources in the reactor. For the RA-6 structures, the results displayed the delayed heating in several of the structural components and the comparison of the heating values for different power levels.

As discussed in the results section, the calculation workflow can accommodate almost any MCNP input file, with some minimal modifications. This workflow streamlines the calculation of delayed heating in research reactor experiments necessary for the development of experiment safety analysis. Overall, this chapter presented delayed heating capabilities that rely on MOAA and that can be applied to any reactor.

Chapter 4

Shutdown Dose Rate

While previous work utilized MOAA for delayed heating calculations [72]–[74], a modification to the calculation workflow allows it to obtain the shutdown-dose rate. The calculation workflow for the shutdown dose rate and delayed gamma heating is very similar, and they only change in how the deposition of energy manifests, either dose or heat.

The purpose of this chapter is to demonstrate the shutdown dose rate calculation workflow. The development of experiment safety analysis also requires the determination of a PIE strategy, including the estimation of the dose rate in the surroundings of the experiments. A capability like this allows to establish a PIE of an experiment from the dose rate standpoint, as well as design optimization. The equivalent dose rate from delayed gamma radiation is also an important factor for safely conducting work after reactor shutdown, such as periodic maintenance, fuel shuffling, among others [84].

Motivated by the calculation of shutdown dose rates in experiments, one experiment conducted at ATR, the AGR-1 [3], requires special attention. The AGR-1 was an experiment fueled with TRISO particles, which is a type of fuel form that is commonly found in HTGRs. So, in order to properly model the AGR-1 experiment, it is insightful to understand the calculation workflow for shutdown dose rates in HTGRs.

Many publications have studied the shutdown dose rate in LWRs. For example, Abrefah et al. [85] estimated the dose rate of the spent fuel of the Ghana Research Reactor-1 (GHARR-1) to establish a proper unloading strategy for the HEU to LEU conversion of the reactor. In another work, Ambrožic and Snoj [86] studied the dose rate in the Jozef Stefan Institute (JSI) TRIGA reactor and validated their results against experimental measurements during normal operation and after shutdown. Bagheri et al. [87] determined the dose rate from the spent fuel of the Tehran Research Reactor (TRR), validated their calculations against experimental measurements, and studied the photon emission rate of several isotopes for different fuel burnup values.

Although the shutdown dose rate associated with LWRs has been extensively studied, there are not many publicly available studies that focus on HTGRs. For example, Suwoto et al. [88] calculated the neutron dose rate in the working areas outside the Reactor Pressure Vessel (RPV) of the HTGR-10 reactor, concluding that the dose rates were compliant with the stipulated limits. Ueta et al. [89] studied the shielding performance for the upper structure of the High-Temperature engineering Test Reactor (HTTR) during normal operation. Their study considered the active core, spent fuel, and equipment installed in the primary circuit as neutron and gamma radiation sources, and concluded that the neutron contribution to the dose was over 90%. Another study by Hamzah et al. [90] analyzed the dose rate distribution in the Reaktor

Daya Eksperimental (RDE), a 10 MWth experimental pebble-bed reactor, during normal operation and after shutdown. Their study estimated the dose rates in the working area and outside the RDE reactor building, concluding that the calculated values were below the stipulated limits. Ho et al. [84] calculated the shutdown gamma distribution in the HTTR, relying on MCNP6 [27] and ORIGEN2 [91] to obtain the gamma source distribution and MCNP6 to conduct the gamma transport at multiple cooling times. Their work utilized the repeated structures from MCNP6 to model the decay sources from all the fuel rods explicitly, with the gamma radiation emitted from each rod with a uniform spatial distribution. Most of the studies described above focus on the dose rates during normal operation or perform some level of homogenization to avoid the explicit modeling of the TRISO particles as decay radiation source.

The objective of this work is to introduce a new shutdown dose rate calculation capability for TRISO-fueled experiments and HTGRs. The modeling of HTGRs is challenging due to the fuel double heterogeneity, a distinctive characteristic in this type of reactor due to its fuel nature - i.e., the TRISO particles. Previous work [72]–[74] utilized MOAA to calculate delayed heating by treating each depleted cell as an independent source of radiation. However, one single HTGR core contains millions of TRISO particles, meaning that modeling each cell independently is not realistically possible. This chapter of the thesis investigates the modeling of the AGR-1 experiment using the repeated structures from MCNP to explicitly model the TRISO particles as decay radiation source.

This chapter is organized as follows. Section 4.1 details the main aspects of the calculation workflow. Section 4.2 introduces a verification exercise consisting of the delayed heating calculation of a simple geometry. Section 4.3 discusses an exercise studying the shutdown-dose rates of the AGR-1 experiment. Additionally, Section 4.4 uses the workflow described here to study potential decommissioning strategies of a high-temperature gas-cooled microreactor.

4.1 Methodology

The calculations use the formal 3-step process to obtain the shutdown dose rates, consisting of the following three steps:

- First, a neutron-transport simulation determines the neutron flux spatial and energy distribution during reactor operation.
- Second, an activation calculation estimates the energy distribution and emission probability of the decay photon sources after reactor shutdown.
- Third, a photon-transport simulation evaluates the dose rate.

As we can see, the calculation workflow for the shutdown dose rate and delayed gamma heating (Sec. 3.2.1) is identical, and the only change is the evaluation of energy deposition resulting from the photon transport. Overall, the calculation scheme, shown in Figure 4.1, follows the formal 3-step process, in which MOAA conducts the first two steps, and an MCNP photon-transport simulation carries out the third step and estimates the dose rate with the following formula

$$\dot{D}[Sv/h] = F4:p[pSv/src] \times \sum_i S_i \times 3.6 \times 10^{-9} \quad (4.1)$$

$$S_i[\gamma/s] = \int_E \varphi_i^\gamma(E) dE \quad (4.2)$$

$$s_i[-] = S_i / \sum_j S_j \quad (4.3)$$

where \dot{D} is the dose rate, $F4:p$ is the F4 tally for photons, S_i is the total photon emission rate from region i , $\varphi_i^\gamma(E)$ is the photon source energy distribution from region i , and s_i is the source emission probability of region i . $F4:p$ result is expressed in pSV per source particle by adding fluence-to-dose conversion factors [92] to the MCNP input file. The conversion factors consider an antero-posterior exposure to photons.

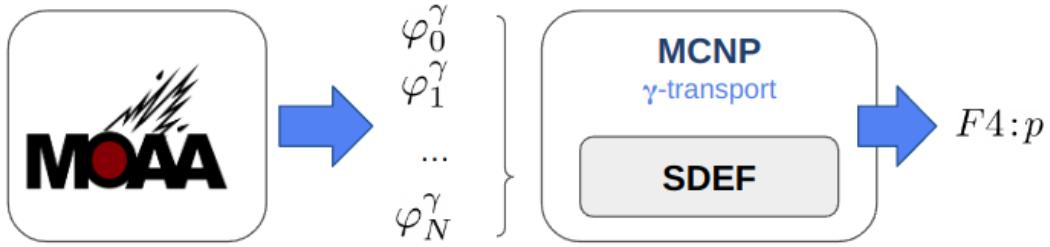


Figure 4.1: Shutdown dose rate calculation scheme.

4.2 Verification

This exercise consists of the delayed gamma heating calculation in a simple geometry. The calculation workflow for the shutdown dose rate and delayed gamma heating is identical, and they only change in how the deposition of energy manifests, either dose or heat. Three calculations were performed for this exercise. The first calculation employs the explicit independent definition of each cell and provides the reference values. The second calculation utilizes the repeated structures' approach.

The following equation calculates the delayed gamma heating

$$H_{\gamma,Tr}[W] = 1.6022 \times 10^{-13} \times *F8[MeV] \times \sum_i S_i \quad (4.4)$$

where $H_{\gamma,Tr}$ is the energy deposited in the region of interest resulting from the photon transport, $*F8$ is the energy deposition calculated by an $*F8$ tally, and S_i is the total photon emission rate from the region i (see Eq. 3.19).

The problem geometry is displayed in Figure 4.2. The core consists of an 8x8 array of fuel pins of 1.25 cm radius with a 4 cm pitch, and it is separated from the reflector tank by an inner shell of aluminum. The shell dimensions are an inner side length of 32 cm and 2 cm thickness. The reflector tank is filled with light water, and its radius is 40 cm. The whole geometry is 80 cm high. The fuel is 7.8%-enriched uranium with a density of 10.5 g/cm³, the inner shell is made of pure aluminum with a density of 2.7 g/cm³, and the water has a density of 1.0 g/cm³.

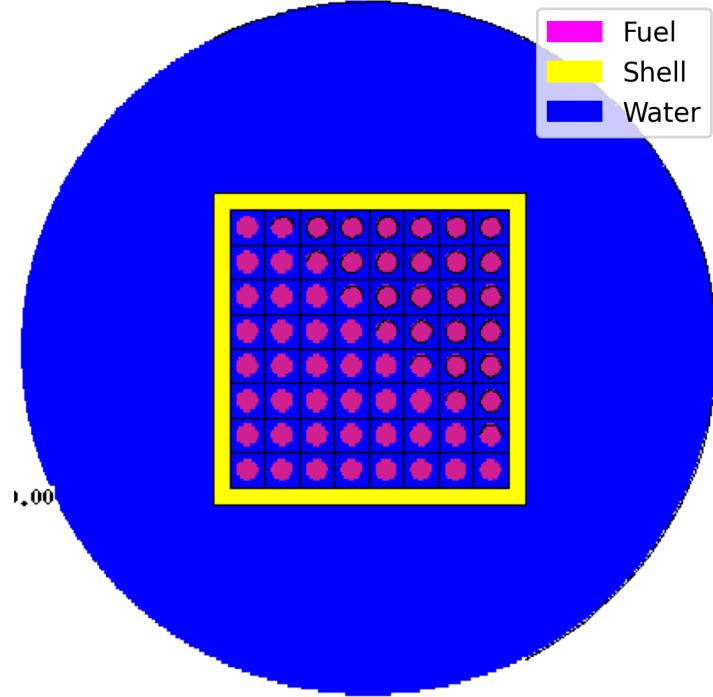


Figure 4.2: Verification problem geometry. The delayed gamma heating is calculated in the core inner shell.

The main objective of the calculation is the estimation of the delayed gamma heating in the core inner shell, considering all the fuel pins as the radiation source. The delayed heating is calculated after a reactor operation at a 1 MW constant power for a 12-month cycle.

Table 4.1 summarizes the main results at the end of irradiation. While the heating equals 675.0 W in the reference case, the repeated structures' case predicts a heating of 780.4 W, which is 15.6% larger than the reference value. To understand this result, we need to investigate other quantities. The average neutron flux and total photon intensity show a relative difference smaller than 0.1%. Figure 4.3 displays the uranium composition evolution during irradiation. The uranium evolution displays a relative difference smaller than 0.1%.

Table 4.1: Comparison of end of irradiation integral results for the reference and repeated structure approaches.

	Reference	Rep. Struct.	Rel. Diff. [%]
Delayed gamma heating [W]	675.0	780.4	15.6
Ave. neutron flux [10^{13} n/cm 2 /s]	1.868	1.867	0.07
Tot. photon intensity [10^{17} γ /s]	3.666	3.668	0.04

Figure 4.4 displays a comparison between the repeated structures and the reference cases of the gamma source intensities in the fuel pins. The sources in the repeated structures' case have a uniform value across the geometry. The definition of the repeated structures requires the utilization of universes, and all the cells defined by the same universe are subject to the same flux level, an average value, and hence their depletion is uniform. The sources in the reference case are strongest in the center of the core and decrease towards the periphery, due to the flux shape, which is strongest in the center. Finally, the gamma source intensities in the

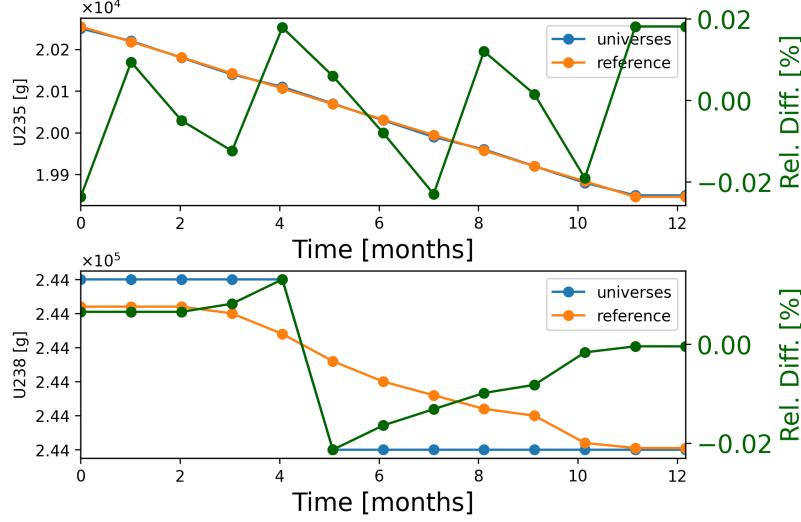


Figure 4.3: Comparison of the uranium mass during reactor operation.

repeated structures' case are stronger in the periphery than in the reference case. As these pins contribute the most to the heating of the shell due to their proximity, the delayed heating in the core inner shell is higher.

Total universes: 3.67e+17, Total reference: 3.67e+17, Diff: 0.0%

5.73e+15 3.78e+15 51.5	5.73e+15 4.24e+15 35.2	5.73e+15 4.89e+15 17.1	5.73e+15 5.23e+15 9.6	5.73e+15 5.24e+15 9.4	5.73e+15 4.89e+15 17.2	5.73e+15 4.25e+15 34.7	5.73e+15 3.79e+15 51.3
5.73e+15 4.25e+15 34.9	5.73e+15 4.97e+15 15.4	5.73e+15 5.84e+15 -1.9	5.73e+15 6.30e+15 -9.1	5.73e+15 6.29e+15 -8.9	5.73e+15 5.85e+15 -2.1	5.73e+15 4.98e+15 15.0	5.73e+15 4.25e+15 35.0
5.73e+15 4.88e+15 17.5	5.73e+15 5.84e+15 -1.8	5.73e+15 6.89e+15 -16.8	5.73e+15 7.43e+15 -22.8	5.73e+15 7.44e+15 -22.9	5.73e+15 6.91e+15 -17.1	5.73e+15 5.84e+15 -1.8	5.73e+15 4.92e+15 16.6
5.73e+15 5.24e+15 9.4	5.73e+15 6.31e+15 -9.2	5.73e+15 7.43e+15 -22.9	5.73e+15 8.03e+15 -28.6	5.73e+15 8.03e+15 -28.7	5.73e+15 7.43e+15 -22.9	5.73e+15 6.30e+15 -9.1	5.73e+15 5.25e+15 9.2
5.73e+15 5.25e+15 9.3	5.73e+15 6.31e+15 -9.2	5.73e+15 7.45e+15 -23.1	5.73e+15 8.04e+15 -28.7	5.73e+15 8.04e+15 -28.7	5.73e+15 7.44e+15 -23.0	5.73e+15 6.30e+15 -9.0	5.73e+15 5.23e+15 9.7
5.73e+15 4.89e+15 17.2	5.73e+15 5.84e+15 -1.8	5.73e+15 6.90e+15 -17.0	5.73e+15 7.44e+15 -23.0	5.73e+15 7.46e+15 -23.2	5.73e+15 6.89e+15 -16.8	5.73e+15 5.85e+15 -2.0	5.73e+15 4.90e+15 16.9
5.73e+15 4.26e+15 34.4	5.73e+15 4.98e+15 15.0	5.73e+15 5.85e+15 -2.0	5.73e+15 6.28e+15 -8.8	5.73e+15 6.31e+15 -9.2	5.73e+15 5.86e+15 -2.2	5.73e+15 5.01e+15 14.5	5.73e+15 4.25e+15 34.7
5.73e+15 3.81e+15 50.5	5.73e+15 4.24e+15 35.0	5.73e+15 4.89e+15 17.2	5.73e+15 5.24e+15 9.4	5.73e+15 5.25e+15 9.2	5.73e+15 4.91e+15 16.8	5.73e+15 4.27e+15 34.2	5.73e+15 3.80e+15 50.8

Figure 4.4: Comparison of the intensity of the gamma sources in the fuel pins. The top value corresponds to the repeated structure approach. The intermediate value corresponds to the reference values. Intensities expressed in decays per second. The lower value corresponds to the relative difference in %.

As shown in Figure 4.4, the accuracy of the repeated structure approach is affected when the problem shows a strong spatial effect. This suggests that using a repeated structure that comprehends the whole core is not the right approach. Instead, a better approach would consider the flux shape variation to choose the repeated structures. This exercise was also conducted using repeated structures but arranging the fuel pins in groups of 4. Table 4.2 displays the results for that case. The relative difference for the delayed gamma heating is 4.3%, while the rest of the results do not change considerably. The results show that grouping the structures based on the location, and hence similar flux levels, yields better results.

Table 4.2: Comparison of end of irradiation integral results for the reference and repeated structure approaches.

	Reference	Rep. Struct.	Rel. Diff. [%]
Delayed gamma heating [W]	675.0	704.3	4.3
Ave. neutron flux [10^{13} n/cm ² /s]	1.868	1.867	-0.03
Tot. photon intensity [10^{17} γ /s]	3.666	3.667	0.03

This exercise studied the delayed heating calculation in repeated structures, such as the ones that are necessary to define HTGR models. Taking advantage of the MCNP input definition using repeated structures allows for simplifying the definition of cells contributing to the delayed heating/shutdown dose rates. Even though the results for the repeated structures' case show an overestimation of the delayed heating, the approach yields a reasonable approximation when considering the flux spatial variation. The results and conclusions drawn from this exercise set the path forward for the next exercises, which use the same approach for calculating the shutdown dose rate in HTGRs.

4.3 AGR-1

The AGR-1 [3] was a TRISO-particle irradiation test in the ATR, a 250-MWth high flux test reactor located at the Reactor Technology Complex of the INL. This experiment was part of the DOE Advanced Gas Reactor Fuel Development and Qualification Program to support the Next Generation Nuclear Power (NGNP) program.

Figure 4.5 shows the MCNP quarter model of ATR, which represents the core east quadrant, which is used for the simulations. This model simplifies the reactor geometry by grouping the fuel plates into three radial zones. The AGR-1 experiment was placed in the B-10 position and irradiated over 13 power cycles for a period corresponding to three years.

The AGR-1 Depletion Benchmark [3] describes the experiment test train, shown in Figure 4.6. The test train consists of six cylindrical capsules vertically stacked, wherein the compacts are separated into three columns, with each column containing four compacts, adding up to a total of 72 compacts. The experiment includes four different types of compact. Capsules 3 and 6 contain the baseline compact, capsule 5 contains variant 1, capsule 2 contains variant 2, and capsules 1 and 4 contain variant 3. The difference between compact types is that the particle layers change slightly in thickness and density, while the fuel kernel remains the same. Additionally, each type of compact has a different number of embedded particles. The main model simplifications are that the hafnium (Hf) shroud surrounds the whole circumference of the capsule, the gas lines, thermocouples, and thru-tubes are neglected, and the TRISO particles are arranged in a regular lattice.

The benchmark data includes for each time step: the lobe power, the OSCC positions, and the neck shim insertion condition. Figure 4.7 shows the benchmark data for the first irradiation cycle or cycle 138B.

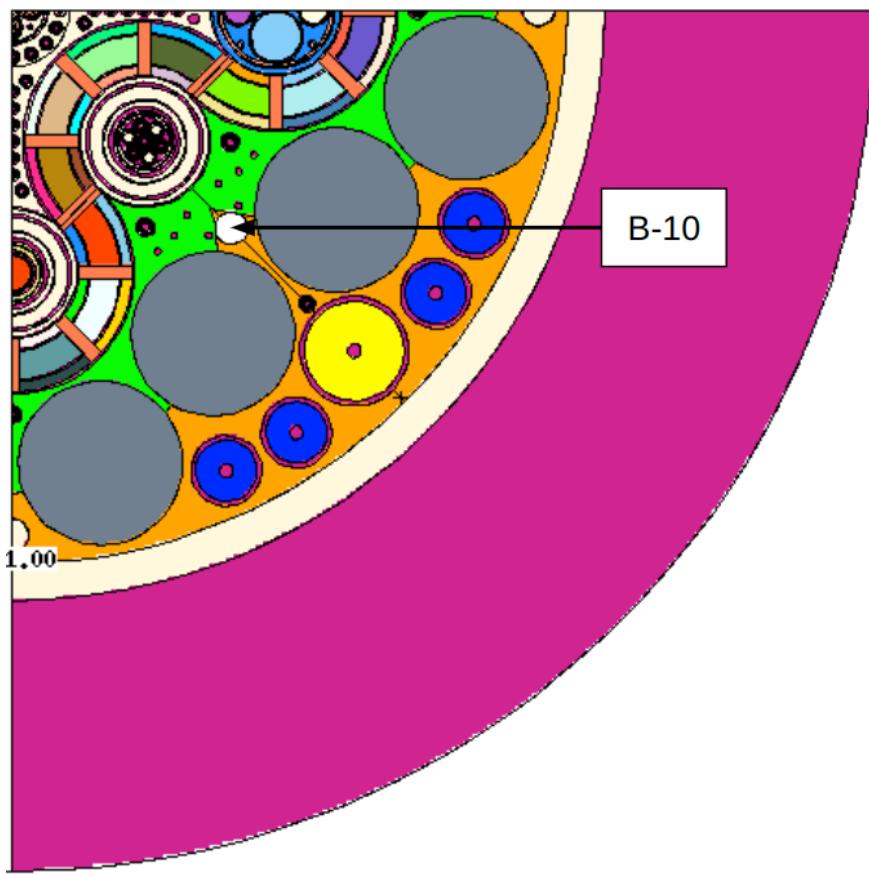


Figure 4.5: MCNP quarter model of ATR.

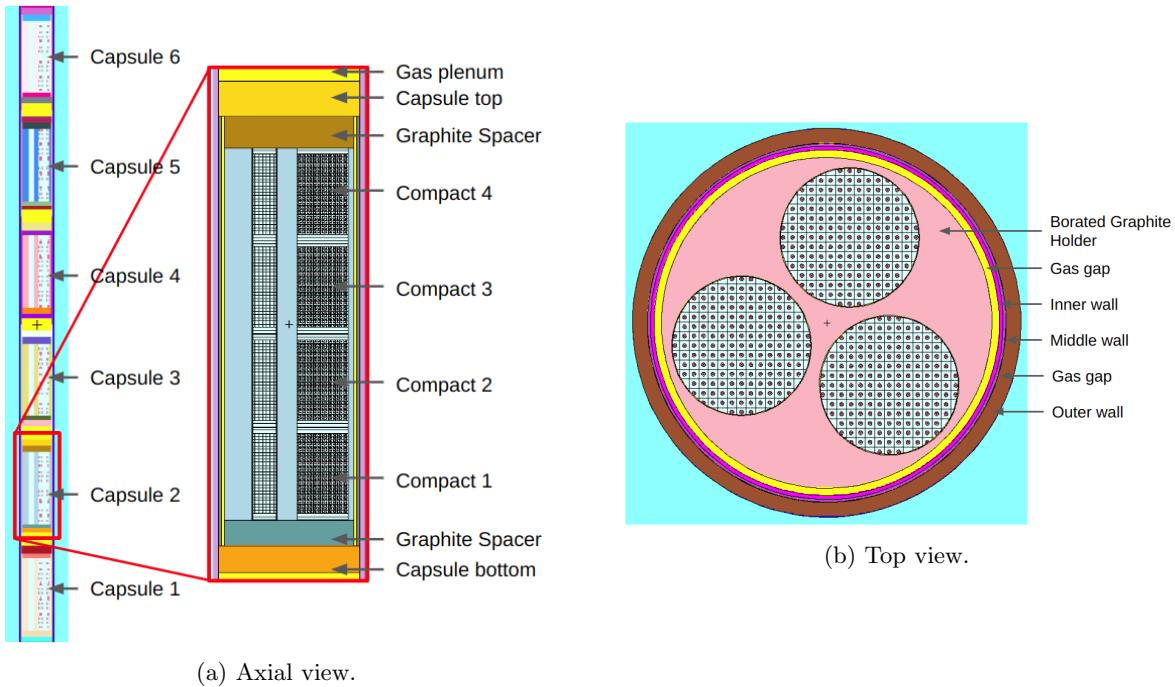


Figure 4.6: AGR-1 experiment geometry.

Because the data includes 662 time steps, and to reduce the complexity of the model, this work considers the power, OSCC position, and neck shim insertion condition to take the cycle average value. For the neck shim insertion, the average value is considered the cycle predominant insertion condition. For example, if the NE 1 rod is inserted through most of the cycle, this work assumes that it is inserted throughout the whole cycle length. For tally normalization, the B-10 position tally data is typically normalized to an east lobe power defined as the average of the northeast, center, and southeast lobe powers. For the fuel composition, the MCNP model defines the driver fuel for the BOC 145A and assumes it to be appropriate for the BOC of all the cycles. This work takes that assumption one step further and considers the driver fuel composition to remain constant during all the simulations.

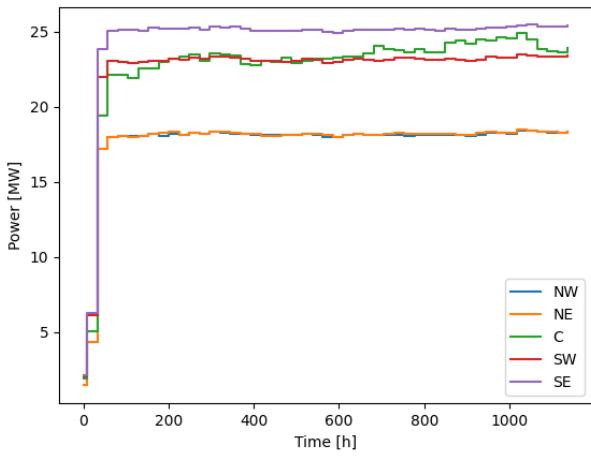
Figure 4.8 displays the experiment burnup as a function of the axial position. This figure compares this work's results versus the benchmark results. As the raw data from benchmark results are not available, this figure shows both results overlapped, allowing for the visualization of the following phenomena. The benchmark results include more data points than this work. While the benchmark subdivides each capsule into eight pieces, this work considers only one average value. Stacks 1 and 2 show the largest burnup values, as they are the closest to the reactor center. This work results agree on this aspect with the benchmark results. The burnup shape for each capsule displays higher values in the bottom and top ends. The reason for this is the presence of the graphite spacers in the bottom and top regions of each capsule. This result is also consistent with the benchmark results. Finally, this work predicts a lower burnup than the benchmark results. The reason for this is the modeling simplification of considering the average values for each cycle, and most importantly, utilizing the cycle average power. Although not shown here, some cycles include a reactor shutdown, in which the reactor total power becomes zero (even for 40 days). For these cycles, the average cycle values simplification is expected to underpredict the cycle burnup. For future reference, this simplification should not be used when trying to follow the benchmark formally. However, for the purpose of this work focusing on shutdown dose rates, this simplification and consistent results are deemed valid.

Table 4.3 shows the contribution from all the photon sources at different decay times. The graphite spacers and holders have the smallest contribution. The holders contain boron and hence have a higher contribution than the spacers. After one day, the largest contribution comes from fuel at 53.19% followed by the Hf shroud with a contribution of 41.64%. After 30 days, the contribution from the fuel decreases to 30.03% while the Hf shroud contribution of 62.11% becomes predominant. After 365 days, the Hf shroud contribution drops, and the largest contributor becomes the fuel at 56.95%, followed by the structures of stainless steel which add up to 42.54%. The total source intensities decrease two orders of magnitude after a 365-day decay.

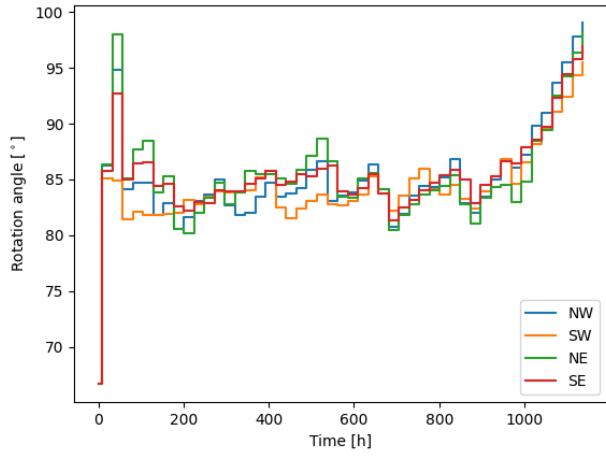
Table 4.4 shows the dose rate at different distances from the center of the AGR test train by decay time. The results show that even after one year decay, and at 98 cm distance from the experiment, the exposure from the experiment decay is high. For that reason, the experiment PIE would need to await more than 1 year to be conducted or utilize a hot cell with the appropriate shielding to prevent any unnecessary exposure of the personnel.

4.4 The μ HTGR

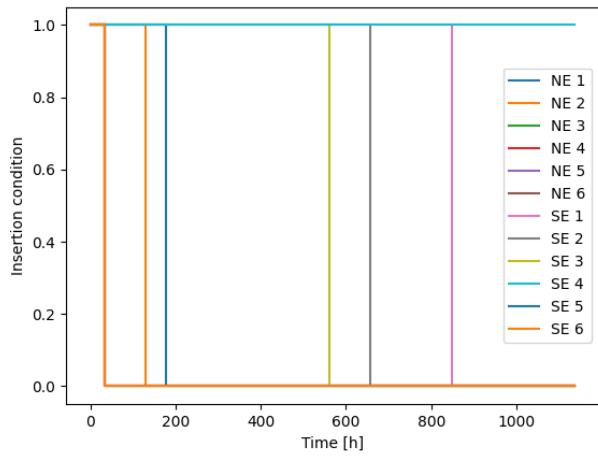
HTGRs have gained a lot of attention in the past few decades. In 2002, the Very High-Temperature Gas-Cooled Reactor (VHTR) was proposed as one of the Generation-IV nuclear energy systems for the co-generation of electricity and hydrogen [93]. In 2006, the DOE and the INL established the NGNP project



(a) Lobe power.



(b) OSCC rotation angle.



(c) Neck shim insertion condition, where 0/1 means withdrawn/inserted.

Figure 4.7: AGR-1 benchmark data for first irradiation cycle (cycle 138B).

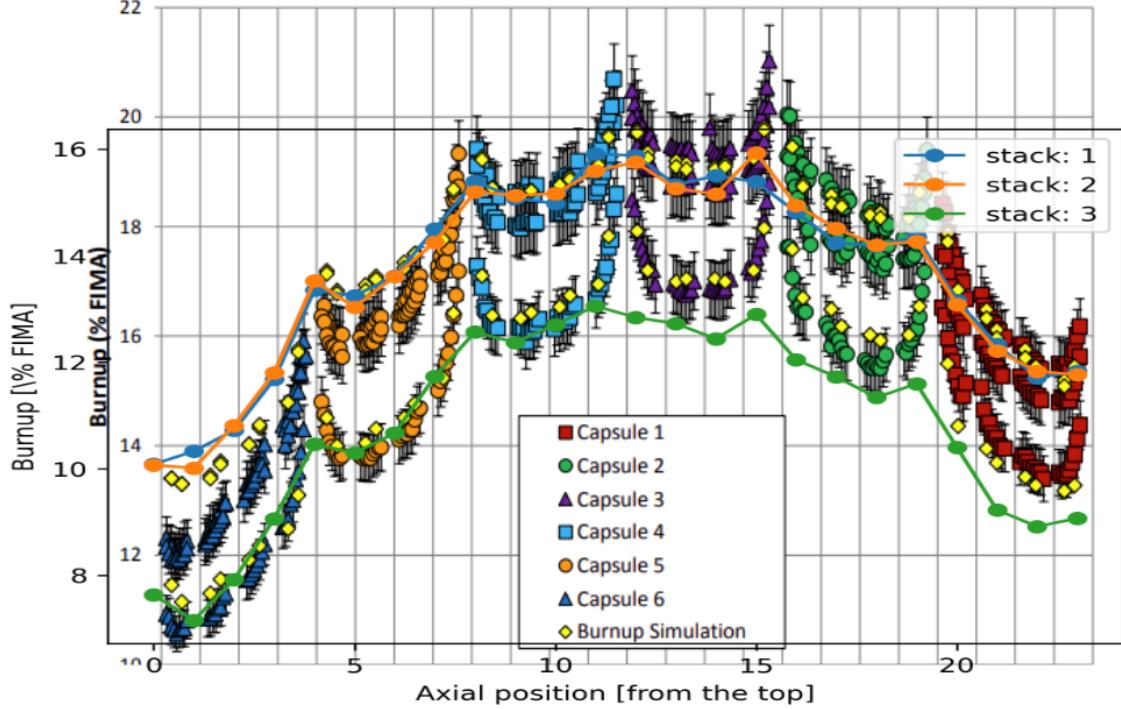


Figure 4.8: AGR-1 experiment burnup as a function of the axial position.

Table 4.3: Photon source contribution by decay time.

Contribution	Units	1 day	30 days	365 days
Fuel	%	53.19	30.03	56.95
Bottom support SS316L	%	0.76	1.17	6.47
Inner wall SS316L	%	0.43	0.63	3.07
Top support SS316L	%	0.85	1.3	7.10
Outer wall SS316L	%	3.13	4.77	25.9
Low graphite spacer	$\%$	6.93×10^{-10}	2.56×10^{-9}	6.37×10^{-8}
Upper graphite spacer	$\%$	8.04×10^{-10}	2.96×10^{-9}	7.38×10^{-8}
Borated graphite holder	$\%$	7.59×10^{-7}	2.77×10^{-6}	6.14×10^{-5}
Hf shroud	$\%$	41.64	62.11	0.50
Total	γ/s	1.199×10^{15}	3.2529×10^{14}	1.305×10^{13}

Table 4.4: Dose rate at different distances from the center of the experiment by decay time. Values expressed in Sv/h.

Distance	1 day	30 days	365 days
6 cm	666.2	190.3	3.7
46 cm	72.7	20.4	0.4
98 cm	25.4	7.3	0.1

to develop, license, build, and operate a prototype modular HTGR to produce hydrogen while generating electric power [94]. More recent examples of HTGRs are new microreactor designs. StarCore Nuclear has partnered with the local government of Pinawa, Canada to demonstrate the powering of off-grid communities and mining companies with an HTGR [95]. The U.S. Department of Defense (DOD) has recently awarded a

contract to BWXT to deliver a transportable 1-5 MWe HTGR to be completed and delivered by 2024 [96]. The UIUC has partnered with the Ultra Safe Nuclear Corporation (USNC) to build a microreactor on its campus, a 10 MWth HTGR [97]. The new reactor facility will offer UIUC a diverse set of opportunities for research, including instrumentation and control, multi-physics validation, reactor prototype testing, and micro-grid operations.

HTGR technology presents many distinct features from the current LWR fleet, and potential decommissioning strategies should take them into consideration. After reactor shutdown, the ongoing decay may inflict dose on workers and equipment in the reactor surroundings. In LWRs, the common strategy entails flooding the entire reactor cavity to remove the spent fuel, as the water column between the assemblies and the operators provides enough shielding. HTGRs cannot, however, implement such a strategy because of the presence of large volumes of high-temperature graphite in the core, and this could lead to graphite oxidation [98], degrading its structural and functional properties. Nevertheless, the proper characterization of the shutdown dose rates can guide establishing a suitable decommissioning strategy.

This exercise uses the shutdown dose rate calculation workflow described above to study a decommissioning strategy for a model based on an early version of the USNC Micro Modular Reactor (MMR) design [97], hereby referred to as μ HTGR [99]. Because most of the technical details of the MMR are not public, the μ HTGR model is based on the publicly available information of the reactor [97], [100]–[102].

The decommissioning strategy places the reactor in a safe shutdown state and allows it to cool down for three months [100]. The plan continues with the removal of the reactor lid at the citadel floor, control rod driving mechanisms, RPV lid, and upper core restraint structures to allow for the fuel assembly unloading before removing the vessel itself. This exercise intends to quantify the dose rate that a worker would experience if standing at ground level on the citadel floor above the reactor top while the reactor core is exposed.

Figure 4.9 displays the μ HTGR core layout. The core region consists of 24 fuel assemblies, 12 control assemblies, and one reserved shutdown assembly in the center. The assembly pitch is 30 cm. Four layers of assemblies stacked on top of each other encompass the core in the axial direction. Additionally, the reactor has a radial, bottom, and top reflectors of graphite. The assemblies and the bottom and top reflectors are 68 cm high and the radial reflector has a 134 cm radius. The model considers a graphite density of 1.75 g/cm³ for the assemblies and reflector regions. Within the assemblies, the fuel and coolant channels have 1.15 cm and 0.775 cm radius, respectively, with a channel pitch of 3.2 cm. While each control assembly has a 4 cm-radius control rod hole, the reserved shutdown assembly has a 6 cm-radius control rod hole.

A column of fuel compacts fills out each fuel channel. The compacts are made of SiC, have a density of 3.2 g/cm³, and hold TRISO particles with a 40% packing fraction. The model considers the particles to be uniformly distributed with a particle pitch of 0.1 cm. Table 4.5 summarizes the characteristics of the TRISO particles.

Table 4.5: TRISO particle definition.

Layer	Thickness [cm]	Material	Density [g/cm ³]
Kernel	0.0250	UO ₂ ($\varepsilon=19.75\%$)	10.8
Buffer	0.0100	C	0.98
IPyC	0.0040	C	1.85
SiC	0.0035	SiC	3.20
OPyC	0.0040	C	1.86

Figure 4.10 displays the side view of the MCNP model. In the radial direction, the model neglects the

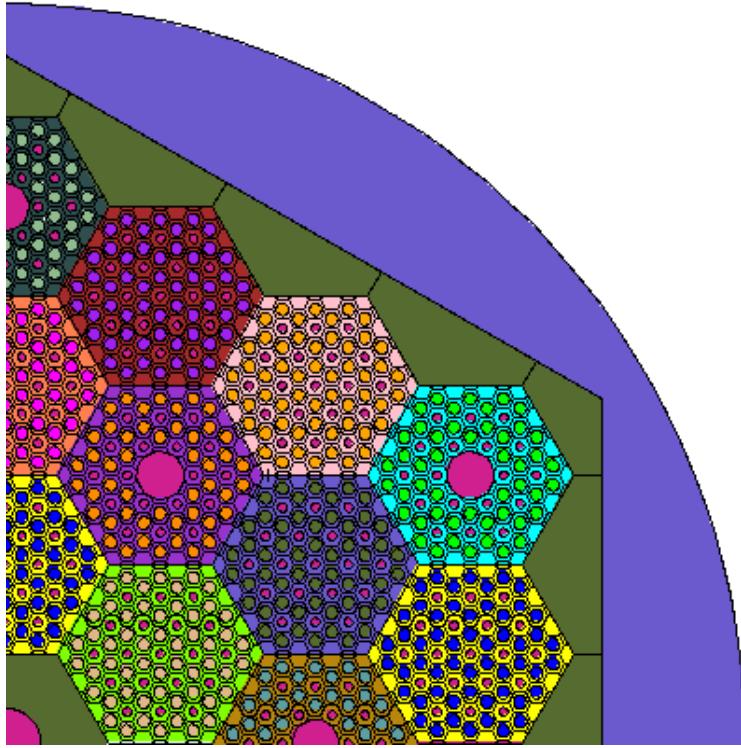


Figure 4.9: Quarter section of the μ HTGR core.

presence of the RPV side wall and considers the reactor to be in a cavity of Portland concrete (2.3 g/cm^3 density). The concrete wall is located 191 cm away from the radial reflector and has a 100 cm-thickness. In the axial direction, the cavity is limited by the citadel floor of Portland concrete with a 70 cm-thickness. The distance from the top of the reactor to the citadel floor (ground level) is 700 cm. The remaining volumes are filled with air (80 vol% nitrogen, 20 vol% oxygen).

The dose rate is calculated for a worker standing at ground level on the citadel floor. The calculations use a standard person (often called “reference man”) of 70 kg weight and 170 cm height, which is represented by an equivalent cylinder. Considering an average human density of 0.985 g/cm^3 yields an equivalent cylinder of 11.53 cm radius.

The calculations consider as radiation sources the fuel in the TRISO particles in all the assemblies and the bottom, top, and radial reflectors. A sensitivity study determined that the radiation contribution from the fuel particles is approximately 8 orders of magnitude larger than the contribution from the graphite in the assemblies. The TRISO particle contribution is separated by assembly, for which each assembly has an independent depletion. Finally, the isotope definitions in the MCNP model used the ENDF/B-VIII.0 cross-sections at room temperature.

Figure 4.11 displays the K_{eff} time evolution. The core has an initial K_{eff} of 1.26797 and reaches the end of life at 16.07 years. Although the original reactor is expected to run for 20 years, the μ HTGR is a simplified version and several of its defining parameters are assumed, as the original parameters are not publicly available. However, for the purposes of this discussion, the assumptions considered yield an adequate model.

Figure 4.12 shows the dose rate map at the citadel after a 3-month cool-down. The region directly above

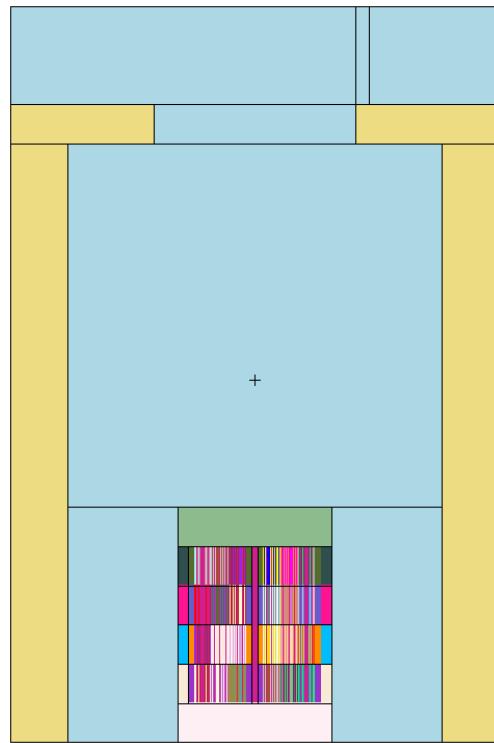


Figure 4.10: Side view of the MCNP model including reactor core, reactor cavity, side walls, and citadel.

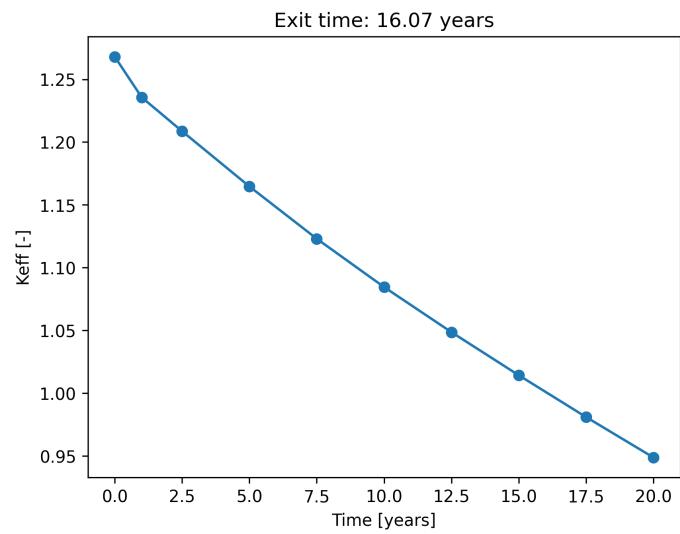


Figure 4.11: μ HTGR K_{eff} time evolution.

the reactor core shows the largest values, some of them being above 40 mSv/h. The dose rate in the standard person is 6.48 ± 1.04 mSv/h.

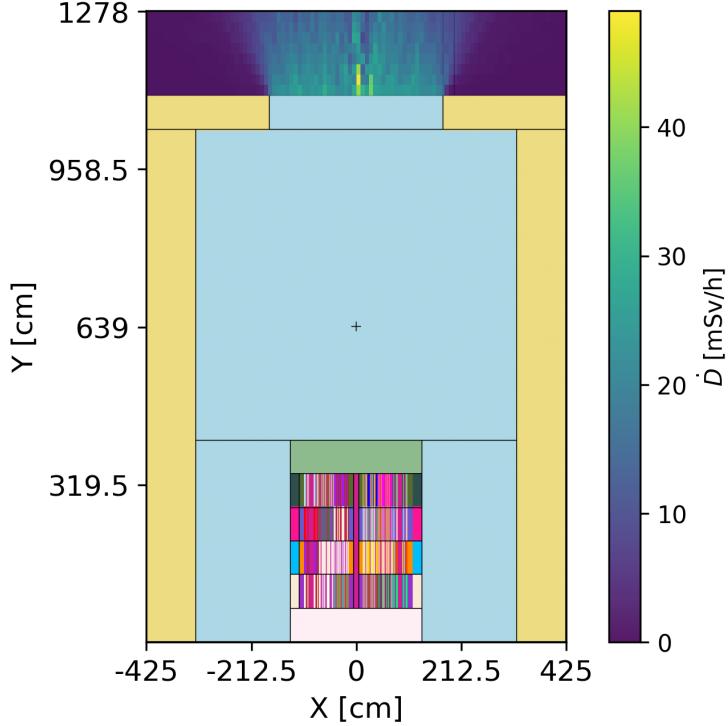


Figure 4.12: Dose rate map at the citadel after 3 month-cool-down.

The annual total effective dose equivalent for the whole body is 50 mSv for radiation workers. Although the calculated dose rate is below this limit, other considerations need to be considered, such as exposure time and worker's distance. These working conditions would allow a radiation worker to perform their tasks for less than 8 hours per year. Additionally, the calculations consider a vertical distance of 7 m between the top of the core and the worker. However, while the assemblies are being removed, such a distance will be considerably smaller, and the dose rate will increase accordingly. For this reason, other decommissioning strategies should be considered to minimize unnecessary exposure. Future work should analyze those strategies, adding shielding to the reactor core/assemblies and allowing for a longer cool-down time.

4.5 Conclusions

This chapter focuses on the shutdown-dose rate calculations, which are necessary for the determination of suitable PIE strategies for experiment irradiation. Benefiting from similar workflows, a minimal modification of the workflow allows to obtain both delayed heating and shutdown dose rate.

Motivated by the calculation of shutdown dose rates in experiments, the AGR-1 experiment sets the path forward for the application of this workflow to HTGRs. To properly model the AGR-1 experiment, it is insightful to understand the calculation workflow for HTGRs. Additionally, as HTGR technology has distinct features from LWRs, their decommissioning requires new strategies that can be guided by the shutdown dose rate calculations.

The current available literature on dose rate calculations of HTGRs is scarce, and it focuses on the normal operation regime. Additionally, previous work on shutdown dose rates of HTGRs lacks the definition of a formal methodology that considers the double heterogeneity of the TRISO fuel. This thesis introduces a new shutdown dose rate calculation capability for HTGRs that relies on MOAA and the MCNP repeated structures to explicitly model the decay radiation emitted from the TRISO particles. MOAA is responsible for conducting the first two steps of the process, while the third step is conducted by an MCNP photon transport simulation.

This work introduces a simple exercise to verify the workflow. For this exercise, one calculation employs the explicit independent definition of each source cell and provides the reference values, while a second calculation utilizes the repeated structures' approach. The results for the repeated structures' case show that the approach yields a reasonable approximation when the flux spatial variations are considered.

Finally, this chapter showcases the shutdown dose rate calculation capabilities by presenting two exercises. These exercises calculate the shutdown dose rate of the AGR-1 TRISO-fueled experiment and a high-temperature gas-cooled microreactor, the μ HTGR. For the AGR-1 experiment, the results showed that the largest contributor to the dose is the fuel for most decay times. However, for different decay times, some of the structures produce comparable intensities to the fuel, such as the Hf shroud and the stainless steel structures. Additionally, the results lead to the recommendation of allowing a decay time of more than one year and the addition of appropriate shielding for conducting the PIE. For the μ HTGR, the results showed that the proposed decommissioning strategy is feasible but not ideal, as it would lead to very high exposures. Hence, these results open the door for future studies, which consider the addition of shielding to the reactor core/assemblies and allowing for a longer cool-down time.

Chapter 5

Generic Irradiation Database

This chapter investigates different methods to accelerate the delayed heating calculation for experiments of any given material composition. The most computationally expensive step in the delayed heating calculation is the neutron-transport simulation during experiment irradiation. This step determines the geometry-dependent parameters, i.e., total neutron flux, neutron spectrum, and reaction rates. Section 5.1 introduces a *fast calculation* method and investigates the possibility of avoiding the repeated run of the neutron-transport simulation to save on computational resources.

Based on the fast calculation results, Section 5.2 introduces the *Generic Irradiation Database* method and discusses the results of two applications. The first application verifies the generic irradiation database method for the simple demonstration exercise introduced in Section 3.3.2. The second application is the verification of the method for an ATR experiment.

5.1 Fast Calculation

This method solves the neutron-transport problem only once for a specific experiment material to obtain the geometry-dependent parameters. Then, these parameters are fixed and become the input to the rest of the delayed heating calculation workflow regardless of the experiment composition. The underlying assumption is that the experiment volume is small enough to perturb the neutron flux in the reactor, and that for all experiment materials the total neutron flux is approximately the same. As shown in Figure 5.1, the total neutron flux in experiments of different chemical elements is bounded between 3.4 to $6.0 \times 10^{14} n/cm^2/s$. These results show that the total neutron flux in the experiment does not vary strongly and always remains in the same order of magnitude. As the first step of the delayed heating calculation obtains the neutron flux, fixing its value within the range shown in the figure may allow for a sufficiently accurate delayed heating calculation.

As the neutron transport simulation must be run at least once to fix the geometry-dependent parameters, the question that remains unanswered is what experiment material should be chosen for the simulation. Two materials are studied: one that considerably perturbs the flux, which is 20%-enriched uranium, and one that barely perturbs the flux, which is carbon. Although not shown here, when utilizing the geometry-dependent parameters from the neutron transport simulation for an experiment of uranium, the delayed heating results are not accurate. That is because perturbing the reactor flux to obtain the fixed geometry-dependent parameters is not desirable as originally expected. When perturbing the flux, as the reactor power remains

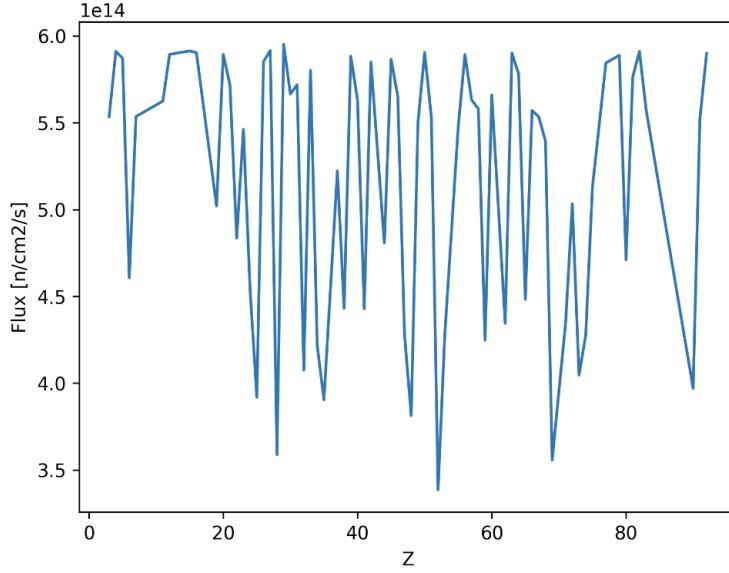
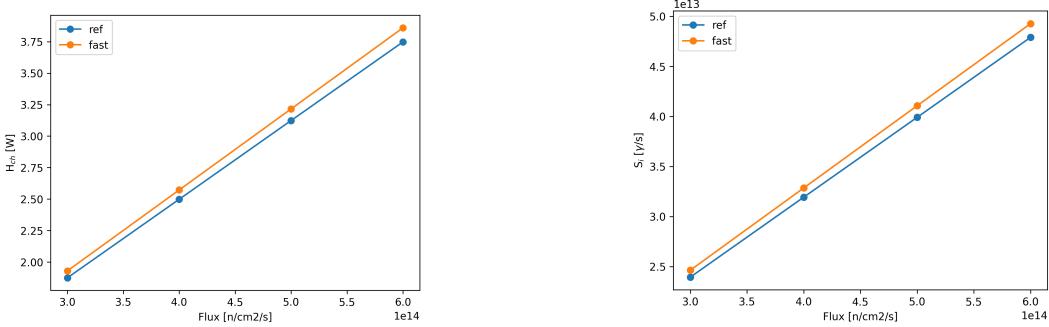


Figure 5.1: Total neutron flux vs. atomic number of the experiment material.

constant, the flux in the experiment increases, while the flux in the fuel assemblies decreases. Hence, the fuel assemblies are subject to a lower depletion, and consequently they produce a lower delayed gamma emission rate. As a result, choosing carbon as the experiment material is more favorable because it does not considerably perturb the flux.

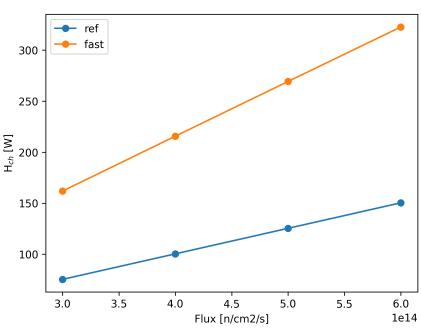
To better understand the effects of the geometry-dependent parameters on the delayed heating, this section presents a sensitivity analysis on some key magnitudes in the calculation workflow, i.e., the energy deposited by charged particles (H_{ch}) and the photon emission rate (S_i). The calculations use the demonstration exercise from Section 3.3.2. The reference calculation is based on the delayed heating calculation for an experiment of arbitrary material, while the fast calculation uses the values from the neutron-transport simulation for the experiment of carbon. Figures 5.2 to 5.5 show a sensitivity analysis of the H_{ch} and S_i for different total flux levels. These results show that the fast calculation overestimates the results for all cases. The overestimation for the experiment of aluminum is minimal, while the overestimation for other cases, such as the experiment of enriched uranium, is relatively large.



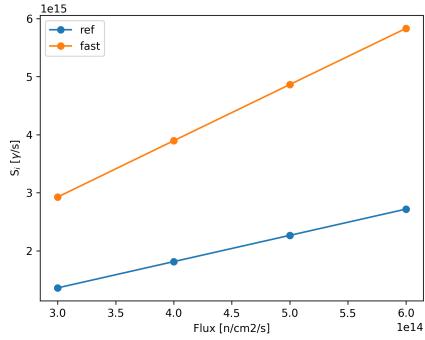
(a) Energy deposited by charged particles.

(b) Photon emission rate.

Figure 5.2: Sensitivity analysis for aluminum.

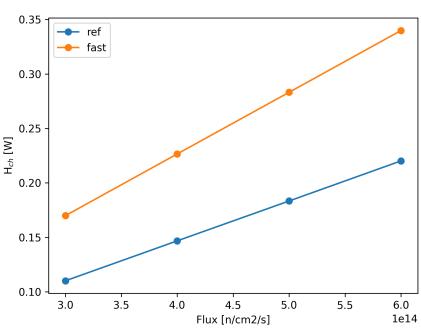


(a) Energy deposited by charged particles.

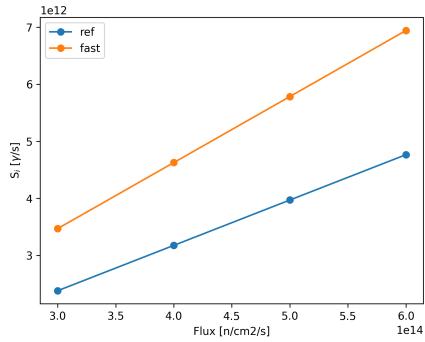


(b) Photon emission rate.

Figure 5.3: Sensitivity analysis for manganese.

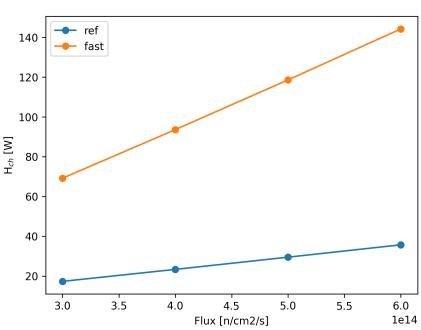


(a) Energy deposited by charged particles.

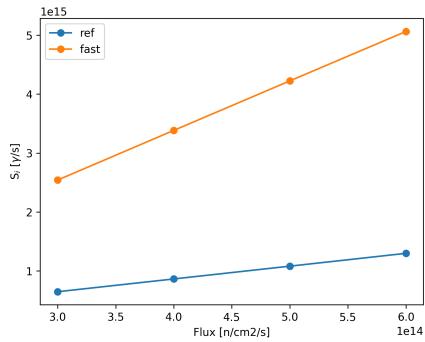


(b) Photon emission rate.

Figure 5.4: Sensitivity analysis for zirconium.



(a) Energy deposited by charged particles.



(b) Photon emission rate.

Figure 5.5: Sensitivity analysis for tungsten.

Table 5.1 compares the results using the fast calculation and the reference results obtained using the delayed heating calculation workflow. The fast calculation method overestimates the delayed heating. And while for some materials, such as aluminum and zirconium, the overestimation is minimal, for other materials, such as manganese, tungsten, and uranium, the overestimation is considerably larger. The last step in the comparison is the delayed heating over time, shown in Figures 5.7 to 5.9. The delayed heating results confirm the observations in the sensitivity analysis, shown above. The fast calculation yields accurate results for

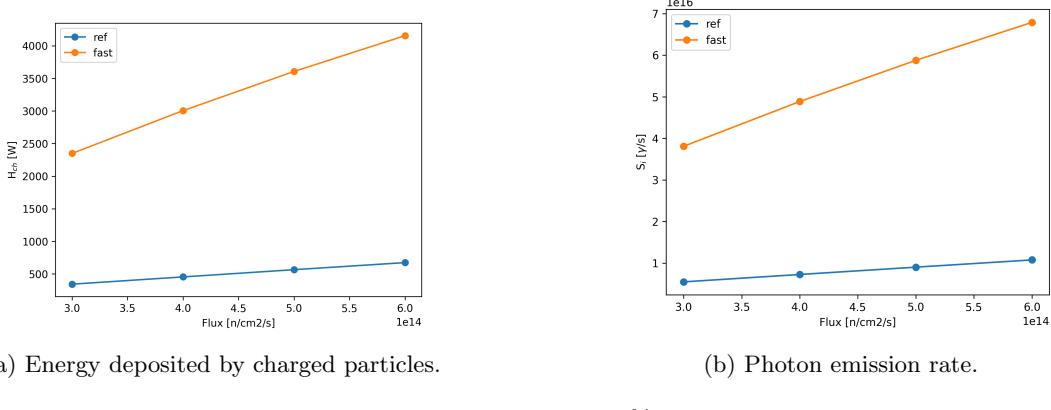


Figure 5.6: Sensitivity analysis for 20%-enriched uranium.

aluminum and zirconium, while it overestimates the delayed heating in experiments of manganese, tungsten, and uranium.

Table 5.1: Comparison of total heat deposited in experiments of selected materials for the fast calculation and the reference results of the delayed heating calculation workflow.

Material	Atomic Number	Fast [W]	Reference [W]	Rel. Diff. [%]
Aluminum	13	18.35	18.20	0.87
Manganese	25	375.1	169.37	121.49
Zirconium	40	38.36	38.24	0.31
Tungsten	74	356.19	164.46	116.58
Uranium ($\varepsilon = 20\%$)	92	5880.80	994.52	491.31

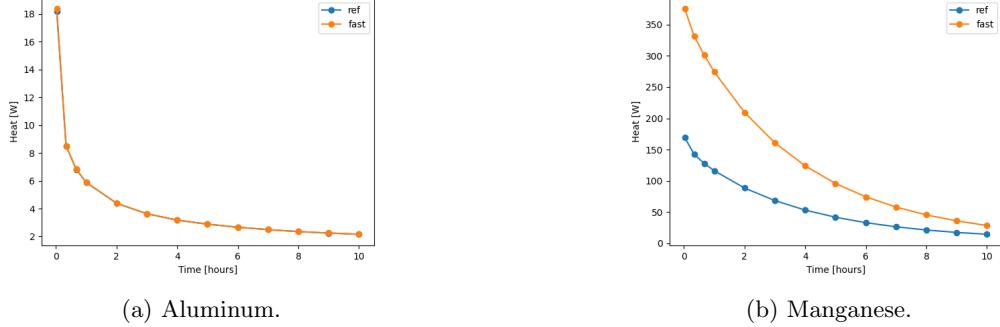
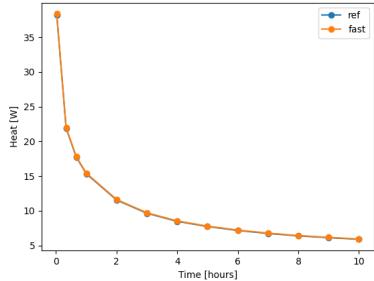
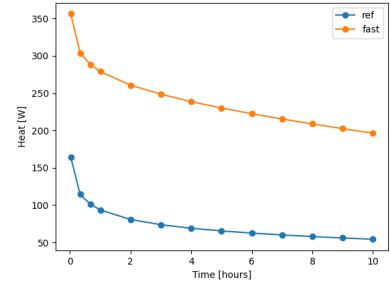


Figure 5.7: Comparison of the delayed heating over time for the reference workflow and the fast calculation for experiments of selected materials.

These results show that the fast calculation is accurate for some cases and excessively conservative for others. While a conservative approach may be desirable for some cases, this thesis investigates more accurate calculation methods. Consequently, this section shows that the fast calculation is not suitable for this work's objectives, concluding that the neutron-transport simulations cannot be avoided. The next section introduces a different approach and investigates its suitability.



(a) Zirconium.



(b) Tungsten.

Figure 5.8: Comparison of the delayed heating over time for the reference workflow and the fast calculation for experiments of selected materials.

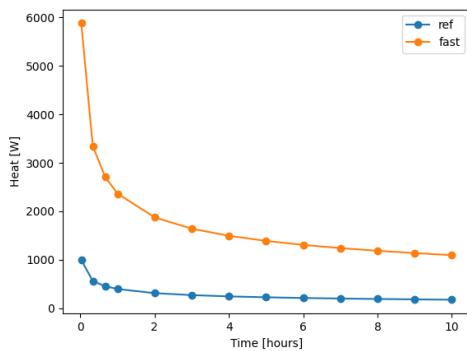


Figure 5.9: Comparison of the delayed heating over time for the reference workflow and the fast calculation for an experiment of 20%-enriched uranium.

5.2 Generic Material Irradiation Database

This section proposes the creation of a generic material irradiation database to decrease the burden of the delayed heating calculation in research reactor experiments. The database is created by utilizing the calculation workflow defined in Chapter 3 repeatedly to obtain the delayed heating in experiments composed of pure chemical elements. Nevertheless, an experiment is usually made of a mix of materials. Hence, this work establishes a method using the database to calculate the delayed heating in experiments of any material composition. This method allows obtaining results based on pre-computed values instead of re-doing the calculations.

The following formula calculates the delayed heating in an experiment composed of several materials

$$H_{j,M} = \sum_i H_{j,i}(\rho_i) \quad (5.1)$$

$$H_{j,i}(\rho_i) = \rho_i f(\rho_i) \quad (5.2)$$

$$f_{non-fiss}(\rho_i) = H_{j,i}^o / \rho_i^o \quad (5.3)$$

$$f_{fiss}(\rho_i) = H_{j,i}^k / \rho_i^k \quad (5.4)$$

where $H_{j,M}$ is the heating component j of the mix and ρ is the density. The heating component j could be either the charged particles, transported gammas, or total contributions. $H_{j,i}$ and ρ_i correspond to the heating values and density of material i in the mix. The formula contemplates two cases, fissile and non-fissile materials. The calculation of $H_{j,i}(\rho_i)$ requires a functional form for $f(\rho_i)$, being constant for non-fissile (Eq. 5.3) and density-dependent for fissile materials (Eq. 5.4). $H_{j,i}^o$ corresponds to the heating values for an experiment made of material i only with a density value equal to its theoretical density ρ_i^o . $H_{j,i}^k$ corresponds to the heating values for an experiment made of material i only with a density value equal to ρ_i^k . As f_{fiss} is density-dependent, it requires multiple samples of $H_{j,i}^k / \rho_i^k$ for interpolating between adjacent values.

5.2.1 Demonstration exercise

The exercise configuration and dimensions can be found in Section 3.3.2. The experiment is irradiated for a period of 50 days with the reactor operating at a constant power of 5 MWth. After that period, the reactor shuts down, and the heat deposited in the experiment is calculated immediately after shutdown. To avoid the prompt-gamma contributions (emitted within 5×10^{-8} seconds [40]), the first step is taken 1 second after shutdown. The reactor regions considered contributing to the experiment heating are the fuel elements and the experiment itself, while the activation of the moderator is neglected.

The following analyses focus on the delayed heating calculation for a mix of materials. The first analysis considers an experiment of an arbitrary alloy, in this case, bronze, which is an alloy of copper and tin. Bronze has a density of 8.73 g/cm³, and the weight fractions are 88% and 12% for copper and tin, respectively. Table 5.2 summarizes the results. The results from the irradiation database method are based on the delayed heating of samples of pure copper (8.96 g/cm³) and pure tin (5.75 g/cm³). The calculated values coincide with the reference values. The relative difference for H_{ch} is the largest, but it is still smaller than 5%.

The following analysis demonstrates why a constant function representing $f_{non-fiss}$ yields a good approximation. Figure 5.10 displays $H_{j,i}/\rho_i$ as a function of ρ for experiments of copper and tin. The densities considered for copper are 2.25, 4.5, 6.75, and 8.96 g/cm³ and for tin 1.5, 3.0, 4.5, and 5.75 g/cm³.

Table 5.2: Comparison of delayed heating for the irradiation database method and reference calculation for an experiment of bronze.

	Irradiation database method [W]	Reference calculation [W]	Rel. Diff [%]
H_{ch}	15.54	16.24	-4.31
$H_{\gamma,Tr}$	49.75	50.17	-0.84
H_T	65.30	66.41	-1.67

These values represent approximately 25, 50, 75, and 100% of the theoretical density for each material. H_{ch} represents less than 30% and less than 2% of H_T for all densities for copper and tin, respectively. For copper, the slope formed between the first and last values are -24%, -9%, and -13% for H_{ch} , $H_{\gamma,Tr}$, and H_T , respectively. For tin, the slope formed between the first and last values are -14%, -16%, and -16% for H_{ch} , $H_{\gamma,Tr}$, and H_T , respectively. The methodology introduced in this paper considers $H_{j,i}/\rho_i$ to be constant for non-fissile materials (Eq. 5.3). Although these curves have negative slopes, their magnitudes are relatively small, and as shown in Table 5.2, a constant function still yields satisfactory results.

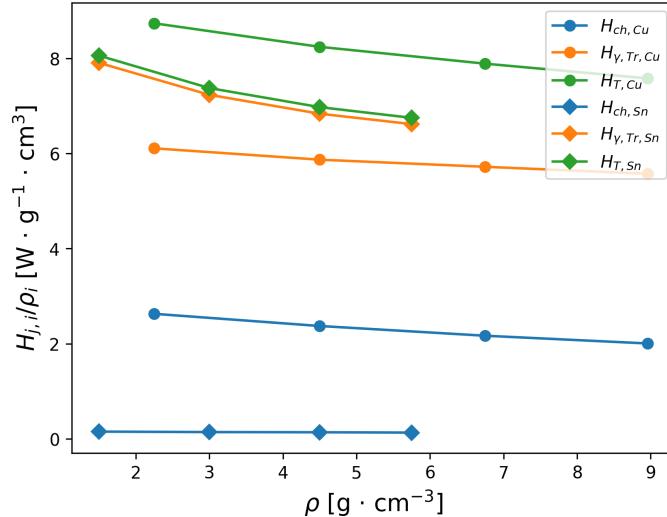


Figure 5.10: H_j/ρ vs. ρ for different experiment materials.

The following exercise considers an experiment including a fissile material, uranium nitride (UN). This type of fuel has been extensively studied as one candidate accident tolerant fuel, due to its high thermal conductivity and high fissile element density [103]. Table 5.3 summarizes the results for an experiment of UN. The exercise adopts a 4% enrichment for uranium. The results from the irradiation database correspond to the calculated value for uranium using a linear interpolation, while the contribution from nitrogen is considered negligible. H_T is calculated by adding together H_{ch} and $H_{\gamma,Tr}$. The relative difference of the results is less than 3%, which confirms the choice of a linear interpolation to represent $f_{fiss}(\rho_i)$.

The following analysis demonstrates why the inclusion of a fissile material into the mix requires multiple values to obtain f_{fiss} . The exercise adopts the following uranium densities for the calculations: 1, 7, 13, and 19 g/cm³. The UN has a density of 13.5 g/cm³, and the atomic fraction is 50% for both elements. Figure 5.11 displays $H_{j,i}/\rho_i$ as a function of ρ for the experiment of uranium. Only the values for the uranium are included because the delayed heating in a sample of pure nitrogen is negligible. $H_{\gamma,Tr}/\rho$ is almost constant,

Table 5.3: Comparison of delayed heating for the irradiation database method and reference calculation for an experiment of uranium nitride.

	Irradiation database method [W]	Reference calculation [W]	Rel. Diff [%]
H_{ch}	273.1	267.0	2.3
$H_{\gamma,Tr}$	233.7	236.0	-0.9
H_T	506.8	503.0	0.8

while H_{ch}/ρ decreases as the density increases. Then, equation 5.4 calculates the delayed heating in the UN.

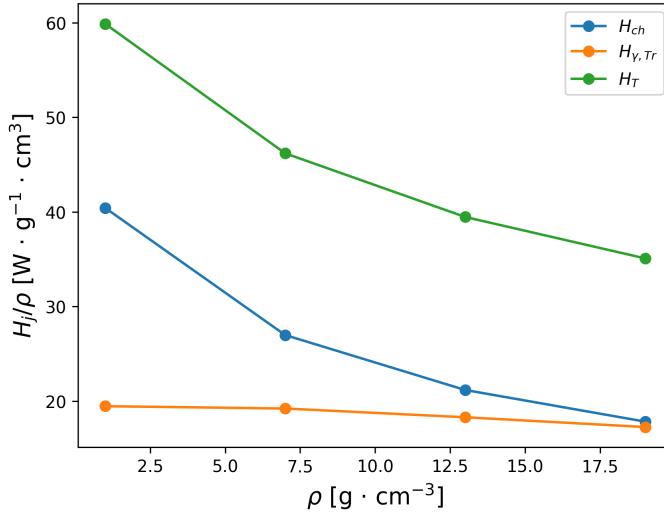


Figure 5.11: H_j/ρ vs. ρ for an experiment of 4%-enriched uranium.

The following exercise considers three experiments of zirconium alloys: Zircaloy-2, Zircaloy-4, and ZrNb. Zirconium is a metal with excellent corrosion resistance, good mechanical properties, and very low thermal neutron cross-section [104]. Zircaloy-2 is composed of Zr-1.5Sn-0.15Fe-0.1Cr-0.05Ni and its primary use is as fuel cladding in BWRs and as calandria tubes in CANDU reactors. Zircaloy-4 is composed of Zr-1.5Sn-0.2Fe-0.1Cr, which is a similar composition to Zircaloy-2 but removes the nickel and increases the iron weight for less hydrogen uptake, and it is used as fuel cladding in PWR and CANDU reactors. ZrNb is composed of Zr-2.5Nb, wherein the niobium increases the alloy's strength, and it is used in pressure tubes in CANDU reactors. Table 5.4 displays the results for the three zirconium alloys. The results from the irradiation database method are based on the delayed heating of samples of pure zirconium (7.13 g/cm³), pure tin (5.75 g/cm³), pure iron (7.87 g/cm³), pure chromium (7.18 g/cm³), pure nickel (8.90 g/cm³), and pure niobium (8.57 g/cm³). The reference calculation considers an experiment density of 6.56 g/cm³, 6.56 g/cm³, and 6.44 g/cm³, for Zircaloy-2, Zircaloy-4, and ZrNb, respectively. The calculated values show good agreement with respect to the reference values. The relative difference for H_{ch} is the largest for Zircaloy-2, while all the other relative differences are below 3%.

The following exercise considers three experiments containing uranium. The first two are uranium-based alloys: U-9Mo and U-10Nb. It is common for fissionable materials to exhibit anisotropic growth during neutron irradiation. However, alloying uranium with elements such as molybdenum and niobium, causes the alloy to retain the metastable gamma phase during irradiation, and the anisotropic growth can be eliminated

Table 5.4: Comparison of delayed heating for the irradiation database method and reference calculation for multiple experiments.

	Irradiation Database method [W]	Reference calculation [W]	Rel. Diff [%]
Zircaloy-2			
H_{ch}	0.19	0.18	5.84
$H_{\gamma,Tr}$	38.42	38.39	0.07
H_T	38.61	38.57	0.10
Zircaloy-4			
H_{ch}	0.19	0.19	1.48
$H_{\gamma,Tr}$	38.42	38.40	0.06
H_T	38.61	38.58	0.06
ZrNb			
H_{ch}	0.19	0.19	2.49
$H_{\gamma,Tr}$	37.66	37.72	-0.15
H_T	37.85	37.91	-0.14
U(nat)-Mo			
H_{ch}	135.38	132.31	2.32
$H_{\gamma,Tr}$	196.06	191.66	2.30
H_T	331.44	323.97	2.31
U(nat)-Nb			
H_{ch}	129.13	127.35	1.39
$H_{\gamma,Tr}$	187.82	183.92	2.12
H_T	316.94	311.27	1.82
U(20%)3Si2Al			
H_{ch}	254.96	247.48	3.02
$H_{\gamma,Tr}$	95.75	108.88	-12.05
H_T	350.71	356.35	-1.58
Tungsten alloy			
H_{ch}	21.35	22.52	-5.22
$H_{\gamma,Tr}$	129.20	132.33	-2.37
H_T	150.54	154.85	-2.78

[105]. The third experiment is composed of $\text{U}_3\text{Si}_2/\text{Al}$, which is a typical material utilized in material-testing reactor (MTR)-fuel plates. Table 5.4 displays the results for the experiments containing uranium.

The calculations consider a natural uranium composition for the experiments of UMo and UNb and a 20%-enriched uranium for the experiment of $\text{U}_3\text{Si}_2/\text{Al}$. The results from the irradiation database method are based on the delayed heating of samples of pure uranium, pure molybdenum, pure niobium, pure silicon, and pure aluminum. The exercise adopts the following uranium densities for the calculations: 1, 7, 13, and 19 g/cm³. For molybdenum, niobium, silicon, and aluminum, the exercise uses densities of 10.22, 8.57, 2.33, and 2.70 g/cm³, respectively. The reference calculation considers an experiment density of 17.35, 16.51, and 5.76 g/cm³, for UMo, UNb, and $\text{U}_3\text{Si}_2/\text{Al}$, respectively. Most of the calculated values show good agreement with respect to the reference values, showing relative differences below 3%, except H_{ch} and $H_{\gamma,Tr}$ for $\text{U}_3\text{Si}_2/\text{Al}$. The latter shows a relative difference of approximately 12%, but as the charged particles have a larger contribution to the heating, the total heat is within 2% of the reference value.

The following exercise considers an experiment of tungsten-nickel-iron alloy. Tungsten alloys are good candidates for fusion applications, due to tungsten's high melting point. However, tungsten alone can be very brittle, and including small amounts of other metals, such as nickel and iron, creates a tougher alloy while retaining the high melting temperature [106]. The alloy composition considered in this work is W-3.5Ni-1.5Fe. Table 5.4 displays the results for this alloy. The calculated values show good agreement with respect to the reference values. The largest relative difference is for H_{ch} , being around 5%, while the rest of the relative differences are within 3%.

5.2.2 ATR experiment

This exercise focuses on the delayed heating in the A1 experiment position, which is an inner A-hole that is hosted by the neck shim housing. This work considers for the calculations a total operating power of 110 MWth that is equally distributed among the 5 lobes. The irradiation time is 30 days, and the decay is recorded in eight non-uniformly distributed steps up to 12 hours after shutdown. As the experiment position is in the core region between the fuel assemblies, the calculations assume that the fission products generate a much higher gamma source intensity than the activation products in the neighboring cells. Hence, the contribution of the reactor components surrounding the experiment is neglected. The only considered contributors to the heating are all the fuel plates and the experiment itself. Section 3.3.3 provides more details of the model and modeling assumptions.

The generic irradiation database is based on the delayed heating calculation for the middle of the equilibrium cycle (MOC) values, which is a conservative approach [40]. Each ATR power cycle typically lasts for 40 to 60 days [3], and this work considers an irradiation time of 30 days.

Figure 5.12 displays $H_j(Z)$ and $H_j(\rho)$ for all the chemical elements included in the irradiation database. $H_j(M)$ was plotted as well but the plot is not presented here. As the relationship between Z and M is linear, the plots are very similar, and $H_j(M)$ does not convey any further information. $H_{\gamma,Tr}(Z)$ increases with an increasing Z , but not monotonically. For example, bromine ($Z=35$), cesium ($Z=55$), and mercury ($Z=80$) have lower heating values than the subsequent materials. $H_{ch}(Z)$ does not present a clear trend with respect to Z . The general trend for $H_j(\rho)$ is to increase with an increasing ρ . This trend is clearer for the transported photon component than for the charged particle component. $H_{ch}(\rho)$ is close to zero for densities smaller than 5 g/cm³, has an increasing trend after 15 g/cm³, and oscillates between 0 and 2 kW for the intermediate values.

Figure 5.13 presents the experiment heat production over time in an experiment of Zircaloy-2. The

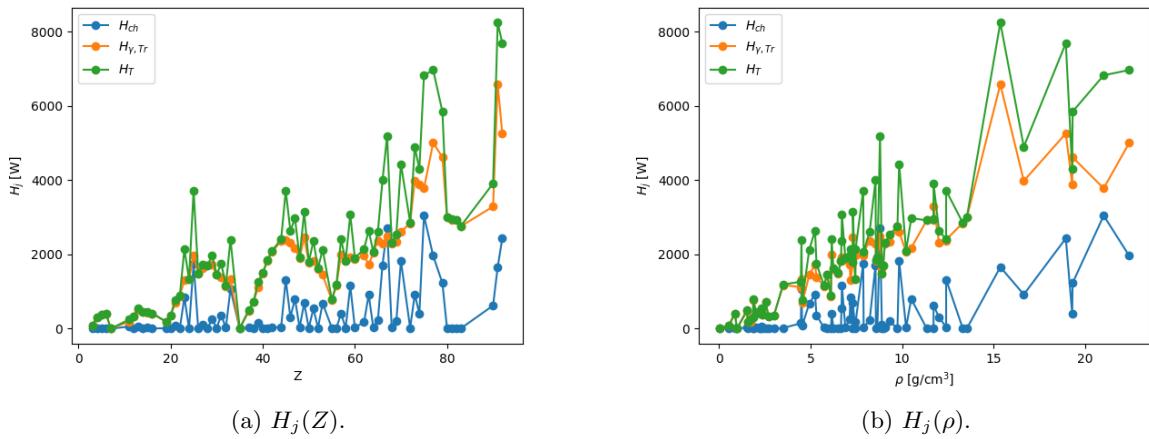


Figure 5.12: Generic irradiation database values for the ATR A1 experiment position.

reference curve is for an experiment of Zircaloy-2 with a density of 6.56 g/cm³ and the following composition Zr-1.5Sn-0.15Fe-0.1Cr-0.05Ni. The calculated curve corresponds to the results from the database method and are based on the delayed heating of samples of pure zirconium (7.13 g/cm³), pure tin (5.75 g/cm³), pure iron (7.87 g/cm³), pure chromium (7.18 g/cm³), and pure nickel (8.90 g/cm³). The delayed heating is 2.36 kW 1 second after shutdown, which is less than 0.01 % of the total power during operation, and it decays below 500 W within 2 hours. The calculated curve shows good agreement with the reference curve, having a relative difference smaller than 1%.

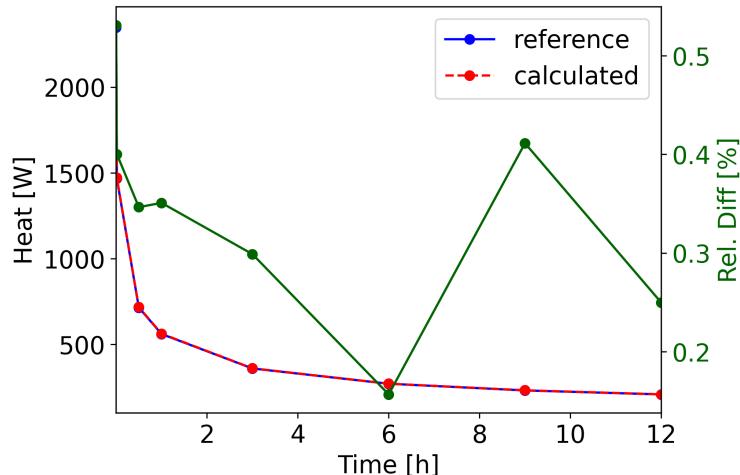


Figure 5.13: Heat production over time in an experiment of Zircaloy-2.

The following exercise considers an experiment of iron-chromium-aluminum (FeCrAl) alloy. This type of alloy exhibits enhanced oxidation resistance in elevated temperature steam environments when compared to Zr-based alloys, making it a good cladding candidate for accident tolerant fuel in LWRs [107]. Additionally, it is stress corrosion cracking resistant and irradiation induced swelling resistant. The typical content ranges from 10 to 22% for chromium, and from 4 to 8% for aluminum. Figure 5.14 presents the experiment heat production over time in the experiment of FeCrAl. The reference curve is for an experiment of FeCrAl with

a density of 8 g/cm^3 and the following composition Fe-22Cr-8Al. The calculated curve corresponds to the results from the irradiation database method and are based on the delayed heating of samples of pure iron (7.87 g/cm^3), pure chromium (7.18 g/cm^3), and pure aluminum (2.70 g/cm^3). The delayed heating is 2.38 kW 1 second after shutdown, which is less than 0.01% of the total power during operation, and it decays below 500 W within 2 hours. The calculated curve shows good agreement with the reference curve, having a relative difference smaller than 1% .

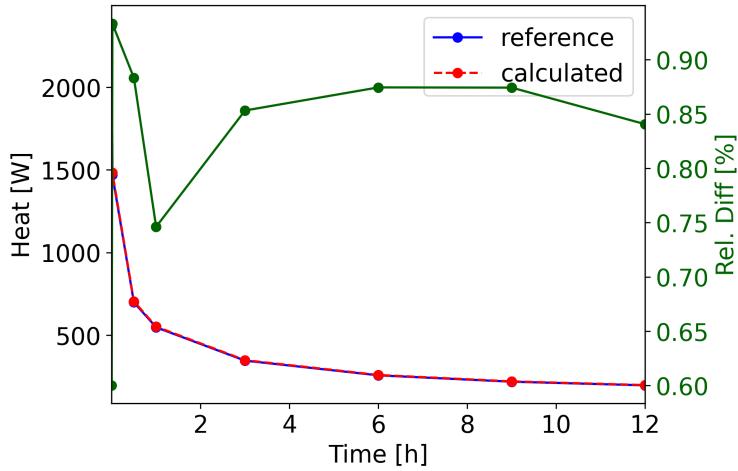


Figure 5.14: Heat production over time in an experiment of FeCrAl.

The following exercise considers an experiment of UMo with natural uranium. Figure 5.15 presents the experiment heat production over time in the experiment of UMo. The reference curve is for an experiment of UMo with a density of 17.35 g/cm^3 . The calculated curve corresponds to the results from the database method and are based on the delayed heating of samples of pure uranium and pure molybdenum. The exercise adopts the following uranium densities for the calculations: $1, 7, 13$, and 19 g/cm^3 , and a molybdenum density of 10.22 g/cm^3 . The delayed heating is 11.29 kW 1 second after shutdown, which is 0.01% of the total power during operation, and it decays below 2 kW within 3 hours. The calculated curve shows good agreement with the reference curve, having a relative difference of approximately 5% .

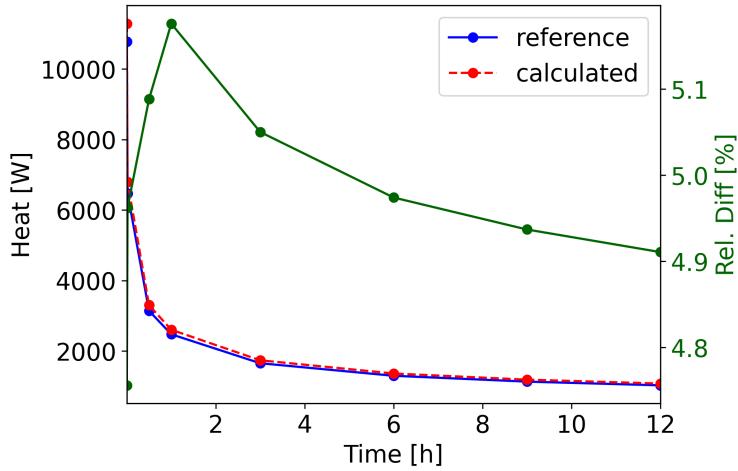


Figure 5.15: Heat production over time in an experiment of UMo.

5.3 Conclusions

Safety analyses in research reactors require the estimation of heat deposited in experiments and reactor structures after shutdown. Research reactors enable the irradiation of a wide variety of unique experiments, requiring the development of safety analyses on a case-by-case basis, which often proves to be effort- and time-consuming. This chapter studies two methods using previous knowledge to expedite the process.

First, the chapter investigates the feasibility of reducing the calculation burden by solving the neutron-transport problem only once. This fast method utilizes the geometry-dependent parameters from the neutron-transport problem for an experiment of carbon, which is a material that does not considerably perturb the neutron flux. Then, the depletion calculation utilizes these parameters to obtain the charged particle heating and photon source intensity. Some cases are studied for selected materials and the results compared to the delayed heating calculation results. The results show that the fast calculation is accurate for some cases while excessively conservative for others. Additionally, the study concludes that this approach is not suitable to this thesis' objectives and that neutron-transport simulations cannot be avoided.

Second, this chapter also describes the irradiation database method, which relies on the delayed heating calculation to obtain the heat produced in experiments of individual chemical elements and through their combination calculate the heat produced in experiments of any material composition. Finally, this chapter demonstrates the generic irradiation database method by presenting two exercises. These exercises include the delayed heating calculation of experiments for a demonstration exercise and ATR. The results verify the method by showing a wide variety of combinations of materials that could integrate a research reactor experiment. Additionally, for the case of the ATR exercise, the heat production over time is shown for several experiments. Overall, the results exhibit good agreement between the calculated and reference values.

Chapter 6

Thermal-Fluids and Machine Learning

This chapter examines the applicability of machine learning to capture the thermal behavior of an experiment during the transient. Data-based modeling, which relies on machine learning, allows to extract features from data generated with the physics-based model and predict the behavior of new data. Hence, a thermal-fluids solver is necessary to determine the thermal behavior of the experiment and generate the required data to train and test the data-based model. Section 6.1 describes the thermal-fluids solvers utilized in this work. Additionally, this work compares the performance of different data-based strategies. Section 6.2 introduces briefly the machine learning algorithms utilized in this work, Section 6.3 introduces a simplified thermal-fluids model, its verification, and it investigates the performance of the different data-based approaches and machine learning algorithms. Section 6.3.2 discusses some necessary changes in the data processing to implement the data-based modeling for an experiment subject to delayed heating. Finally, Section A.2 demonstrates the data-based modeling of an ATR experiment undergoing the accident.

6.1 Thermal-Fluids Solvers

6.1.1 MOOSE

The Multi-physics Object-Oriented Simulation Environment (MOOSE) [108], originally released in 2008, is an open-source Finite Element Method (FEM) framework that supports engineering analysis applications, targeting systems of non-linear partial differential equations. The MOOSE applications define weak forms of the governing equations and modularize the physics expressions into *kernels*. The kernels are C++ classes containing methods for computing the residual and Jacobian contributions of individual pieces of the governing equations. All the software built on the MOOSE framework shares the same Application Programming Interface (API). This feature facilitates a relatively easy coupling between different phenomena. Additionally, the framework and its applications use Message Passing Interface (MPI) for parallel communication and allow deployment on massively parallel cluster-computing platforms.

In this work, MOOSE solves a simplified thermal-fluids model (Section 6.3) to examine the applicability of machine learning to capture the experiment thermal behavior. Three reasons motivated the choice of MOOSE for such a study. First, the author's familiarity with the software made it the natural choice. Second, the MOOSE input sequence consists of only one script, which reduces the burden of handling a large number of files. Third, the MOOSE model uses the Boussinesq approximation to reduce the computational expense of the simulation. The approximation considers that the density-variation effects are only present in the body

force term of the conservation of momentum equations. This approximation reduces the system of equations in the fluid region to three equations for solving the motion and temperature

$$\nabla \cdot \vec{u} = 0 \quad (6.1)$$

$$\rho \left(\frac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \nabla) \vec{u} \right) + \nabla p - \rho \vec{g} + \rho \vec{g} \beta (T - T_{ref}) - \mu \Delta \vec{u} = 0 \quad (6.2)$$

$$\rho c_p \left(\frac{\partial T}{\partial t} + (\vec{u} \cdot \nabla) T \right) - k \Delta T - q''' = 0 \quad (6.3)$$

where \vec{u} is the velocity, ρ is the density, p is the pressure, \vec{g} is the gravity, β is the thermal expansion coefficient, T is the temperature, T_{ref} is the reference temperature, μ is the dynamic viscosity, c_p is the specific heat capacity, and k is the thermal conductivity. MOOSE solves the heat conduction equation to obtain the temperature in the solid regions

$$\rho c_p \frac{\partial T}{\partial t} - k \Delta T - q''' = 0 \quad (6.4)$$

$$q'''(t)[W/m^3] = q_0 e^{-t/\tau} \quad (6.5)$$

where q''' is the heat source which is uniformly distributed across the experiment volume and is defined by a decaying exponential representing the delayed heating.

6.1.2 OpenFOAM

OpenFOAM [109] is a Finite Volume Method (FVM) C++ class library originally released as the Field Operation And Manipulation (FOAM) library in 1989, becoming open source in 2004. It comprises an extensible range of libraries, solvers, and pre- and post-processing tools, for which the developers intended to make the top-level syntax of the code as close as possible to conventional mathematical notation. The result is a C++ class library capable of modeling continuum-mechanics phenomena, such as compressible and incompressible flow, multiphase flow, including various turbulence modeling techniques.

Although it primarily targets CFD problems, it solves other continuum-mechanics problems as well, such as structural mechanics [109] or molecular dynamics [110]. Moreover, it supports parallelization via domain decomposition, in which each available processor runs the code in each mesh subdomain.

Further inspection of the thermal-fluids model revealed that even when MOOSE uses the Boussinesq approximation and OpenFOAM does not, OpenFOAM simulations are considerably faster. This reason motivated the choice of OpenFOAM as the main solver for more detailed models. Furthermore, OpenFOAM has several turbulence models readily available and is capable of solving the density, motion, and temperature of the fluid.

OpenFOAM calculates the capsule temperature during the channel draining event by solving the mass, momentum, and energy conservation equations (Eq. 6.6-6.8) along with the equation of state (Eq. 6.9) in the

air regions, and the heat conduction equation (Eq. 6.11) to solve the temperature in the solid regions [111]

$$\frac{\partial}{\partial t}\rho + \nabla \cdot (\rho \bar{u}) = 0 \quad (6.6)$$

$$\left(\frac{\partial}{\partial t}(\rho \bar{u}) + \nabla \cdot (\rho \bar{u} \bar{u}) \right) + \nabla p - \rho \bar{g} - \mu \Delta \bar{u} = 0 \quad (6.7)$$

$$\left(\frac{\partial}{\partial t}(\rho h) + \nabla \cdot (\rho \bar{u} h) \right) + \frac{\partial}{\partial t} \left(\rho \frac{|\bar{u}|^2}{2} \right) + \nabla \cdot \left(\rho \bar{u} \frac{|\bar{u}|^2}{2} \right) - \frac{\partial}{\partial t} p - \alpha \Delta h - \rho \bar{g} \cdot \bar{u} = 0 \quad (6.8)$$

$$p = \rho R T \quad (6.9)$$

$$h = c_p T \quad (6.10)$$

$$\rho c_p \frac{\partial}{\partial t} T - k \Delta T - q''' = 0 \quad (6.11)$$

where ρ is the density, \bar{u} is the velocity, p is the pressure, \bar{g} is the gravity, μ is the dynamic viscosity, h is the enthalpy, α is the thermal diffusivity, R is the ideal gas constant, c_p is the specific heat capacity, T is the temperature, k is the thermal conductivity, and q''' is the volumetric heat source provided by the delayed heating calculation workflow.

The calculations use OpenFOAM v2012 (December 2020 release) and the application *chtMultiRegionFoam*. This application solves transient fluid flow and solid heat conduction with conjugate heat transfer between regions. The solver follows a segregated solution strategy, wherein the equations for each characteristic variable are solved sequentially and their solution inserted in the subsequent equations. The coupling between fluid and solid follows a similar approach, wherein the fluid equations are solved using the solid temperatures from the previous iteration as the boundary condition. Then, the equation for the solid is solved using the temperature of the fluid of the previous iteration as the boundary condition. And this process is repeated until convergence.

6.2 Machine Learning

This work uses several classification algorithms to determine if the capsule encasing the experiment melts in case of a channel draining event. Classification is the process of recognizing certain patterns in a dataset and grouping the samples into categories based on those patterns. Machine learning classification algorithms use input training data to determine such categories and then predict the likelihood that subsequent data points will fall into them. This work utilizes the following classification techniques: SVM, Decision Tree (DT), Random Forest (RF), KNN, LogR, and FNNs.

Additionally, this work utilizes several methods to predict the time evolution of the capsule temperature. These methods rely on the following algorithms: DTR, RFR, FNNs, and LSTMs. The implementation of the FNNs and LSTMs utilizes *Keras* [112] and the rest of the models utilize *scikit-learn* [113]. This section presents a brief introduction to the different machine learning algorithms.

6.2.1 Support Vector Machine

This algorithm searches for a hyperplane that separates the dataset, as shown in Figure 6.1. The support vectors are a small subset of the training samples that are closest to this hyperplane. The method determines the hyperplane that maximizes the distance or margin to the support vectors [113].

Equation 6.12 defines the hyperplane, and Equation 6.13 defines the hyperplanes of the support vectors,

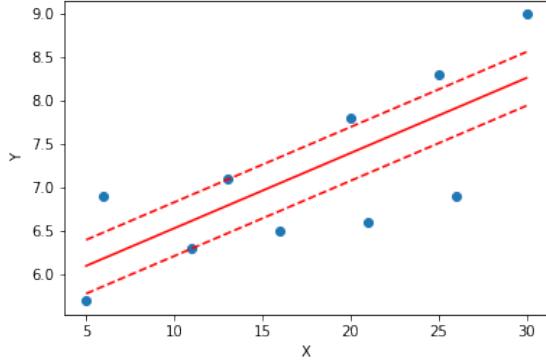


Figure 6.1: Visualization of SVM classification.

where $\phi(x)$ is the transformation from feature to kernel space that allows for non-linear solutions. The margin between hyperplanes is equal to $1/\|w\|$, and then maximizing the margin is equivalent to minimizing $\|w\|$. The optimization problem is solved by minimizing $J(w)$ in Equation 6.14:

$$w \cdot \phi(x_i) + b = 0 \quad (6.12)$$

$$w \cdot \phi(x_i) + b = \pm 1 \quad (6.13)$$

$$J(w) = \frac{1}{2} \|w\|^2 + C \sum_i \xi_i \quad (6.14)$$

such that

$$y_i(w \cdot \phi(x_i) + b) \geq 1 - \xi_i \quad (6.15)$$

$$\xi_i \geq 0 \quad (6.16)$$

where C is the regularization factor, which gives more weight to minimizing the flatness or the error, ξ_i is the slack variable, which guards against outliers. Some problems are not perfectly separable with a hyperplane. Equation 6.14 allows some samples to be at a ξ_i distance from their correct boundary. A larger regularization parameter penalizes the approximation error, leading to its decrease during the optimization.

6.2.2 Decision Tree

The algorithm divides the predictor space into simple regions. The set of splitting rules segmenting the predictor space can be visualized as an inverted tree, as shown in Figure 6.2. Its branches connect internal nodes where a test determines in which sub-branch to continue. The end of the tree contains the leaf nodes which make the final predictions. The leaf nodes assign the most commonly occurring class of the training observations to new observations falling into those regions.

DT uses a recursive binary split to grow the tree by minimizing an impurity function, which in this case is the Gini index [113], defined as

$$G = \sum_k p_{mk}(1 - p_{mk}) \quad (6.17)$$

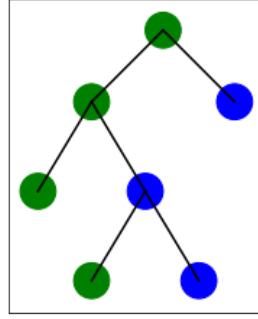


Figure 6.2: Visualization of a decision tree.

where p_{mk} is the proportion of class k observations in the region node m . A low Gini index indicates that a node contains predominantly observations from a single class. Other measures of impurity include entropy and misclassification. The package *scikit-learn* uses the Gini index by default.

The algorithm for DTR is slightly different. The tree can be visualized as a piecewise constant approximation. For every new observation falling into the same space subregion, the leaf nodes assign the same prediction value, which is the average value of the training observations. The training algorithm minimizes the Mean Squared Error (MSE) [113]

$$MSE = \frac{1}{n_m} \sum_{y \in Q_m} (y - \bar{y}_m)^2 \quad (6.18)$$

$$\bar{y}_m = \frac{1}{n_m} \sum_{y \in Q_m} y \quad (6.19)$$

where \bar{y}_m is the learned mean value of the node. Q_m represents the data at node m with n_m samples.

6.2.3 Random Forest

This method improves the performance of a single tree, as individual DTs may exhibit high variance. This method reduces the overfitting and increases its precision. The algorithm consists of a collection of decision trees. It is based on the bootstrap aggregating method or *bagging* [114], which chooses the training data for each decision tree at random with replacement, fits the DTs, and then combines their results by voting. The purpose of the randomness is to decrease the variance of the forest and decouple the prediction error [113].

The algorithm for RFR is very similar, but instead of combining the different DT results by voting, the result is an average of the predictions of the different trees.

6.2.4 K-Nearest Neighbors

Neighbors-based classification does not create a model, but instead it stores instances of the training data [113]. Then, the algorithm classifies new samples based on the stored instances. The KNNs are the K training observations closest in distance to a new observation. The Euclidean distance is the most common choice for measuring the distance between observations. When classifying a new observation, the method assigns

the data class which has the most representatives within the nearest neighbors of the point. KNN is often successful in classification situations where the decision boundary is very irregular.

Figure 6.3 displays an example of KNN classification, where the green dot is the sample to classify. For a K value equal to three, the method uses the three closest samples to the green dot. Two values are blue, and one is orange. As the majority of the samples is blue, the method assigns the blue category to the new sample.

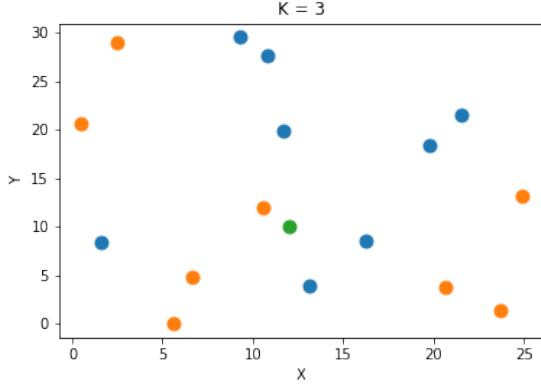


Figure 6.3: Visualization of KNN classification.

6.2.5 Logistic Regression

This type of regression determines the relationship between a binary dependent variable and a set of independent variables. The method utilizes the sigmoid function, shown in Equation 6.20, to determine the likelihood of an observation to fall within each category. The method determines the coefficients of the sigmoid function by minimizing a certain cost function. This work uses the default setting of *scikit-learn* which is to minimize the L_2 -penalized cost function $J(w)$

$$\sigma(x_i) = \frac{1}{1 + e^{-(w \cdot x_i + b)}} \quad (6.20)$$

$$J(w) = \frac{1}{2} \|w\|^2 + C \sum_i \log(\exp(-y_i(w \cdot x_i + b)) + 1) \quad (6.21)$$

where C is the regularization factor. Like in SVM, a larger regularization factor decreases the approximation error during the optimization.

Figure 6.4 displays an example of logistic regression. Most of the orange samples are located in the lower range of the independent variable X while the blue samples are located in the upper range. Then, training the model will assign values to the sigmoid function coefficients such that for new samples in the lower range of X , it will predict the 0 category, while for new samples in the upper range of X , it will predict the 1 category.

6.2.6 Feed-forward Neural Networks

FNNs are deep-learning architectures that learn complex relationships between inputs and outputs by mimicking biological networks of neurons. A network consists of an input and output layer, and M hidden

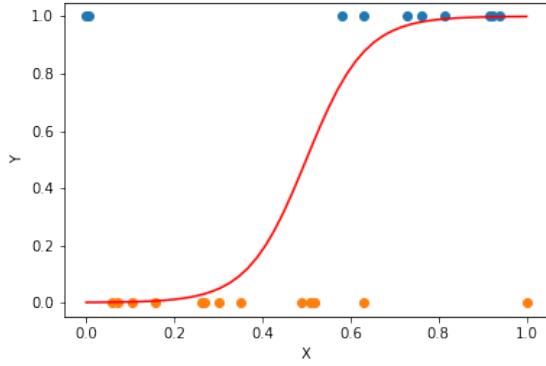


Figure 6.4: Visualization of logistic regression classification.

layers of neurons, as shown in Figure 6.5.

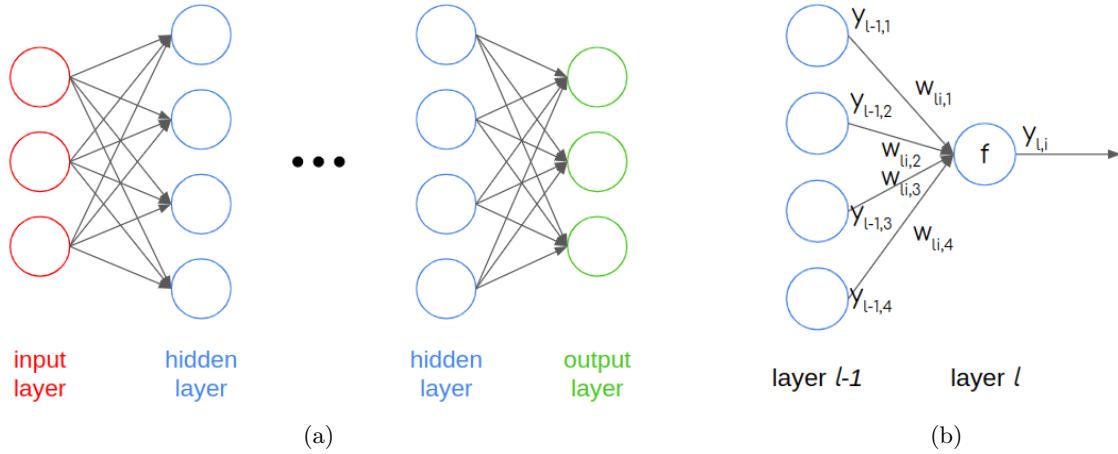


Figure 6.5: Example of an FNN.

A neuron is connected to the neurons of the previous layer by weights. Each neuron adds together the product of the neuron values in the previous layer and the weights connecting them plus a bias. This summation serves as input to the next layer after applying an activation function, as stated in the following equation

$$y_{l,i} = f(w_{l,i}^T y_{l-1} + b_{l,i}) \quad (6.22)$$

$$H = -\frac{1}{N} \sum_i y_i \log(p(y_i)) + (1 - y_i) \log(1 - p(y_i)) \quad (6.23)$$

where $y_{l,i}$ is the output of the node i in layer l , y_{l-1} output values of layer $l-1$, f the activation function, and $w_{l,i}$ and $b_{l,i}$ the weights and biases connecting layer $l-1$ to the node i of layer l .

Training the network sets the values of the connecting weights and biases by minimizing an error metric. For the classification problem, the error metric is the binary cross-entropy, shown in Equation 6.23, where y_i is the label, $p(y_i)$ is the predicted probability of the point to be *true*, and N is the number of samples.

When using the FNN for regression, the algorithm trains the network by minimizing the MSE

$$MSE = \frac{1}{N} \sum_i (Y_i - \hat{Y}_i)^2 \quad (6.24)$$

where \hat{Y}_i is the predicted output of the node i of the output layer of the network, Y_i is the target value, and N is the number of nodes in the output layer times the number of samples.

6.2.7 Long-Short Term Memory Networks

Although RNNs are neural networks that handle inputs and outputs correlated as part of a sequence, they cannot learn long-term dependencies. When training the network, the back-propagated gradients may grow or shrink significantly at each time step. Therefore, the gradients may vanish or explode for time series with long duration. LSTMs are a special kind of RNN that were explicitly designed to avoid the vanishing gradient problem and are able to learn long-term dependencies.

The LSTM cell consists of three parts, better known as gates: the forget gate, the input gate, and the output gate, shown in Figure 6.6. The forget gate determines if the information from the previous timestamp is forgotten, the input gate controls the flow of new information, and the output gate updates the information into the next timestamp. The gates allow the LSTM to add or remove information to the cell state C_t and hidden state h_t . The cell and hidden states are also known as long-term and short-term memory, respectively.

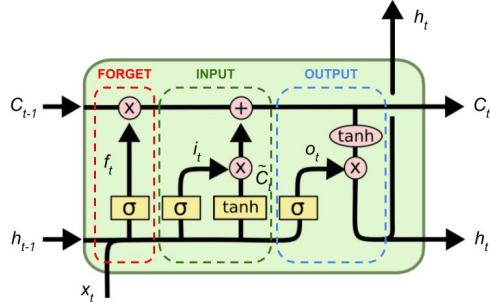


Figure 6.6: LSTM structure.

The following sets of equations are the mathematical definition of the LSTM cell

$$f_t = \sigma(W_f h_{t-1} + U_f x_t + b_f) \quad (6.25)$$

$$i_t = \sigma(W_i h_{t-1} + U_i x_t + b_i) \quad (6.26)$$

$$\tilde{C}_t = \tanh(W_C h_{t-1} + U_C x_t + b_C) \quad (6.27)$$

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \quad (6.28)$$

$$o_t = \sigma(W_o h_{t-1} + U_o x_t + b_o) \quad (6.29)$$

$$h_t = o_t \cdot \tanh(C_t) \quad (6.30)$$

where W and b are the weights and biases for the different layers as in any other type of network. The forget gate output f_t is a number between 0 and 1, wherein a value of 0 means forget everything, and a value of 1, forget nothing. In the input gate, i_t defines which values to update, and \tilde{C}_t comprises the candidate

values that can be added into the cell state. The cell state C_t is calculated as the addition of the values not forgotten from the previous cell state C_{t-1} and the new updated values \tilde{C}_t . Finally, the hidden state h_t is a filtered version of the cell state C_t , and o_t decides what information is included.

Overall, the network can be visualized as a mathematical function whose inputs are the time-dependent features x_t which produce the time-dependent outputs h_t . The training of the network is a bit more cumbersome than for other neural networks. As both the features and outputs are time-dependent, a common training strategy entails converting the time-dependent problem to a static supervised problem, wherein the output of previous time steps act as input to the next time step. The supervised problem separates the data into *look-back* and *look-forward* steps. The look-back steps are the number of steps that the network uses to predict a certain number of look-forward steps at a time. Figure 6.7 displays an example of the conversion process. For the output time-series h , with N_t time steps, the input and output of the supervised problem take the shape $(N_t - N_{lb} - N_{lf}) \times N_{lb}$ and $(N_t - N_{lb} - N_{lf}) \times N_{lf}$, respectively, where N_{lb} and N_{lf} are the number of look-back and look-forward steps. Additionally, the network is able to learn a dependency on the features x_t by adding this information to the inputs of the supervised problem. Assuming static features $x = x_t$, the input of the supervised problem takes the shape $(N_t - N_{lb} - N_{lf}) \times (N_x + N_{lb})$, where N_x is the number of features.

Input	Output
$h_1 \ h_2 \ h_3$	$h_4 \ h_5$
$h_2 \ h_3 \ h_4$	$h_5 \ h_6$
...	...

Figure 6.7: Example of conversion of a time-series into a supervised problem.

Finally, training the network sets the values of the connecting weights and biases by minimizing an error metric. This work uses the Mean Absolute Error (MAE)

$$MAE = \frac{1}{N_{samples} \times (N_t - N_{lb} - N_{lf}) \times N_{lf}} \sum_i |Y_i - \hat{Y}_i| \quad (6.31)$$

where \hat{Y}_i is predicted output, Y_i is the output of the supervised problem or target value, and $N_{samples}$ is the number of samples.

6.2.8 Optimization

Some of the machine learning algorithms described above allow the user to specify their hyperparameter values to improve the results. To obtain the best result, this work uses Bayesian optimization for the hyperparameter tuning.

Bayesian optimization targets systems where the objective function f does not have a closed form or is expensive to evaluate. The idea of Bayesian methods is to update the current hypothesis as new information becomes available. The process initialization uses a Gaussian process as a prior probability distribution. In Gaussian processes, every finite collection of random variables has a multivariate normal distribution.

The prior probability represents what is originally believed before new evidence is introduced. The routine

updates a posterior probability distribution of f every time a new function evaluation is reported. This posterior model is a simpler surrogate function \hat{f} approximating the objective function.

The surrogate is easier to optimize than the objective function. The method finds the sample that performs best on the surrogate function to evaluate it on the objective function. An acquisition function guides the selection of the next sample from the search space. The *expected improvement* is one of the most common acquisition functions, and it indicates the region of the search space that is most likely to improve. Overall, this method allows to reduce the number of evaluations necessary to determine the optimal parameters.

In summary, the algorithm can be itemized as follows

1. Place a Gaussian process prior on f .
2. Observe f for random samples of the search space.
3. Update the \hat{f} using all the available data.
4. Find the point in the search space x_i that maximizes the acquisition function.
5. Get a new observation of f at x_i .
6. Repeat 3- to 5- until convergence.

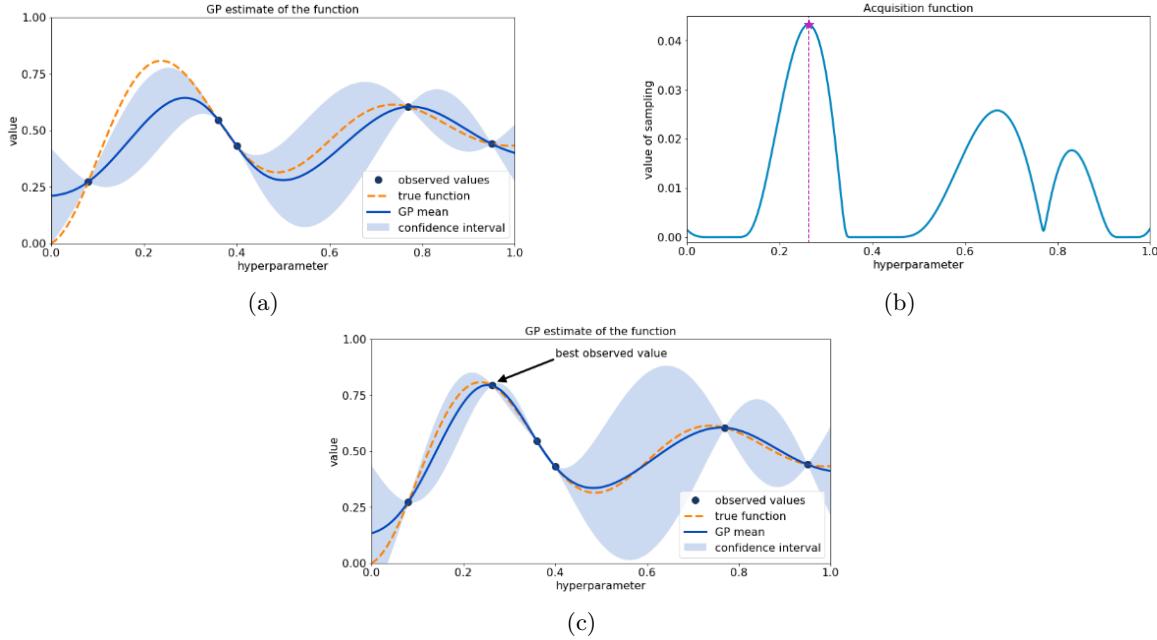


Figure 6.8: Example of Bayesian Optimization process. Image reproduced from [115].

Figure 6.8 displays items 3- to 5-, where \hat{f} is updated with all the available data, the process finds the point x_i that maximizes the acquisition function, and finally, f is evaluated at the new observation point x_i . This work utilizes Bayesian optimization using Gaussian processes implemented through the *gp_minimize* function [116] to optimize the different machine learning algorithms.

6.3 Simplified Model

This section introduces a simplified physics-based model to investigate the feasibility of training and testing a data-based model that can capture the physics of the problem. The model represents a post-irradiation experiment that is cooled by the natural circulation of air, and it comprises a conjugate heat transfer problem characterized by a cavity filled with air and a multi-region solid, as shown in Figure 6.9. The solid consists of an experiment embedded in a capsule. To decrease the computational burden, the model is reduced to a two-dimensional problem. This model neglects the heat production ($q''' = 0$) in the capsule to simplify the analysis. This assumption is justified by the fact that the volume of the experiment is considerably larger than the capsule's.

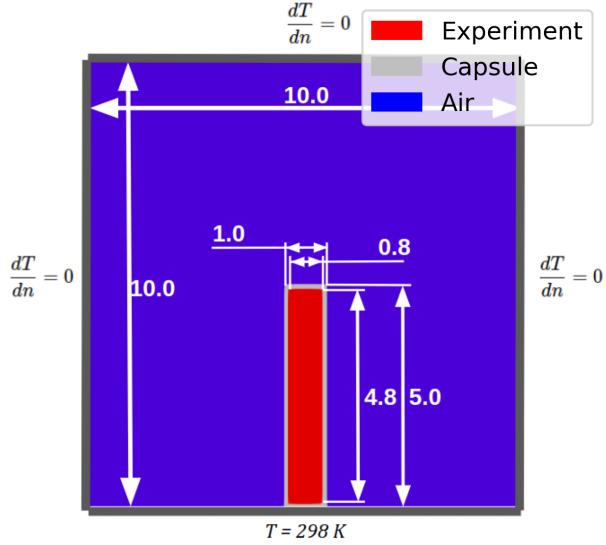


Figure 6.9: Two-dimensional geometry of a post-irradiation experiment cooled by the natural circulation of air. All dimensions are expressed in *cm*.

The boundary conditions defining the problem are adiabatic walls on the side and top edges, while the temperature at the bottom of the geometry is fixed at room temperature (298 K or 25 °C), as shown in Figure 6.9. The initial condition is room temperature for the whole geometry.

This model assumes a laminar flow for solving the equations of motion and temperature of the air. The following correlation defines the transition to the turbulence regime for a vertical plate in natural circulation [117]

$$Ra_{x,c} = \frac{g\beta\Delta Tx^3}{\nu\alpha} \approx 10^9, \quad (6.32)$$

where $Ra_{x,c}$ is the critical Rayleigh number at x distance from the bottom, g is the acceleration due to gravity, β is the thermal expansion coefficient, ΔT is the difference between the surface temperature and the air reference temperature, ν is the kinematic viscosity of the air, and α is the thermal diffusivity of the air. For these simulations, the Rayleigh number is well below 10^9 .

6.3.1 Model verification

This exercise uses MOOSE to solve the physics-based model and OpenFOAM to verify it. As mentioned earlier, MOOSE relies on the Boussinesq approximation for solving the equations of motion and temperature of the air. This approximation may be inaccurate for large gradients of temperature in the fluid. For this reason, this section compares the results from MOOSE and OpenFOAM, which solves the motion and temperature of the air as well as its density without approximation. Additionally, even though the Boussinesq approximation may yield an inaccurate fluid temperature, the comparison metric is the time evolution of the highest temperature in the capsule. MOOSE and OpenFOAM determine the point in the capsule's geometry with the highest temperature, and this work refers to it as the highest temperature in the capsule.

Table 6.1 specifies the material properties of the different regions included in the model. For simplicity, both the capsule and the experiment are made of iron. Note that the properties of the fluid for the thermal-fluid solvers differ, as MOOSE solves the Boussinesq equations while OpenFOAM solves for the density. The experiment heat source decays with respect to time with the shape of $1.25 \times 10^9 \text{ W/m}^3 e^{-t}$ whose initial value is equivalent to 10 kW.

Table 6.1: Material properties of the thermal-fluids model.

Properties	Units	Capsule	Experiment	Air MOOSE	Air OpenFOAM
ρ	kg/m^3	8×10^3	2.7×10^3	1.2	-
k	W/m/K	80	80	2.6×10^{-2}	-
c_p	J/kg/K	450	450	1000	1000
μ	kg/m/s	-	-	1.8×10^{-5}	1.8×10^{-5}
Pr	-	-	-	-	0.7
β	$1/\text{K}$	-	-	3.356×10^{-3}	-

Figure 6.10 shows the time evolution of the highest temperature in the capsule calculated by MOOSE and OpenFOAM. The temperature in the capsule increases sharply from the initial condition of 298 K. The maximum temperature is 562 K and is reached at 4.2 s of the simulation start. Both MOOSE and OpenFOAM predict an almost identical time evolution, demonstrating that MOOSE predicts an accurate temperature evolution.

Figure 6.11 displays the temperature distribution in the experiment and the capsule when the capsule reaches the peak temperature. The highest temperature in the capsule is located in the half top, at 1.85 cm from the top. In the radial direction, and as expected, the highest temperature is located on the inner wall, where the capsule wall meets the experiment. As the experiment produces heat, it is natural to consider that its temperature will be above the capsule temperature. Hence, the highest temperature in the capsule is in the region closest to the experiment.

The model used in this exercise is generic and reactor-agnostic, meaning that it can be tailored to any research reactor. Additionally, this exercise considers an adiabatic boundary condition on the top of the geometry, which does not allow for heat removal or fluid flow in case of a real accident. The final dimensions and boundary conditions of the model are specific to each experiment/reactor configuration and should be tailored accordingly. Moreover, the large temperature shown in the results highlights that radiation heat transfer needs to be included in the model to obtain a more accurate representation of the experiments. Furthermore, the turbulence regime is highly dependent upon the geometry of the experiment, which is specific to the reactor under study. Hence, the assessment of the turbulence regime for each specific geometry is necessary. Although this model has certain limitations, this exercise intends to simplify the creation and

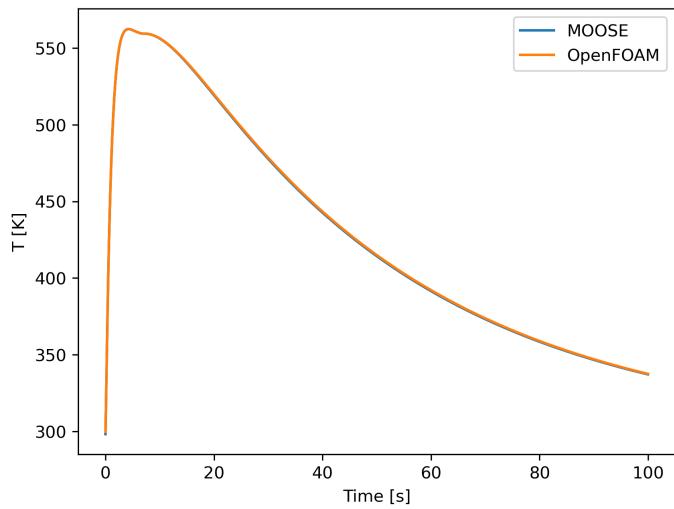


Figure 6.10: Time evolution of the highest temperature in the capsule calculated by MOOSE and OpenFOAM.

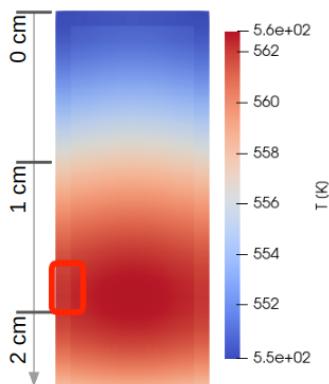


Figure 6.11: Temperature distribution in the experiment and capsule when the capsule reaches the peak temperature. The red square highlights the location of the highest temperature in the capsule.

understanding of a data-based model as well as drawing some useful conclusions.

6.3.2 Data Processing and Model Training

Machine learning derives relationships from data, allowing data-based models to solve complex problems quickly. In this exercise, the collected data comprises the time evolution of the highest temperature in the capsule for certain material properties: density ρ , thermal conductivity k , and heat capacity c_p and the coefficients defining the delay heating: initial value q_0 and decay time τ . These five coefficients are the input variables or *features* of the model, and the highest temperature in the capsule is the output or *label*. The thermal properties values correspond to the 33 first materials that are shown in Appendix E, which correspond to aluminum through iridium. q_0 comprises 45 uniformly distributed values between 10^7 and 10^8 W/m³, and 20 between 10^8 and 5×10^8 W/m³, totaling 65 unique values. τ comprises five uniformly distributed values between 20 and 100 seconds. The combination of all these variables leads to a total of 10725 samples.

The simulation time of the physics-based model was 500 seconds, recording the time evolution in 51 non-uniformly distributed time steps. In summary, the data constitute the input variable X and the output variable Y , having shapes of $N_{samples} \times N_{features}$ and $N_{samples} \times N_{timesteps}$, respectively, with $N_{samples}=10725$, $N_{features}=5$, and $N_{timesteps}=51$. Figure 6.12 provides a visual representation of the input/output pairs of the physics-based model.

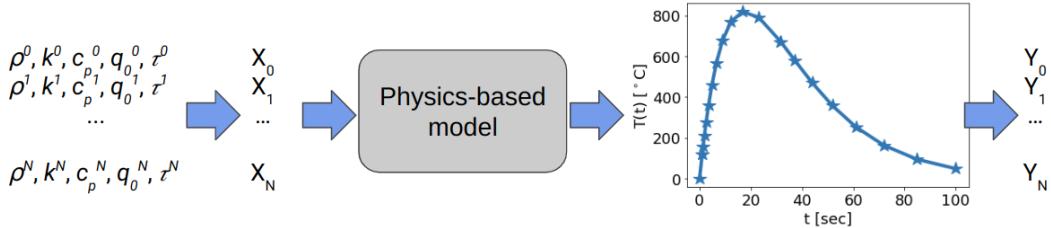


Figure 6.12: Physics-based model input/output pairs.

The prediction of the capsule state considered two different strategies. The first strategy forecasts only whether the capsule melts or not. This is a binary classification problem, in which the two possible outcomes are *melt* and *does not melt*. The second strategy solves a time-evolution prediction problem, which determines if the capsule melts as well as the time of occurrence. The time of occurrence can better guide the operators' countermeasures in case of an accident. When the melting time is a required feature of the model, the data-based model has to forecast the entire time evolution of the highest temperature in the capsule. Once the model predicts the time evolution, it becomes straightforward to resolve if the capsule melts and determine the melting time.

The chosen performance metrics are the accuracy, recall, and training time for the classification problems, and the accuracy, recall, mean melting-time relative error, and maximum melting-time relative error for the time-evolution prediction problem. The accuracy and recall are calculated as follows

$$Accuracy = \frac{TN + TP}{TN + FN + TP + FP} \quad (6.33)$$

$$Recall = \frac{TP}{TP + FN} \quad (6.34)$$

where TN is the number of true negative results, TP is the number of true positive results, FN is the number of false negative results, and FP is the number of false positive results.

The accuracy is the ratio of the correctly classified samples over the total number of samples. The recall is the ratio between correctly classified relevant samples over the total number of relevant samples. The accuracy does not provide information on the distribution of the correctly classified samples. The recall resolves this issue, indicating the quality of the classification within each category. This work uses the recall of the melting category, which is the ratio between the number of samples predicted to melt over the total number of samples that melt. This approach is more conservative than using the recall of the non-melting category. If the model predicts that a sample melts when it does not, any consequential corrective measure will not trigger a radioactive material release. The confusion matrix, shown in Figure 6.13, aids in the visualization of the performance of classification algorithms.

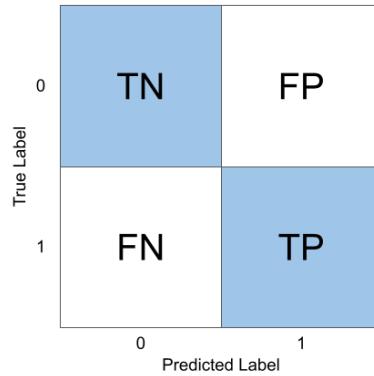


Figure 6.13: Confusion matrix. T: True, F: False, N: Negative, P: Positive.

The data handling process for the classification problem is summarized in Figure 6.14. The output variable Y is transformed into the binary variable \bar{Y} , in which 0 represents that the capsule *does not melt* and 1 that it *melts*. If the temperature of a sample goes over the capsule melting temperature, then, \bar{Y} takes the value 1, and 0 otherwise. This work considers a capsule melting temperature of 923 K (650 °C).

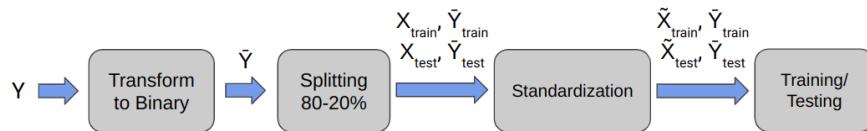


Figure 6.14: Classification problem data handling.

The pre-processing of the data involves the splitting and standardization. The samples are randomly split into training and testing datasets. Of the samples, 80% integrate the training dataset, and the remaining 20%, the testing dataset. The standardization method scales the datasets by subtracting the mean value μ of the samples and dividing the result by the standard deviation σ for each feature j

$$\tilde{X}^j = (X^j - \mu^j)/\sigma^j. \quad (6.35)$$

Standardization is a feature scaling technique that is a common requirement for many machine learning estimators. If a feature has a variance that is orders of magnitude larger than the others, it might dominate the objective function, falling within a local minimum during training and making the estimator unable to learn as expected. After the data pre-processing, the data-based model is trained using the training dataset for the standardized input and output variables \tilde{X}_{train} and \tilde{Y}_{train} . The data-based model uses the following machine learning techniques: DT, FNN, KNN, LogR, RF, and SVM. Once the model is trained, the model can predict the outcome of the transient by reading the material properties of a specific sample, as shown in Figure 6.15.

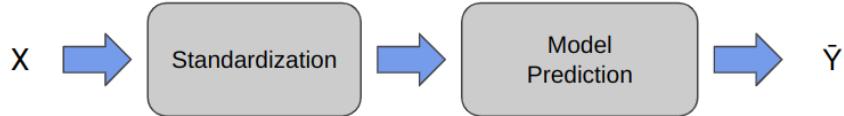


Figure 6.15: Visualization of classification prediction.

The only optimized methods were the KNNs and the FNNs, while the rest of the methods utilized the *scikit-learn* default parameters. The optimization of the KNN was carried out via grid search and explored different numbers of N-nearest neighbors to maximize the recall. The optimization of the FNN was based on Bayesian optimization using Gaussian Processes (Section 6.2.8) to maximize the recall. The training of the network used the Adam optimizer to minimize binary cross-entropy. The FNN number of layers was fixed to four, for which the input and output layers had five and one node, respectively, and the two hidden layers had a variable number of nodes. The optimization of the FNN determined the number of nodes and training batch size. The optimal configuration was 104 and 155 nodes for the hidden layers, and a batch size of 16.

Figure 6.16 displays the prediction of the time-evolution prediction problem. The output is the whole time evolution of the highest temperature in the capsule, which is translated into \tilde{Y} containing the *melt/does not melt* outcome as well as the capsule's melting time. This work uses three different approaches to predict the time evolution, and the data handling process is slightly different for each of them.

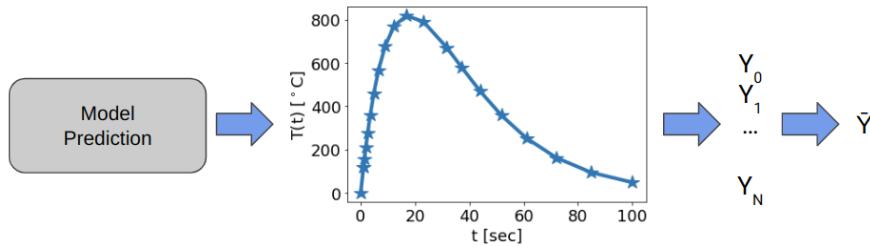


Figure 6.16: Visualization of time-evolution prediction.

The first method is based on the regression of non-linear fitting coefficients defining the time evolution. As a decaying exponential defines the heat production in the experimental device, it is natural to assume that the temporal behavior of the sample temperature is also a combination of decaying exponentials, and the curve $A(e^{-\lambda_1 t} - e^{-\lambda_2 t})$ approximates the temperature response well. A non-linear curve fitting uses the *curve_fit* function [118] to determine the coefficients of the curve A , λ_1 , and λ_2 . Figure 6.17 displays an example of the non-linear curve fitting to discretize the capsule temperature evolution.

The data handling process for this method is summarized in Figure 6.18. The coefficients A , λ_1 , and λ_2 compose an intermediate output variable \hat{Y} with shape $N_{samples} \times 3$. The samples are separated into

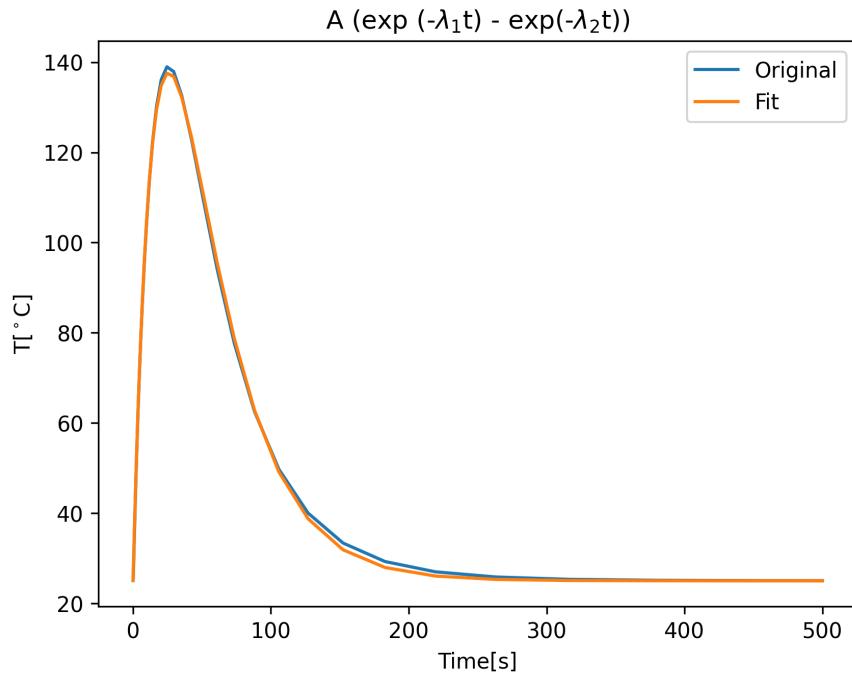


Figure 6.17: Example of the non-linear curve fitting to discretize the capsule temperature evolution.

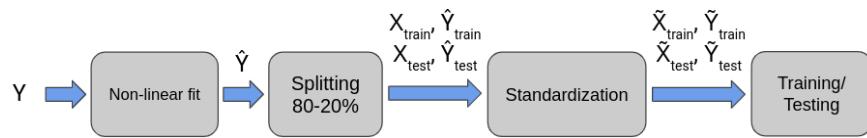


Figure 6.18: Data handling process for the time-evolution prediction using regression of non-linear fitting coefficients.

an 80-20 training-testing split, and standardization is applied. The data-based model is trained with the standardized input and output variables, \tilde{X}_{train} and \tilde{Y}_{train} . Once the model is trained, the model can predict the outcome of the transient by reading the material properties of a specific sample, as shown in Figure 6.19. Then, the trained model predicts \tilde{Y} that is used to reconstruct the temperature time evolution Y , and obtain the output variable \bar{Y} . This work uses Linear Regression (LR), SVR, DTR, and RFR to build the data-based model. However, some of these methods are unable to capture the non-linear behavior of the different curve fitting coefficients. Section 6.3.3 discusses the results for the best-performing methods – DTR and RFR. These models were not optimized and used *scikit-learn* default parameters to reduce the complexity.

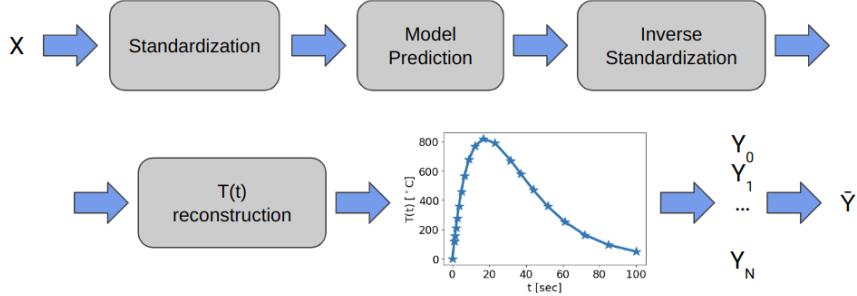


Figure 6.19: Visualization of time-evolution prediction using regression of non-linear fitting coefficients.

The second method uses FNNs to predict the whole time evolution of the highest temperature in the capsule at once. The data handling process for this method is summarized in Figure 6.20. The samples are separated into an 80-20 training-testing split. After splitting the data into training and testing datasets, the datasets are scaled. The input variables X_{train} and X_{test} are standardized, while the output variables Y_{train} and Y_{test} are normalized instead. Normalization is a feature scaling technique that uses the maximum and minimum value of each feature to scale the data

$$\tilde{X}^j = (X^j - X_{min}^j) / (X_{max}^j - X_{min}^j). \quad (6.36)$$

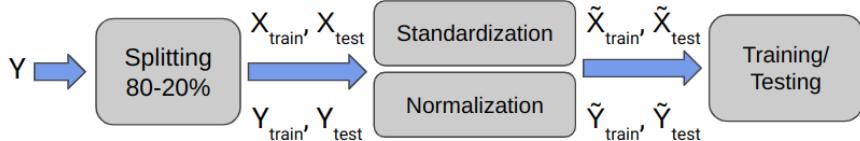


Figure 6.20: Data handling process for the time-evolution prediction using FNNs.

The network training used the Adam optimizer to minimize the MAE. The neural network had a 5-node input layer, two hidden layers, and a 51-node output layer. The network optimization set the number of nodes in the hidden layers, the training learning rate, and training batch size. Bayesian optimization using Gaussian processes (Section 6.2.8) optimized the network by maximizing the R^2 score. The optimal configuration was 169 and 133 nodes for the hidden layers, a learning rate of 6.9×10^{-4} , and a batch size of 49. Once the model is trained, the model can predict the outcome of the transient by reading the material properties of a specific sample, as shown in Figure 6.21. Then, the trained model predicts \tilde{Y} that is de-normalized into Y , and translated into the output variable \bar{Y} .

The third method uses LSTMs to predict the whole time evolution of the highest temperature in the

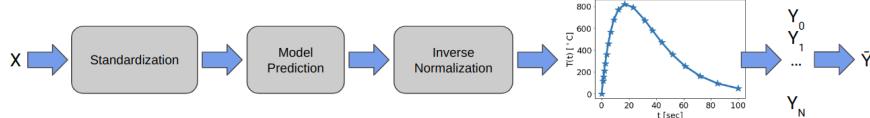


Figure 6.21: Visualization of time-evolution prediction using FNN.

capsule at once. The data handling process for this method is summarized in Figure 6.22. The first step in the LSTM implementation transforms the problem into a supervised one, as described in Section 6.2.7. The number of look-forward steps is fixed to 40, while the network optimization defines the number of look-back steps. Then, the samples are normalized and separated into an 80-20 training-testing split.

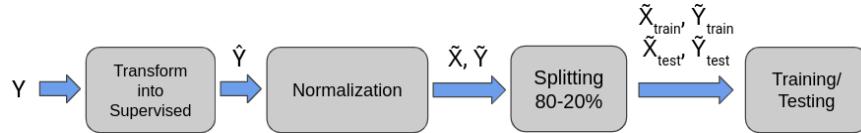


Figure 6.22: Data handling process for the time-evolution prediction using LSTMs.

The network has four LSTM layers and one dense output layer. The network optimization defines the number of nodes in the LSTM layers, the number of look-back steps, and the training batch size. The training of the network uses the Adam optimizer to minimize the MAE, and the learning rate is fixed to 5×10^{-4} . Bayesian optimization using Gaussian processes (Section 6.2.8) optimized the network by maximizing the R^2 score. The optimal configuration was 167, 98, 62, and 139 nodes for the hidden layers, six look-back steps, and a batch size of 12. Once the model is trained, the model can predict the outcome of the transient by reading the material properties of a specific sample, as shown in Figure 6.23. Then, the trained model predicts \tilde{Y} that is de-normalized into \hat{Y} , which is used to reconstruct the time-series Y and translated into the output variable \bar{Y} .

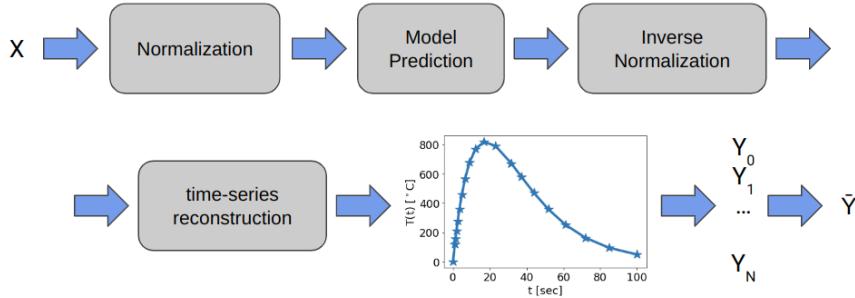


Figure 6.23: Visualization of time-evolution prediction using LSTM.

6.3.3 Strategy Comparison

This section compares the performance of the two different strategies described in Section 6.3.2. The first strategy relies on classification methods to predict the state of the experiment's capsule. The second strategy uses several approaches to predict the time evolution of the capsule temperature.

Table 6.2 summarizes the main results of the classification problem for the testing dataset. In terms of accuracy, LogR and FNN are the worst and best-performing methods, respectively. All the methods show

an accuracy higher than 0.96. When focusing on the recall, both methods show again the worst and best performances. LogR is the only method with a recall lower than 0.97. The last comparison metric is the model training time. The FNN took longer to train than the other models. All the methods are trained quickly in under 1 second, except for FNN which took almost half a minute. This training time does not include the hyperparameter tuning, which takes a substantially longer time, in the order of 10 to 20 minutes. Overall, all these values are small and acceptable, and the FNN remains the preferred method. Figure 6.24 displays the confusion matrix for the LogR and the FNN cases. The comparison of both results shows a clear improvement from LogR to FNN in the mislabeled samples.

Table 6.2: Performance metrics of the machine learning techniques for the classification problem.

Method	Accuracy	Recall	Train Time [sec]
DT	0.9902	0.9705	<1
FNN	0.9963	0.9937	27
KNN	0.9860	0.9747	<1
LogR	0.9655	0.9116	<1
RF	0.9911	0.9726	<1
SVM	0.9841	0.9832	<1

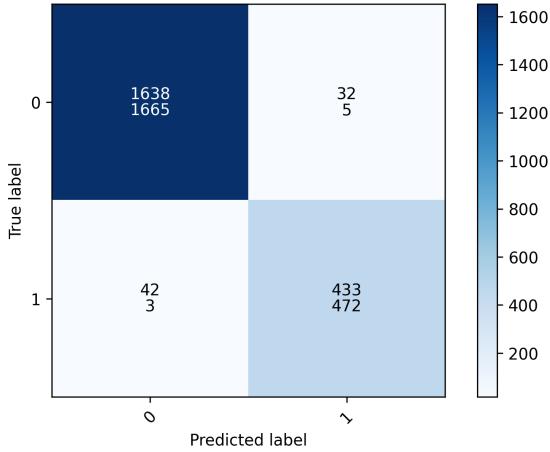


Figure 6.24: Confusion matrix for the LogR (top) and FNN (bottom) cases. The labels 1/0 represent melt/does not melt.

Table 6.3 summarizes the main results of the time-evolution prediction problem for the testing dataset. Overall, all methods achieved high accuracy and recall. DTR shows the worst performance, with an accuracy lower than 0.99, and a recall lower than 0.97. The FNN shows the highest accuracy and recall. In terms of melting-time relative error, all the methods show large maximum relative errors, in the order of 30 to 50%. The mean relative error, on the other hand, is low for the FNN and the LSTM. The RFR shows the worst performance, with a mean relative error close to 18%.

Regarding the accuracy, recall, and mean melting-time relative error, the FNN is the best-performing method. LSTM has high accuracy and recall and a low mean melting-time relative error. Hence, it is the second best performing method. DTR and RFR are the worst-performing methods.

Depending on the final application of the models, one may benefit from choosing the regression or the deep-learning methods. Although the tables do not present the training times, FNN and LSTM required

Table 6.3: Performance metrics of the different methods for the time-evolution prediction problem.

Method	Accuracy	Recall	Mean t_{Melt}	Rel. Error	Max. t_{Melt}	Rel. Error
DTR	0.9889	0.9617		8.2 %		43.0 %
FNN	0.9976	1.0000		1.0 %		31.1 %
LSTM	0.9952	0.9933		3.4 %		29.0 %
RFR	0.9928	0.9797		17.6 %		45.1 %

optimization, which translates into high training times. Another desirable feature can be visualization. Deep-learning methods work in a “black box”-fashion, where the visualization of the network does not provide much insight to the scientist. The regression of the non-linear fitting coefficients, on the other hand, is a method relatively simple and easy to visualize. However, the method contains multiple steps, wherein uncertainties could be introduced, deteriorating the quality of the final result.

Figure 6.25 displays some examples of time-evolution prediction using the three approaches. It is possible to discern the time discretization in all the curves. As machine-learning algorithms have to predict the temperature evolution, it simplifies the learning process to reduce the number of time steps. For the most part, when trying to determine whether the capsule melts or not, a coarse time discretization is still enough to capture this behavior. When trying to predict the exact time when the capsule melts, such a coarse time discretization introduces considerably large errors for some of the testing samples, which is the cause of the large maximum t_{Melt} relative errors shown in Table 6.3. Overall, the predicted values approximate well the expected values, and hence the mean t_{Melt} relative error is low, as shown in Table 6.3.

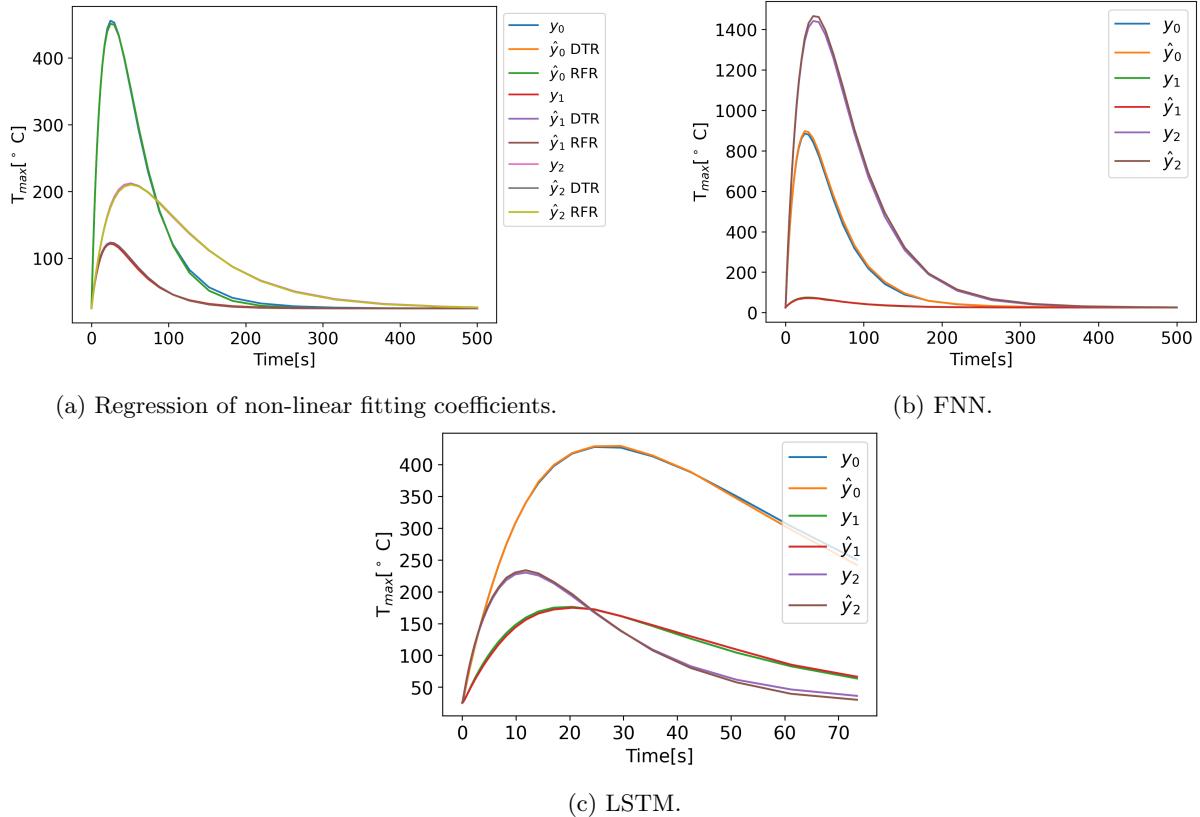


Figure 6.25: Example of time-evolution prediction using the three approaches.

6.3.4 Non-dimensional thermal properties

This section investigates an alternative for the classification problem to reduce the model complexity. Instead of using all the material thermal properties to train/test the model (three features), the model uses only one feature: the thermal diffusivity

$$\alpha = \frac{k}{\rho c_p} \quad (6.37)$$

where k is the thermal conductivity, ρ is the density, and c_p is the specific heat capacity. The motivation for this study is that reducing the number of features should allow the model to be trained more “easily”. The more features the model has, the more complex the relationship between input and outputs becomes. By reducing the number of features, the input/output relationship should become simpler, and hence the model is easier to train. This study intends to confirm/refuse this hypothesis by using the same data from the study from previous section.

Table 6.4 summarizes the main results of the classification problem for the testing dataset. In terms of accuracy, LogR and DT are the worst and best-performing methods, respectively. All the methods show an accuracy higher than 0.95. When focusing on the recall, both methods show again the worst and best performances. LogR is the only method with a recall lower than 0.95. The last comparison metric is the model training time. The FNN took longer to train than the other models. All the methods are trained quickly in under 1 second, except for FNN which took 1 minute. However, all these values are very small and acceptable, and the FNN remains as the preferred method. Figure 6.26 displays the confusion matrix for the LogR and the DT cases. The comparison of both results shows a clear improvement from LogR to DT in the mislabeled samples.

A comparison between the results from Section 6.3.3 and these results shows that all methods perform better when considering all the thermal properties as features. Although condensing the thermal properties into one value seems to decrease the complexity of the model, the accuracy and recall of the model worsen slightly. This is because using the thermal diffusivity as a feature causes the model to lose some resolution, as the thermal diffusivity “hides” the variations in the thermal properties. Figure 6.27 helps to visualize this phenomenon. For example, if the model was trained with materials with extreme values - i.e., lithium and iridium, all the rest of the materials’ thermal properties would fall within the lithium and iridium bounds. If sodium becomes the testing sample, for the model trained with all the thermal properties, its thermal properties are within the bounds of the training data. On the other hand, for the model trained with the thermal diffusivity, the sodium testing sample falls outside the bounds of the training data, deteriorating the model prediction. Although these results suggest that training the model with all the thermal properties is best, the results do not worsen considerably, and more investigation should confirm/refuse this analysis.

6.3.5 Synthetic Materials

This section investigates the training/testing of the models using synthetic thermal properties. The motivation for this study is to augment the training/testing data to broaden the range of applicability of the data-based model to more possible irradiated materials. These synthetic thermal properties are generated by randomly sampling each thermal property assuming a uniform distribution between the minimum and maximum values from the list of 82 real materials included in Appendix E.

The thermal properties values correspond to 200 synthetic materials and 82 real materials. q_0 comprises

Table 6.4: Performance metrics of the machine learning techniques for the classification problem using thermal diffusivity as a feature.

Method	Accuracy	Recall	Train Time [sec]
DT	0.9883	0.9797	<1
FNN	0.9739	0.9684	60
KNN	0.9776	0.9684	<1
LogR	0.9585	0.8736	<1
RF	0.9855	0.9729	<1
SVM	0.9692	0.9503	<1

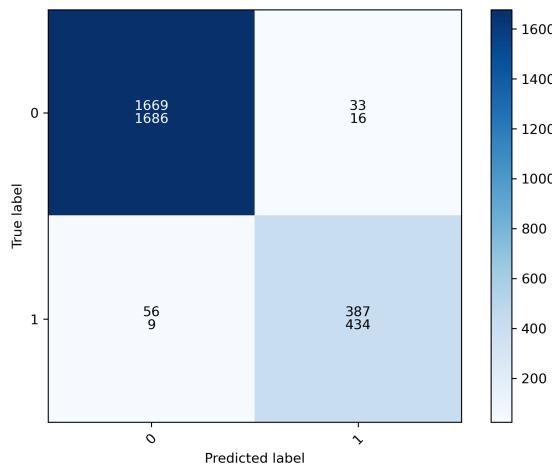


Figure 6.26: Confusion matrix for the LogR (top) and DT (bottom) cases. The labels 1/0 represent *melt/does not melt*.

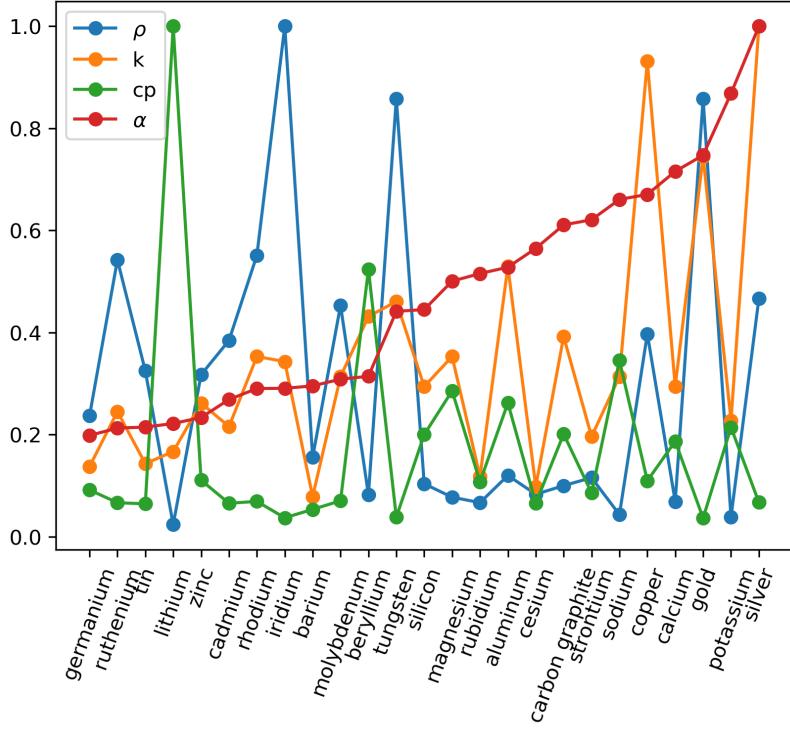


Figure 6.27: Thermal properties for selected materials. Values are normalized to maximum values of materials in the shown range.

18 uniformly distributed values between 10^7 and 10^8 W/m^3 , and 20 between 10^8 and $5 \times 10^8 \text{ W/m}^3$, totaling 38 unique values. τ comprises five uniformly distributed values between 20 and 100 seconds. The combination of all these variables leads to a total of 53390 samples. However, some of the samples were discarded because the simulations did not reach completion for uninvestigated reasons, and the total number of usable samples was 40550.

Table 6.5 summarizes the main results of the classification problem for the testing dataset. In terms of accuracy, LogR and FNN are the worst and best-performing methods, respectively. All the methods show an accuracy higher than 0.96. When focusing on the recall, both methods show again the worst and best performances. LogR is the only method with a recall lower than 0.94. The last comparison metric is the model training time. The FNN took longer to train than the other models. All the methods are trained quickly in under 1 second, except for FNN which took over 4 minutes. However, all these values are very small and acceptable, and the FNN remains as the preferred method. Figure 6.28 displays the confusion matrix for the LogR and the FNN cases. The comparison of both results shows a clear improvement from LogR to FNN in the mislabeled samples.

A comparison between the results from Section 6.3.3 and these results shows that all methods perform better when considering real data for the thermal properties. Figure 6.29 shows the relationship between thermal properties for selected real materials. The plots show certain correlation between the different thermal properties. The one most obvious is the relationship between heat capacity and density which follows a $1/x$ -shape. For the other thermal properties, the correlations are not so easily discernible. However, a clear aspect is that the plotted values are not uniformly distributed, but concentrated in specific areas of the domain. For example, the relationship between heat capacity and thermal conductivity shows only a few

points above 1000 J/kg/K, while the rest of them are concentrated below 500 J/kg/K and below 100 W/m/K. These figures prove that assuming a uniform distribution for the thermal properties between minimum and maximum values is not appropriate. In conclusion, this assumption causes the models to learn the correlations between input features in areas of the domain that are not representative of reality and should be avoided.

Table 6.5: Performance metrics of the machine learning techniques for the classification problem.

Method	Accuracy	Recall	Train Time [sec]
DT	0.9861	0.9433	<1
FNN	0.9948	0.9808	4'26"
KNN	0.9847	0.9452	<1
LogR	0.9676	0.8528	<1
RF	0.9898	0.9525	<1
SVM	0.9938	0.9771	<1

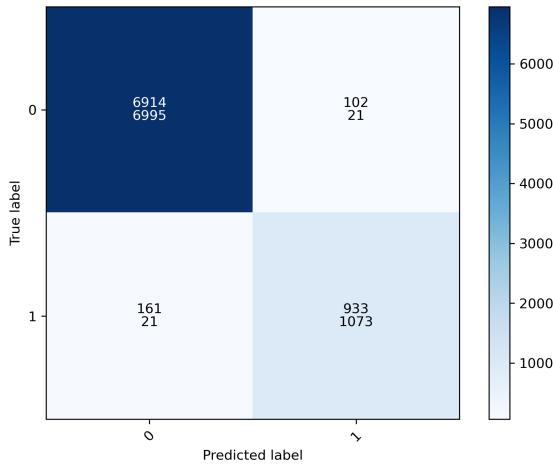


Figure 6.28: Confusion matrix for the LogR (top) and FNN (bottom) cases. The labels 1/0 represent melt/does not melt.

6.4 Source Decomposition

The analyses described above assume that the experiment produces heat with a source of shape $q_0 e^{-t/\tau}$. Further inspection of the delayed heating results from Chapter 3 shows that the experiment heat does not follow the shape of a pure exponential decay. Figure 6.30 shows the delayed heating in the demonstration problem from Section 3.3.2. This figure also shows the consequence of approximating the delayed heating with a pure exponential decay. While the pure exponential is able to capture the mean behavior of the delayed heating, it does not capture the instant values.

This section studies the possibility of decomposing the delayed heating over time into a linear combination

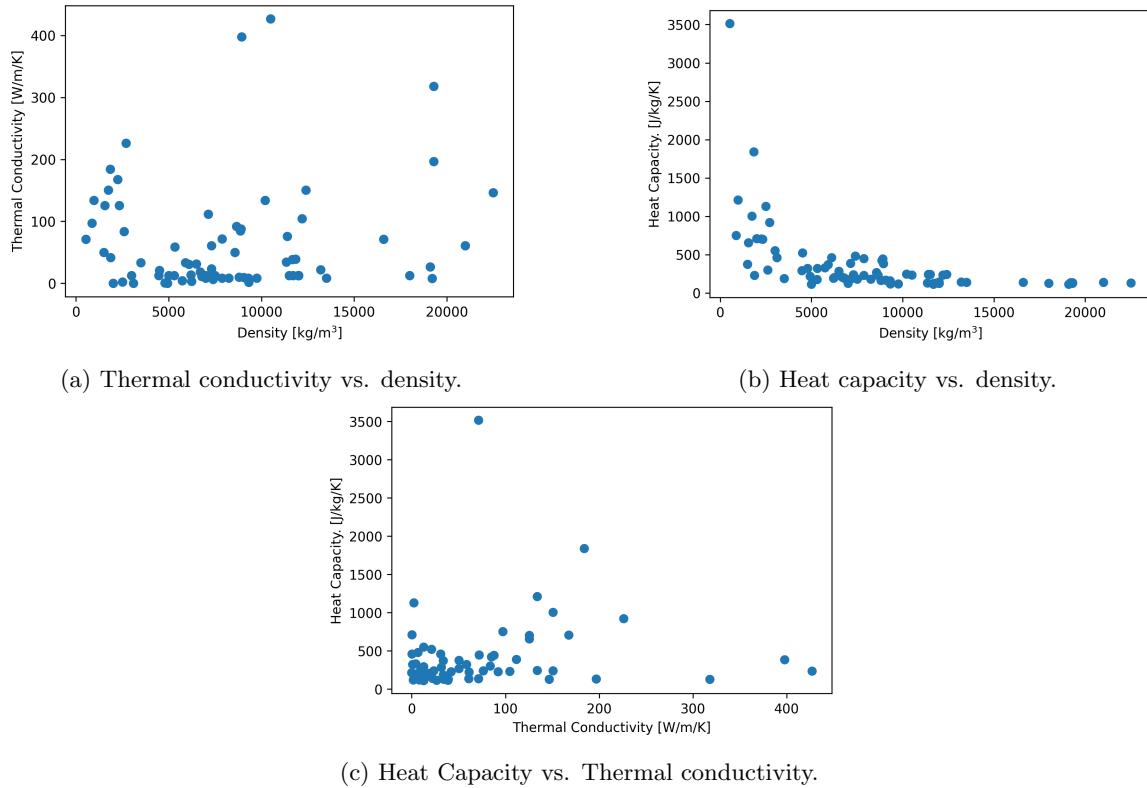


Figure 6.29: Relationship between thermal properties for selected materials.

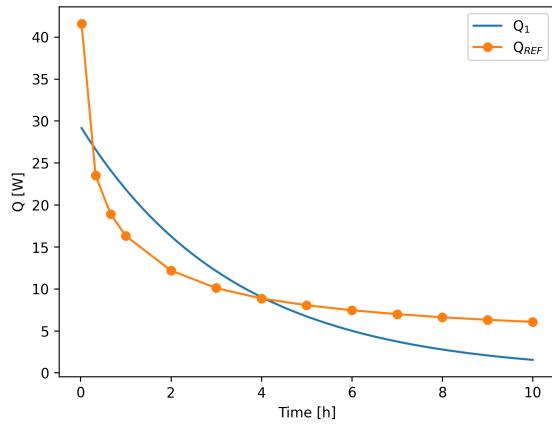


Figure 6.30: Source representation with one pure exponential decay.

of exponential decay functions

$$Q(t) = \sum_i^N Q_i(t) \quad (6.38)$$

$$Q_i(t) = Q_{i,0}e^{(-t/\tau_i)} \quad (6.39)$$

where $Q_{i,0}$ is the initial value, τ is the decay constant, and N the number of terms. The result from some investigation in available algorithms to solve this problem highlights the Prony analysis developed in 1795. More recent work on practical implementations of the method [119][120] intended to solve the method's drawbacks. Since the original method finds the exponential coefficients using the roots of a polynomial, the coefficients can be positive, negative, or complex-valued. The newer methods still have some drawbacks, and some results show estimation errors in the order of 50 % [121].

For this reason, this work uses a slightly different approach where a function of variable number of terms is chosen and their coefficients are fitted to the delayed heating. Then, by looking at the L_2 -norm error between the reference values and the fitted functions, it is possible to determine the most appropriate number of terms. The implementation is based on the *curve_fit* function [118].

To investigate this source decomposition strategy, this study uses the delayed heating over time from the demonstration problem from Section 3.3.2 and the ATR experiment from Section 3.3.3. Figures 6.31 and 6.32 display the results for the demonstration case and the ATR experiment. Figures 6.31a and 6.32a show the approximation to the original data using an increasing number of terms. A function using three terms is good enough to represent the delayed heating over time. Figures 6.31b and 6.32b confirm this by displaying the L_2 -norm error for the cases with an increasing number of terms. For the ATR case, the L_2 -norm error for the 3-term and 4-term functions are equal. The *curve_fit* function fails to resolve the coefficients of the 4th term because more delayed heating data-points are required.

Figures 6.31c and 6.32c display the individual terms of the approximated delayed heating. The delayed heating of the demonstration case is decomposed into the following functions

$$Q_1(t)[W] = 14.96e^{(-t/1.296h)} \quad (6.40)$$

$$Q_2(t)[W] = 9.97e^{(-t/19.586h)} \quad (6.41)$$

$$Q_3(t)[W] = 19.89e^{(-t/0.150h)} \quad (6.42)$$

And the delayed heating of the ATR experiment is decomposed into the following functions

$$Q_1(t)[W] = 550.7e^{(-t/0.019h)} \quad (6.43)$$

$$Q_2(t)[W] = 117.0e^{(-t/19.078h)} \quad (6.44)$$

$$Q_3(t)[W] = 188.4e^{(-t/1.057h)} \quad (6.45)$$

Both cases are decomposed into three exponential decay functions. Each of the functions represents the source behavior at different time periods: short, intermediate, and long. Both cases show a sharp decay at the beginning of the transient and a slower decay at the end. This behavior is captured by the functions with smallest and largest decay times, respectively.

These results show that the delayed heating can be approximated with a linear combination of exponential decay functions. Utilizing three terms in the linear combination has proven a good approximation for the

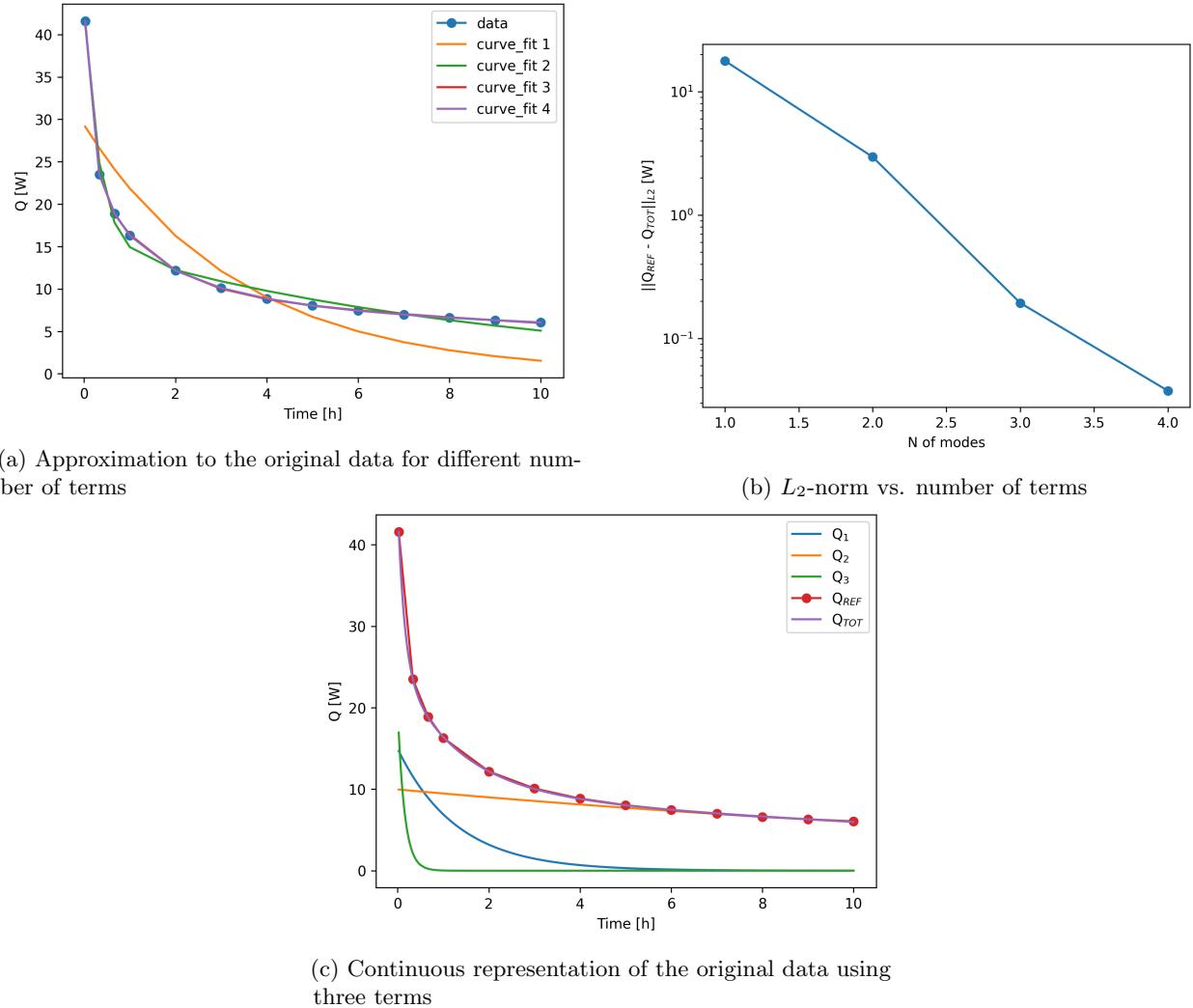


Figure 6.31: Delayed heating over time approximated by a linear combination of exponential decay functions for demonstration problem of Section 3.3.2.

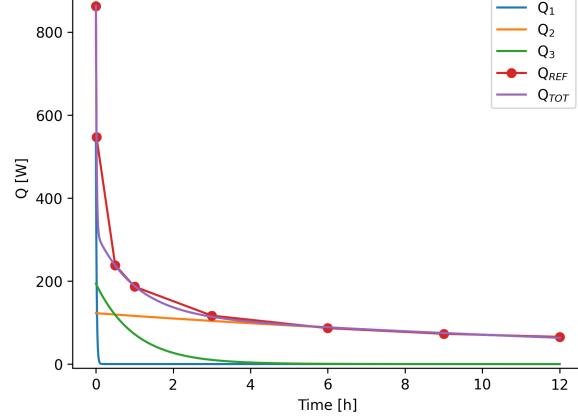
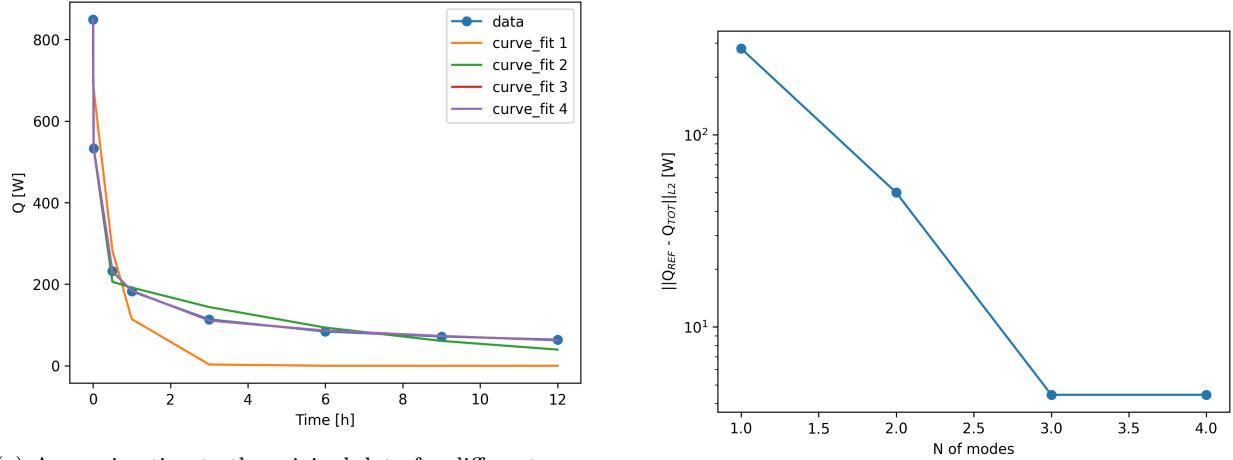


Figure 6.32: Delayed heating over time approximated by a linear combination of exponential decay functions for ATR experiment of Section 3.3.3.

demonstration problem and the ATR experiment. Additionally, the main implication from these results is that the data-based models utilized in the simplified problem from Section 6.3 are not directly applicable to a real reactor without a small modification. Previously, the data-based models of the simplified model defined five input features - i.e., ρ , c_p , k , q_0 , and τ . The new models should also include the exponential coefficients defining all the terms in the delayed heating decomposition: ρ , c_p , k , $q_{1,0}$, $q_{2,0}$, $q_{3,0}$, τ_1 , τ_2 , and τ_3 .

6.5 ATR experiment

This section studies the application of data-based modeling to a full-size experiment hosted by the ATR. The training/testing data is the temperature evolution of the capsule during the channel draining event, and it is generated using OpenFOAM. As the capsule/experiment specification for ATR is not openly available, the designs utilized in HFIR experiments guide the modeling assumptions. Figure 6.33 shows a typical finned rabbit design utilized in HFIR. The term rabbit housing refers to the outer capsule, which is in direct contact with the reactor primary coolant, used to contain the sample material.

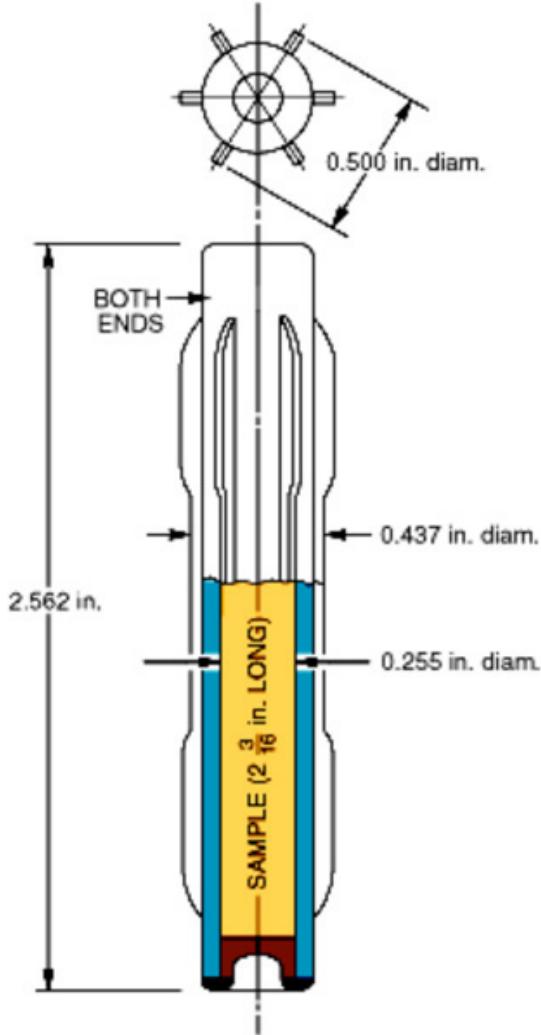


Figure 6.33: Typical finned capsule (also known as rabbit) design utilized in HFIR experiments. Image reproduced from [10].

There are a total of six different rabbit housing designs approved for HFIR [10]:

- standard finned rabbit: this is the original design developed for use in a single tube hydraulic facility to conduct irradiations supporting isotope production.
- increased bore finned rabbit: similar to the standard finned rabbit with a larger inner diameter to provide larger volume for sample material.
- reduced bore finned rabbit: similar to standard finned rabbit with a decreased inner diameter to minimize sample temperature when irradiating small samples.
- un-finned rabbit: the whole concept of finned tubes is to increase the outside surface area of the tube in contact with the coolant and enhance its cooling. Un-finned rabbits allow for higher testing temperatures in the samples and are primarily used to conduct irradiations supporting materials irradiation testing.
- un-finned selenium rabbit: similar to un-finned rabbit, with the specific use of selenium pellet irradiation.
- perforated rabbit: allows the reactor coolant to flow into the specimens to maintain test specimen temperature as low as possible. It is exclusively used for material irradiation testing when low testing temperatures are required.

This work adopts the un-finned rabbit capsule design as the problem geometry for the thermal-fluids model of the ATR experiment, as shown in Figure 6.34. The geometry uses a two-dimensional representation due to azimuthal symmetry, and it is represented as a wedge with a symmetric 2° opening with respect to the symmetry plane. Two aspects should be mentioned about the modeling choice. First, modeling azimuthal symmetry in OpenFOAM requires a wedge representation of the geometry instead of a plane as in FEM tools as it is a FVM tool and requires the definition of volumes in the mesh. Second, the choice of a two-dimensional representation for other rabbit capsule designs that include fins may be inaccurate and should be investigated.

The dimensions of the geometry are based on the ATR experiment dimensions and typical dimensions of HFIR finned and un-finned rabbits. This exercise focuses on the A1 experiment position which has a fixed 0.625" diameter or 7.94 mm radius and sets the radius of the channel outer wall. The gap between the capsule and channel outer walls is defined by the fin width which is 0.8 mm. The thickness of the capsule is defined by the thickness of the increased bore finned rabbit which is 1.4 mm, and it is assumed for the capsule walls in the radial and axial direction. Finally, the capsule length is set by the experiment position height, being 1.22 m [9].

The problem defines a multi-region geometry with two solid regions: capsule and experiment, and one fluid region: air. The capsule is made of aluminum 6061-T6 [9] with the material properties specified in Table 6.6. The emissivity of the aluminum is necessary when including a thermal radiation model. Its value is highly dependent on the material surface conditions. For highly polished aluminum, the emissivity ranges from 0.039 to 0.057, and this work assumes an average value of 0.05. The air material properties assume a 323 K (50°C) temperature and 100 kPa pressure and are shown in Table 6.6. Since the experiment material is variable, its material properties are not fixed and are not included in the table. Appendix E shows all the materials included in the model and their properties.

The mesh of the model was created with GMSH v4.10.5[122] and is shown in Figure 6.35. It was created as coarse as possible to reduce the computation expense, while still allowing for a smooth representation of the temperature in the different regions. The two-dimensional representation of the geometry in OpenFOAM requires the boundary surfaces to straddle the symmetry plane equally with a single layer of cells in the

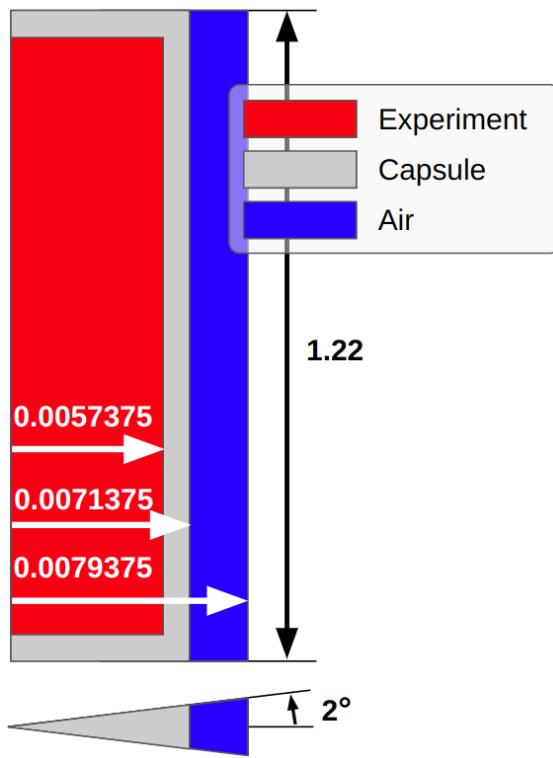


Figure 6.34: ATR experiment geometry for the thermal-fluids model. Dimensions not at scale to better display the model regions. Dimensions expressed in m .

Table 6.6: Material properties of capsule and air.

Properties	Units	Capsule	Air
Molecular weight	g/mol	27	28.9
ρ	kg/m ³	2.7×10^3	-
k	W/m/K	167	-
c_p	J/kg/K	896	1000
ε	-	0.05	-
Melting temperature	°C	650	-
μ	kg/m/s	-	1.8×10^{-5}
Pr	-	-	0.7

azimuthal direction, and the points on both surfaces should appear collocated when viewed in the symmetry plane normal direction.

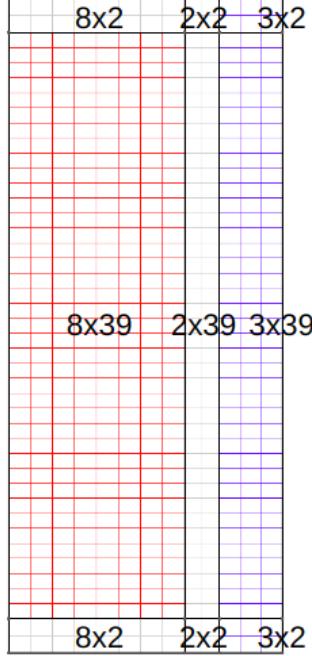


Figure 6.35: Mesh of the ATR experiment geometry for the thermal-fluids model.

This model assumes a laminar flow for solving the equations of motion and temperature of the air. The following correlation defines the transition to the turbulence regime for a vertical channel with natural circulation [117]

$$Ra_S = \frac{g\beta\Delta TS^3}{\alpha\nu} \quad (6.46)$$

where Ra_S is the Rayleigh number, g is the acceleration due to gravity, β is the thermal expansion coefficient of air, ΔT is the difference between the surface temperature and air inlet temperature, S is the channel thickness, α is the thermal diffusivity of air, and ν is the kinematic viscosity of air. Assuming an air reference pressure and temperature of 100 kPa and 323 K (50 °C), β , ν and α take the values 0.003096 1/K, 1.48×10^{-5} , and 2.114×10^{-5} . Assuming a capsule wall temperature of 2000 K, the Ra_S is 83. With a transition to turbulent occurring Ra equal to 10^9 , and a capsule melting temperature of 923 K, the flow in the channel remains laminar. As discussed in Section 6.3, the addition of a thermal radiation model is necessary. The model of the ATR experiment includes the thermal radiation model P1 already available in OpenFOAM.

The boundary conditions defining the problem are specified in Figure 6.36. The two-dimensional representation with azimuthal symmetry in OpenFOAM uses the *wedge* boundary condition in the surfaces on each side of the wedge. This boundary condition applies to all the regions in the geometry. The top and bottom surfaces of the capsule are insulated, which is defined in OpenFOAM with the *zeroGradient* boundary condition. In the fluid, OpenFOAM requires boundary conditions for the temperature T , velocity U , and incident radiation field G . Additionally, OpenFOAM does not solve for the static pressure directly but solves

for

$$p_{rgh} = p - \rho gh \quad (6.47)$$

where p is the static pressure and ρgh is the hydrostatic pressure. The temperature in the channel wall is assumed to be fixed at 323 K (50 °C), and it is defined with the *fixedValue* boundary condition. The channel and capsule walls consider a *noSlip* boundary condition that sets the velocity equal to zero. The *p_rgh* in the channel and capsule walls takes the *fixedFluxPressure* boundary condition, which sets the pressure gradient to a provided value such that the flux on the boundary is that specified by the velocity boundary condition. The temperature at the inlet and outlet uses the *InletOutlet* boundary condition, which applies a zero gradient boundary condition to the outflow and fixes the temperature of the reverse flow (fixed at 323 K). The inlet velocity uses the *pressureInletOutletVelocity* boundary condition, which assigns a zero gradient boundary condition to the outflow and assigns a velocity based on the flux in the boundary-normal direction to the inflow. The outlet velocity uses the *InletOutlet* boundary condition, and it applies a fixed zero velocity for the reverse flow. The *p_rgh* uses the *totalPressure* and *fixedValue* boundary conditions for the inlet and outlet, respectively. The *totalPressure* imposes a value of 100 kPa for the total pressure p_0 , and the value for *p_rgh* is calculated as $p_0 - \frac{\rho \|U\|^2}{2}$. The outlet *p_rgh* is fixed at 100 kPa. All fluid boundaries assume the *MarshakRadiation* boundary condition for the incident radiation field G . The initial condition is room temperature (323 K) for the whole geometry.

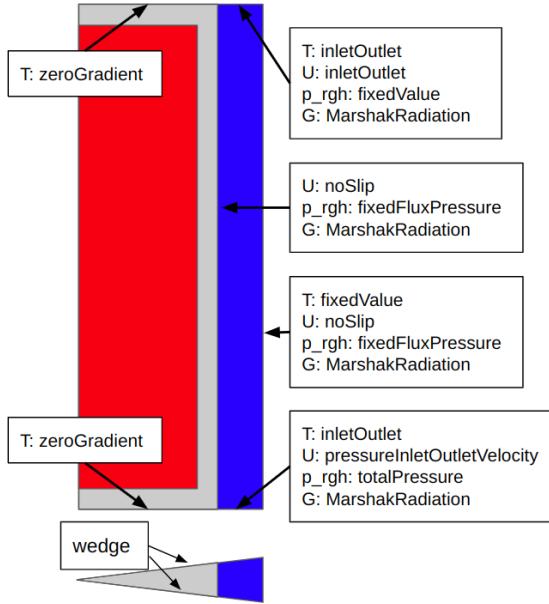


Figure 6.36: Boundary conditions applied to the ATR experiment model in OpenFOAM.

The model considers that the experiment produces volumetric heat defined by the linear combination of three exponential decay functions

$$q'''(t) = q_{1,0}e^{(-t/\tau_1)} + q_{2,0}e^{(-t/\tau_2)} + q_{3,0}e^{(-t/\tau_3)}. \quad (6.48)$$

The source is defined in OpenFOAM using the *fvOptions* file and a *scalarCodedSource* object. This model neglects the heat production ($q''' = 0$) in the capsule to simplify the analysis. This assumption is justified by

the fact that the volume of the experiment is considerably larger than the capsule's.

As mentioned earlier, OpenFOAM supports the parallelization of the simulation via domain decomposition, wherein the mesh and associated fields are separated into pieces and allocated to separate processors for solution. The parallelization involves: decomposition of mesh and fields (*decomposePar* utility), running the application in parallel, and re-composition of the fields (*reconstructPar* utility). This work uses the *scotch* method for the domain decomposition, which requires no geometric input from the user and attempts to minimize the number of boundaries.

To determine the best mesh discretization-number of processors pair, Table 6.7 shows the execution time for simulations running up to 100 second-simulation time with an increasing number of processors and four different meshes with increasing number of cells. For all the meshes, the execution time decreases for an increasing number of processors, up to eight processors. After this number, the execution time increases. For a small number of cells, the communication between processors may dominate the computations, which causes the execution time to increase. Based on these results, the simulations running in parallel use the coarser mesh and eight processors. However, the simulation with eight processors is only 3 times faster than the serial simulation, which means that in cases where the number of processors is limited, it is preferable to run eight simulations in serial simultaneously than eight simulations in parallel subsequently. The parallel runs are reserved for HPC machines where the job scheduling is not based on the number of processors per node but the allocated number of nodes.

Table 6.7: Execution time in seconds for simulations with an increasing number of processors and four different meshes with increasing number of cells.

	Number of processors				
	1	4	8	12	16
A	1591.6	747.21	564.84	583.72	598.51
B	1132.87	961.53	682.86	-	731.33
C	2264.35	1513.1	1092.81	-	1104.77
D	4988.23	2540.45	1982.96	-	2321.46

6.5.1 Results

This section shows the results for two complete cases: first, for an experiment of aluminum, and second, for an experiment of U_3Si_2/Al with a 20% enrichment. Figure 6.37 shows the delayed heating in the experiment of aluminum. The volumetric heat can be expressed as

$$q'''(t)[W/m^3] = 1.954 \times 10^6 e^{(-t/68.649s)} + 4.150 \times 10^5 e^{(-t/68680.862s)} + 6.682 \times 10^5 e^{(-t/3803.589s)}. \quad (6.49)$$

The material properties of aluminum can be found in Table 6.6. Figure 6.38 shows the evolution of the highest temperature in the capsule for the experiment of aluminum. The capsule reaches the maximum temperature of 388.9 K at 580 seconds, and it is located in the capsule's inner wall at the middle of the total height. The temperature exhibits quick and slow responses corresponding to the smallest and largest decay times of the volumetric heat. Hence, the temperature increases rapidly from the initial condition, and after reaching the maximum, it decays slowly.

Figure 6.39 shows the delayed heating in the experiment of U_3Si_2/Al . The volumetric heat can be

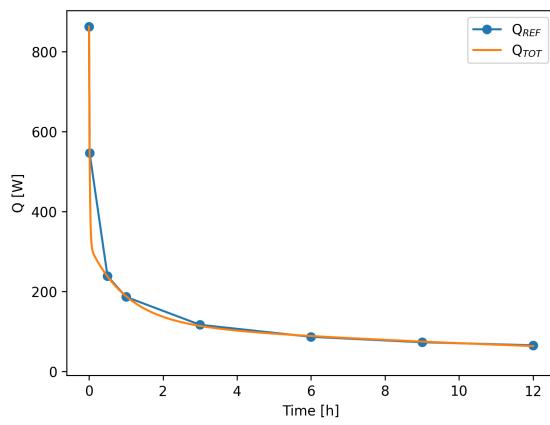


Figure 6.37: Delayed heating over time for ATR experiment of aluminum.

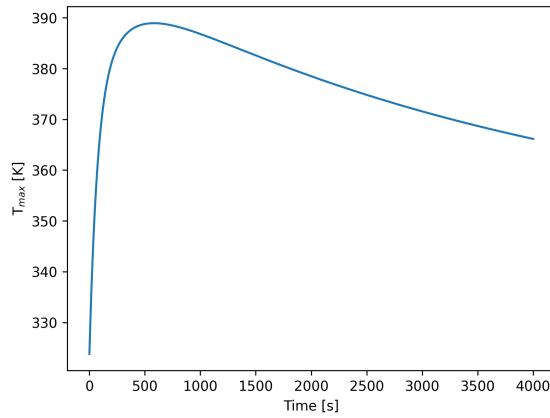


Figure 6.38: Time evolution of the highest temperature in the capsule for ATR experiment of aluminum.

expressed as

$$q'''(t)[W/m^3] = 6.491 \times 10^6 e^{(-t/65308.004s)} + 2.935 \times 10^7 e^{(-t/52.145s)} + 8.485 \times 10^6 e^{(-t/3110.743s)}. \quad (6.50)$$

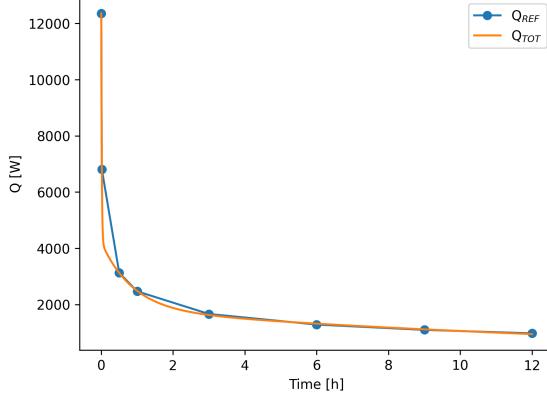


Figure 6.39: Delayed heating over time for ATR experiment of $U_3(20\%)Si_2/Al$.

The material properties of U_3Si_2/Al were calculated using [123] as reference. The molecular weight is 90.32 g/mol, the density is 5660 kg/m³, the heat capacity is 1290 J/kg/K, and the thermal conductivity is 84 W/m/K. Figure 6.40 shows the evolution of the highest temperature in the capsule for the experiment of U_3Si_2/Al . The capsule reaches the maximum temperature of 1047.6 K at 1200 seconds, and it is located in the capsule's inner wall at the middle of the total height. The temperature in the capsule shows a similar behavior than for the experiment of aluminum. The temperature exhibits quick and slow responses, corresponding to the smallest and largest decay times in the volumetric heat. Hence, the temperature increases rapidly from the initial condition, and after reaching the maximum, it decays slowly.

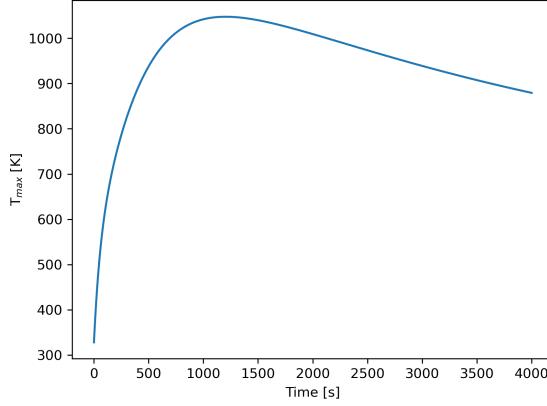


Figure 6.40: Time evolution of the highest temperature in the capsule for ATR experiment of U_3Si_2/Al .

The next step is the creation of the data-based model. This step requires the repeated run of the thermal-fluids model described above to create the training/testing data. The data-based model relies on the classification strategy described in Section 6.3.2 to predict the state of the experiment's capsule. Based on the results from Section 6.4, the features of the model are: ρ , c_p , k , $q_{1,0}$, $q_{2,0}$, $q_{3,0}$, τ_1 , τ_2 , and τ_3 .

The material data (ρ , c_p , k) is generated assuming a multivariate normal distribution centered on real

material data, which is shown in Appendix E. Based on the results from Section 6.3.5, the use of synthetic material data wherein the variables are independent reduces the accuracy of the model. Hence, using a multivariate normal distribution allows to preserve a certain correlation between the thermal properties. The synthetic data is generated using the *multivariate_normal* function from the *NumPy random* package. This function takes a mean value and covariance. The mean value is given by the real material data at which the normal distribution is centered. The covariance is calculated using all the real material data, and it is down-scaled to 1% of the original value to reduce the spread around each sample. Figure 6.41 shows some examples of synthetic material properties, showing the correlation between thermal properties.

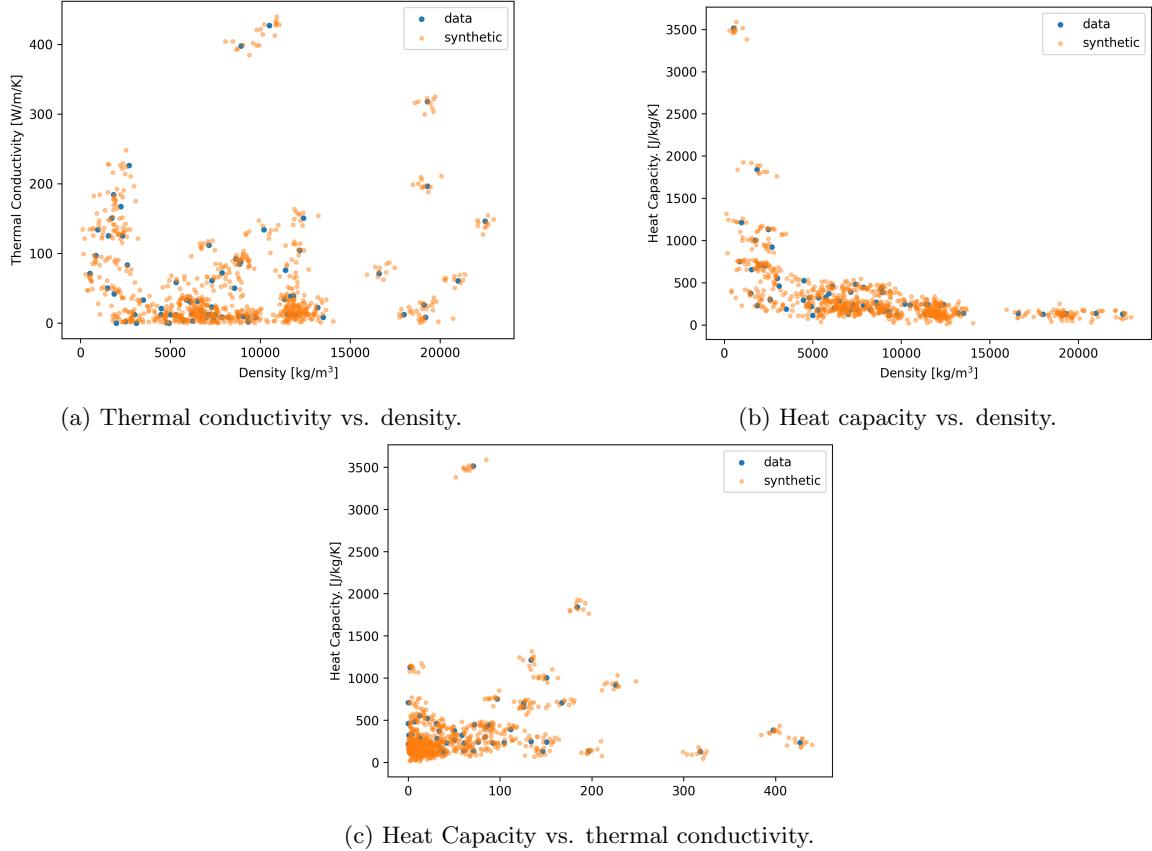


Figure 6.41: Synthetic material properties.

The source data ($q_{i,0}, \tau_i$) is created using the delayed heating workflow on the material experiments shown in Appendix D. Figure 6.42 shows the relationship $q_{i,0}$ vs. τ_i for all samples. The samples exhibit three clusters based on the values of τ_i , and they can be grouped into slow, intermediate, and fast response. The synthetic source data was generated using the *multivariate_normal* function, assuming a normal distribution for $q_{i,0}$ and τ_i centered around each cluster's mean value. Figure 6.43 shows each of the clusters with the original and the generated data.

The combination of the synthetic material properties and source data creates 1227 samples, giving the input data X the shape 1227×9 . The output data is generated by running the thermal-fluids model. The simulation time was two hours (7200 seconds) after the accident. Because the simulations utilized an adaptive time stepper, the outputs were adjusted to give them a fixed length by linear interpolation. The interpolation was calculated for 100 points between 0-500 seconds, and 101 points between 500-7200 seconds. Consequently,

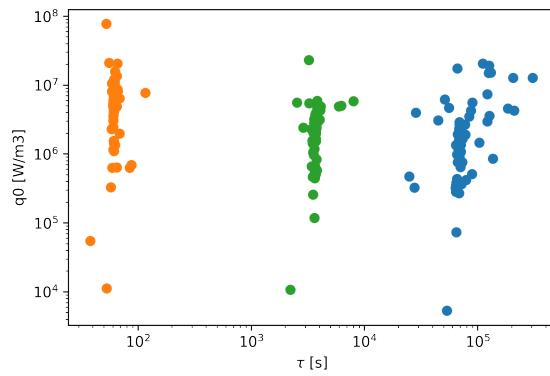


Figure 6.42: Relationship between $q_{i,0}$ and τ_i .

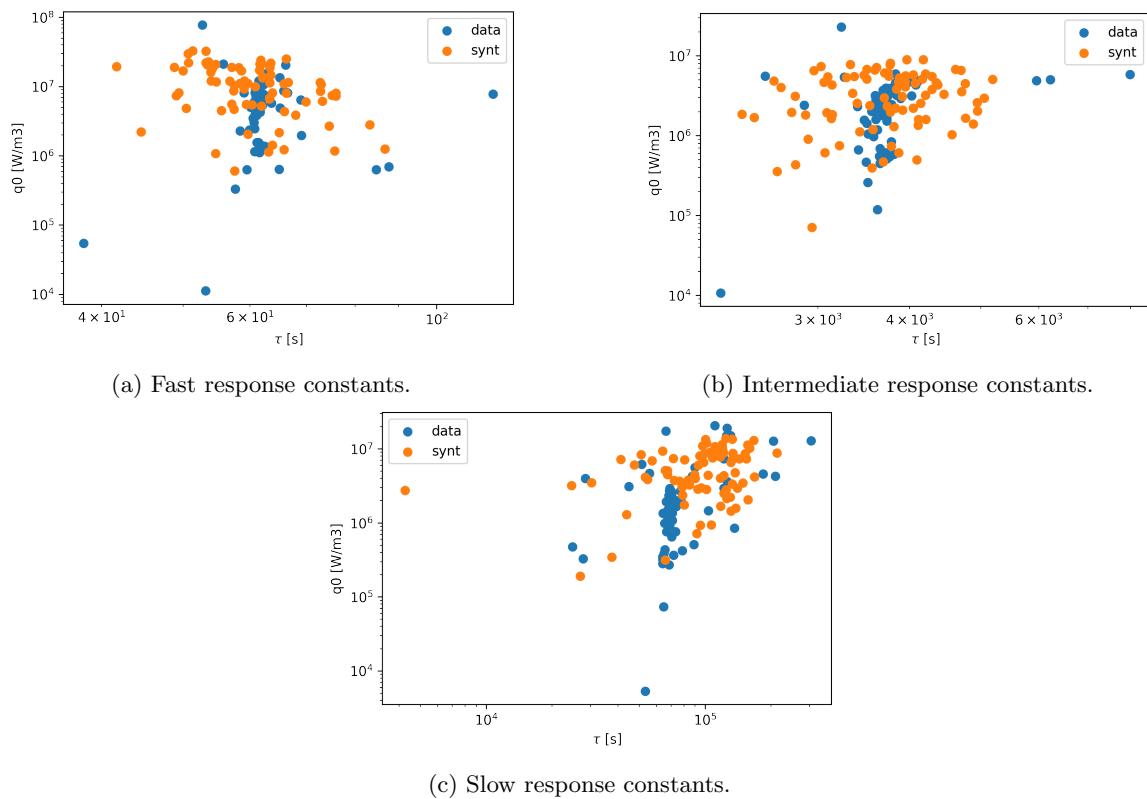


Figure 6.43: Synthetic source data.

the shape of the output data Y was 1227×201 . Finally, the data is prepared using the scheme described in Figure 6.14.

Table 6.8 summarizes the main results of the classification problem for the testing dataset. In terms of accuracy, KNN is the worst-performing method, and DT, FNN, and LogR are the best-performing methods. All the methods show an accuracy higher than 0.95. When focusing on the recall, KNN shows the worst performance with a value below 1. All the rest of the methods show a recall of 1. The last comparison metric is the model training time. All the methods are trained quickly in under 1 second, except for FNN which took 4 seconds (excluding hyperparameter tuning). Although several methods show a recall of 1, high recall is only good in combination with high accuracy. Hence, the best performance is achieved by DT, FNN, and LogR. Finally, Figure 6.44 displays the confusion matrix for the KNN and the FNN cases. The comparison of both results shows a clear improvement from KNN to FNN in the mislabeled samples.

Table 6.8: Performance metrics of the machine learning techniques for the classification problem.

Method	Accuracy	Recall	Train Time [sec]
DT	1.0000	1.0000	<1
FNN	1.0000	1.0000	4
KNN	0.9593	0.9610	<1
LogR	1.0000	1.0000	<1
RF	0.9959	1.0000	<1
SVM	0.9837	1.0000	<1

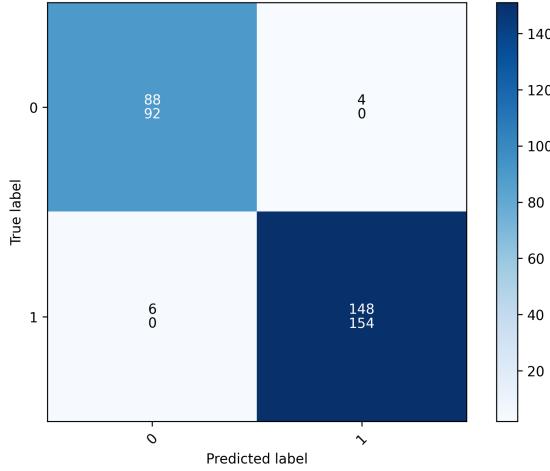


Figure 6.44: Confusion matrix for the KNN (top) and FNN (bottom) cases. The labels 1/0 represent melt/does not melt.

6.6 Conclusions

The prediction of the temperature in selected locations of a nuclear reactor is of paramount importance to ensure their integrity during a transient. This work focuses on the safety analysis of research reactor experiments. This chapter studied the response of a simplified model experiment and an ATR experiment to the channel draining event, where the experiments are left uncovered and the natural circulation of air cools

them down.

Additionally, this chapter investigated the applicability of data-based modeling and machine learning to forecast the state of an experiment's capsule during the accident. The study focused first on the simplified model experiment to guide the choice of a method for the ATR experiment. The study considered two different strategies. The first strategy focused only on answering whether the capsule melted or not. The second strategy added the answer to the question of when the capsule melted. Both strategies obtained satisfactory results in terms of accuracy and recall. The melting time prediction had large maximum relative errors, in the order of 30 to 50%. Nevertheless, the different methods showed low mean relative errors, showing an overall successful performance. In general, this simplified model proved that data-based modeling enables the quick determination of whether the capsule melts or not for any given irradiated material.

Further analysis of the delayed heating results in Chapter 3 showed that the application of the data-based modeling to the ATR experiment required a slight modification to properly capture the heat evolution in the experiment. Consequently, the delayed heating in the ATR experiment was modeled as the linear combination of three exponential decay functions. This chapter also described the assumptions and simplifications considered in the making of the thermal-fluids model of the ATR experiment. Two results were shown. First, for an experiment of aluminum, and second, for an experiment of U_3Si_2/Al . While the first case showed temperatures lower than the melting temperature, the second case showed a temperature exceeding 1000 K, meaning that it would melt.

Finally, the repeated run of the thermal-fluids model with both synthetic material and source data allows to obtain data for the training/testing of a data-based model of the ATR experiment. The results of the testing dataset show high accuracy and recall for most of the machine learning algorithms. These results prove that this work's proposed method relying on data-based modeling decreases the computational expense and enables the fast determination of the accident outcome.

Chapter 7

Conclusions

Nuclear research reactors enable a wide span of applications. Consequently, a vast variety of materials can be inserted in a research reactor to meet the requirements of those applications. These materials are usually contained within experiments, for which the development of safety analyses is often required. Research reactor safety analyses have a strong focus on the reactor core, disregarding a thorough examination of their experiments. However, the creation of a detailed safety analysis preceding the experiment irradiation is necessary to prove compliance with the safety margins in case of an accident. This work investigates detailed methods to support the development of safety analyses for research reactor experiments.

Additionally, as experiments comprise unique combinations of materials, the creation of experiment safety analysis occurs on a case-by-case basis. Such an approach becomes effort-consuming, time-consuming, and error-prone in the absence of a streamlined workflow. Chapter 1 laid out the motivation and objectives of this work and introduced a workflow to streamline the calculations for experiment safety analyses. This work studied the most thermally challenging scenario for an experiment or bounding accident, i.e., a channel draining event. These studies focused on the integrity of the experiment capsule during the accident, for which the creation of a delayed heating calculation workflow and a thermal-fluids model of the experiment were necessary. Furthermore, this work investigated different approaches to decrease the computational burden of the calculations by looking into the construction of a generic material irradiation database and the feasibility of capturing the physics of the accident with data-based models, which rely on several machine learning algorithms.

Chapter 2 summarized previous work in delayed heating calculations, transport/depletion solvers, and machine learning applied to nuclear engineering. The survey concluded that the formal 3-step process is the only method that can fulfill the objectives of this work. Two main reasons justify this conclusion. First, the 3-step process has the flexibility to consider any photon sources, those being either fission or activation products. Second, this work calculates the delayed heating evolution throughout time after the reactor shutdown, and other methods do not have such a capability.

Regarding the transport/depletion solvers, Chapter 2 described several software packages developed in the past 25 years with such capabilities. In general, the objectives of this work require a tool relying on the CRAM solver, which can be applied to ATR, which is actively maintained, and with relatively low computational requirements. The survey concluded that the best fit for these requirements was MOAA.

Finally, the examination of previous work implementing machine learning to nuclear engineering problems gave some broad understanding of the state-of-the-art of the field. The examination concluded that different

machine learning algorithms have been applied to nuclear engineering to some extent, and although several similar studies have been conducted using machine learning, none had comparable objectives to our case. Nevertheless, this examination allowed to find some algorithms compatible with these work objectives.

Chapter 3 defined the methodology to streamline the calculation of the delayed heating in any reactor location, including reactor structures and reactor experiments. The chapter described all the developed and utilized computational tools necessary to create the workflow. Overall, MOAA introduces a tool that controls MCNP tally card writing, reads MCNP tally information, automates ORIGEN input file creation, executes SCALE, and standardizes results post-processing. This work establishes a delayed heating calculation workflow based on the formal three-step method that builds on top of MOAA.

This chapter demonstrated the delayed heating calculation capabilities by presenting four exercises. Some of these exercises had the purpose to verify, demonstrate, and ease the visualization of the workflow. Two of the exercises demonstrated the use of the delayed heating calculation workflow on full-size research reactors, wherein the focus of the calculations was the heat in the reactor structures and the heat in an experiment. In general, the developments presented in this chapter meet this thesis objective of streamlining the calculation of delayed heating in experiments, which provides a volumetric heat source term to the thermal-fluids models introduced in the following chapters.

Chapter 4 builds on top of the developments from Chapter 3 and introduces a shutdown-dose rate calculation workflow as a value-added. After an experiment was irradiated in a research reactor, the ongoing decay exposes the reactor surroundings, including workers and equipment. Hence, the determination of the shutdown dose rates is required in safety analyses.

This chapter presented the shutdown dose rates of the AGR-1 experiment, which helped guide the PIE of the experiment. The AGR-1 experiment was a TRISO-fueled experiment, and the workflow introduced here is also applicable to TRISO-fueled reactors, such as HTGRs. This chapter also demonstrated the shutdown dose rate calculation in a high-temperature gas-cooled microreactor, relying on the explicit modeling of the TRISO particles and the repeated structures of MCNP. Overall, the calculation workflow introduced in this chapter meets this thesis objective of creating a shutdown-dose rate calculation workflow and removes the need for the creation of an additional workflow.

Chapter 5 investigates different methods using previous knowledge to accelerate the calculation of delayed heating in experiments. The first method investigates the feasibility of reducing the calculation burden by solving the neutron-transport problem only once. This method relies on the delayed heating calculation for an experiment of carbon and uses the geometry-dependent parameters from the neutron transport-problem to solve the depletion problem of experiments of arbitrary material composition. The results showed that this method is overly conservative, and the following sections introduce another method relying on a generic irradiation database.

The generic irradiation database method relies on the delayed heating calculation workflow to obtain the heat produced in experiments of individual chemical elements and through their combination calculate the heat produced in experiments of any material composition. This chapter verified the method by presenting the results of a vast range of materials that could be introduced in a research reactor for study. The verification of the method was done for a simple demonstration exercise and an ATR experiment, for which the delayed heating over time was also verified. Overall, the generic material irradiation database introduced in this chapter meets this thesis objective of accelerating the calculation of delayed heating in experiments of any material composition.

Chapter 6 investigated the applicability of machine learning algorithms to predict the state of the

experiment capsule during a channel draining event. Creating a data-based model allows to quickly determine whether the capsule melts for any given irradiated material. The chapter also describes the thermal-fluids solvers and the solved equations that calculate the temperature in the experiment, capsule, and surrounding air. Moreover, the chapter explains the different machine learning algorithms utilized to build the data-based models in the following studies.

The data-based modeling was demonstrated for a simplified thermal-fluids model. Such a model consisted of a simple two-dimensional conjugate heat transfer problem representing an experimental device after irradiation and cooled by the natural circulation of air. Additionally, this chapter summarizes the data preparation process necessary for the training and testing of the data-based model. The analysis considered two strategies: one using classification to predict whether the capsule melts or not, and another one predicting the same outcome as well as the time of occurrence. The main results compared the performance of the different methods, concluding that the machine learning algorithms were able to capture the physics of the problem and accurately predict the state of the capsule during the accident.

Moreover, the slight modification of the previous data-based model allowed for the modeling of a full-size research reactor experiment, an ATR experiment. The repeated run of the thermal-fluids model with both synthetic material and source data generated output data for the training/testing of the data-based model. The results presented in the chapter corroborate that data-based modeling of research reactor experiments is technically feasible.

Overall, the developments presented in this chapter meet this thesis objective of creating a CFD model of the experiment, and it was demonstrated for the case of the ATR experiment. Additionally, this chapter also meets this thesis objective of building a data-based model of a research reactor experiment, and it was also demonstrated for the ATR experiment.

In conclusion, this thesis introduces a workflow to streamline the delayed heating and shutdown-dose rate calculations to support the development of safety analysis for research reactor experiments. Additionally, this thesis developed a method to quickly calculate the delayed heating in experiments of arbitrary material composition. Finally, the determination of the capsule's integrity can be quickly performed by training a data-based model of the experiment. Altogether, the results presented in this thesis prove that the proposed methodology decreases the computational burden and enables quick calculations for research reactor experiments.

References

- [1] “Research reactors: Purpose and future,” *IAEA*, 2016.
- [2] H. Blaumann, A. Vertullo, F. Sanchez, F. Brollo, and J. Longhino, *RA-10: A New Argentinian Multipurpose Research Reactor*. International Atomic Energy Agency (IAEA): IAEA, 2012, ISBN: 1991-2374. [Online]. Available: http://inis.iaea.org/search/search.aspx?orig_q=RN:44019508.
- [3] J. W. Sterbentz and J. J. Cogliati, “AGR-1 Depletion Benchmark,” English, United States, Tech. Rep., Jul. 2018. DOI: [10.2172/1475404](https://doi.org/10.2172/1475404).
- [4] E. Mattar and N. Jawerth, “Exploring research reactors and their use,” English, *IAEA Bulletin*, vol. Vol. 60-4, pp. 4–5, Nov. 2019. [Online]. Available: <https://www.iaea.org/bulletin/research-reactors/exploring-research-reactors-and-their-use>.
- [5] A. M. Johnsen and K. Ünlü, “Neutron activation analysis capabilities and applications at the Penn State Radiation Science and Engineering Center,” *Forensic Chemistry*, vol. 7, pp. 56–64, 2018, ISSN: 2468-1709. DOI: [10.1016/j.forc.2017.12.004](https://doi.org/10.1016/j.forc.2017.12.004).
- [6] *Technetium-99m Radiopharmaceuticals: Status and Trends*, ser. Radioisotopes and Radiopharmaceuticals Series 1. Vienna: INTERNATIONAL ATOMIC ENERGY AGENCY, 2010, ISBN: 978-92-0-103509-7. [Online]. Available: <https://www.iaea.org/publications/8110/technetium-99m-radiopharmaceuticals-status-and-trends>.
- [7] *Safety Assessment for Research Reactors and Preparation of the Safety Analysis Report*, ser. Specific Safety Guides SSG-20 (Rev. 1). Vienna: INTERNATIONAL ATOMIC ENERGY AGENCY, 2022, ISBN: 978-92-0-141521-9. [Online]. Available: <https://www.iaea.org/publications/14812/safety-assessment-for-research-reactors-and-preparation-of-the-safety-analysis-report>.
- [8] *Safety in the Utilization and Modification of Research Reactors*, ser. Specific Safety Guides SSG-24. Vienna: INTERNATIONAL ATOMIC ENERGY AGENCY, 2012, ISBN: 978-92-0-129110-3. [Online]. Available: <https://www.iaea.org/publications/8854/safety-in-the-utilization-and-modification-of-research-reactors>.
- [9] INL, “Advanced Test Reactor National Scientific User Facility Users’ Guide,” INL, Technical Report INL/EXT-08-14709, 2009.
- [10] “High Flux Isotope Reactor (HFIR): User Guide,” Oak Ridge National Laboratory, Technical Report NSFU Rev. 1.0, May 2011.
- [11] T. Bajs, D. Grgic, and N. Cavlina, “LOCA analysis of the IRIS reactor,” Slovenia: Nuclear Society of Slovenia, 2003, p. 827, ISBN: 961-6207-21-0. [Online]. Available: http://inis.iaea.org/search/search.aspx?orig_q=RN:36115766.

- [12] H. Kazeminejad, “Thermal-hydraulic modeling of flow inversion in a research reactor,” *Annals of Nuclear Energy*, vol. 35, no. 10, pp. 1813–1819, 2008, ISSN: 0306-4549. DOI: [10.1016/j.anucene.2008.05.006](https://doi.org/10.1016/j.anucene.2008.05.006).
- [13] A. Hainoun, A. Doval, P. Umbehaun, S. Chatzidakis, N. Ghazi, S. Park, M. Mladin, and A. Shokr, “International benchmark study of advanced thermal hydraulic safety analysis codes against measurements on IEA-R1 research reactor,” *Nuclear Engineering and Design*, vol. 280, pp. 233–250, 2014, ISSN: 0029-5493. DOI: [10.1016/j.nucengdes.2014.06.041](https://doi.org/10.1016/j.nucengdes.2014.06.041).
- [14] A. Abdelmaksoud, S. Haggag, M. M. Zaky, and M. Osman, “Analysis of a Hypothetical Complete Loss-of-Coolant Accident in a Typical Research Reactor,” *Nuclear Technology*, vol. 0, no. 0, pp. 1–13, 2022. DOI: [10.1080/00295450.2022.2035644](https://doi.org/10.1080/00295450.2022.2035644).
- [15] “Decay Heat Power in Light Water Reactors - American National Standard,” American Nuclear Society, Tech. Rep. ANSI/ANS-5.1-1994, 1994.
- [16] L. Giot, *Decay Heat calculations with SERPENT 2*, Presentation, Espoo, Finland, 2018. [Online]. Available: http://montecarlo.vtt.fi/mtg/2018_Espoo/Giot1.pdf.
- [17] “SCALE Code System, ORNL/TM-2005/39, Version 6.2.3,” Oak Ridge National Laboratory, Oak Ridge, Tennessee, USA, Technical Report ORNL/TM-2005/39, 2018.
- [18] H. C. Lee, C. K. Jo, H. J. Shim, Y. Kim, and J. M. Noh, “Decay heat analysis of VHTR cores by Monte Carlo core depletion calculation,” *Annals of Nuclear Energy*, vol. 37, no. 10, pp. 1356–1368, 2010, ISSN: 0306-4549. DOI: [10.1016/j.anucene.2010.05.011](https://doi.org/10.1016/j.anucene.2010.05.011).
- [19] G. Ilas, D. Chandler, B. J. Ade, E. E. Sunny, B. R. Betzler, and D. Pinkston, “Modeling and simulations for the High Flux Isotope Reactor Cycle 400,” Technical Report ORNL/TM-2015/36, Mar. 2015. DOI: [10.2172/1185903](https://doi.org/10.2172/1185903).
- [20] *Sensitivity Evaluation of the Daily Thermal Predictions of the AGR-2 Experiment in the Advanced Test Reactor*, vol. ASME 2015 Nuclear Forum, International Conference on Nuclear Engineering, V001T04A002, Jun. 2015. DOI: [10.1115/NUCLRF2015-49698](https://doi.org/10.1115/NUCLRF2015-49698).
- [21] M. Lemaire, C. Vaglio-Gaudard, A. Lyoussi, and C. Reynard-Carette, “For a better estimation of gamma heating in nuclear material-testing reactors and associated devices: Status and work plan from calculation methods to nuclear data,” *Journal of Nuclear Science and Technology*, vol. 52, no. 9, pp. 1093–1101, 2015. DOI: [10.1080/00223131.2015.1009957](https://doi.org/10.1080/00223131.2015.1009957).
- [22] J. L. Peterson-Droogh and R. H. Howard, “Current Neutronic Calculation Techniques for Modeling the Production of Ir-192 in HFIR,” English, Research Org.: Oak Ridge National Lab. (ORNL), Oak Ridge, TN (United States), Jun. 2018. [Online]. Available: <https://www.osti.gov/biblio/1464014>.
- [23] Y. Chen and U. Fischer, “Rigorous mcnp based shutdown dose rate calculations: Computational scheme, verification calculations and application to ITER,” en, *Fusion Engineering and Design*, vol. 63-64, pp. 107–114, Dec. 2002, ISSN: 09203796. DOI: [10.1016/S0920-3796\(02\)00144-8](https://doi.org/10.1016/S0920-3796(02)00144-8).
- [24] R. G. Ambrosek and G. S. Chang, “Improved methodology for temperature predictions in advanced reactors,” Oct. 1995. [Online]. Available: <https://www.osti.gov/biblio/116642>.
- [25] Y.-K. Lee and K. Sharma, “Tripoli-4 Gamma-Ray Dose Calculation for Spent PWR Fuels,” vol. 1, Jul. 2013. DOI: [10.1115/ICONE21-15498](https://doi.org/10.1115/ICONE21-15498).

- [26] V. Mozin, “Delayed Gamma-Ray Assay for Nuclear Safeguards,” PhD Dissertation, University of California, Berkeley, Berkeley, CA, US, 2011. [Online]. Available: https://escholarship.org/content/qt5mm1v58k/qt5mm1v58k_noSplash_ca99c596dd23532d4de40c098aa706cd.pdf.
- [27] “MCNP Users Manual - Code Version 6.2,” Los Alamos National Laboratory, Los Alamos, New Mexico, USA, Technical Report LA-UR-17-29981, 2017.
- [28] M. L. Fensin, J. S. Hendricks, and S. Anghaie, “The Enhancements and Testing for the MCNPX 2.6.0 Depletion Capability,” *Nuclear Technology*, vol. 170, no. 1, pp. 68–79, 2010, Publisher: Taylor & Francis. DOI: [10.13182/NT10-2](https://doi.org/10.13182/NT10-2).
- [29] E. Brun *et al.*, “Tripoli-4, CEA, EDF and AREVA reference Monte Carlo code,” *Annals of Nuclear Energy*, vol. 82, pp. 151–160, 2015, Joint International Conference on Supercomputing in Nuclear Applications and Monte Carlo 2013, SNA + MC 2013. Pluri- and Trans-disciplinarity, Towards New Modeling and Numerical Simulation Paradigms, ISSN: 0306-4549. DOI: [10.1016/j.anucene.2014.07.053](https://doi.org/10.1016/j.anucene.2014.07.053).
- [30] A. Tsilanizara *et al.*, “Darwin: An Evolution Code System for a Large Range of Applications,” *Journal of Nuclear Science and Technology*, vol. 37, no. sup1, pp. 845–849, 2000. DOI: [10.1080/00223131.2000.10875009](https://doi.org/10.1080/00223131.2000.10875009).
- [31] T.-y. Noh, B.-G. Park, and M.-S. Kim, “Estimation of nuclear heating by delayed gamma rays from radioactive structural materials of HANARO,” en, *Nuclear Engineering and Technology*, vol. 50, no. 3, pp. 446–452, Apr. 2018, ISSN: 17385733. DOI: [10.1016/j.net.2018.01.010](https://doi.org/10.1016/j.net.2018.01.010).
- [32] R. Forrest and J. Sublet, “FISPACT-99 User manual,” United Kingdom, Tech. Rep., 1998, UKAEA-FUS-407, p. 201. [Online]. Available: http://inis.iaea.org/search/search.aspx?orig_q=RN:31048113.
- [33] S. Sheludjakov and A. Serikov, “The advanced 3D method for activation analysis of fusion reactor materials,” *Fusion Engineering and Design - FUSION ENG DES*, vol. 63, pp. 487–492, Dec. 2002. DOI: [10.1016/S0920-3796\(02\)00183-7](https://doi.org/10.1016/S0920-3796(02)00183-7).
- [34] P. Sauvan *et al.*, “Development of the R2SUNED Code System for Shutdown Dose Rate Calculations,” *IEEE Transactions on Nuclear Science*, vol. 63, no. 1, pp. 375–384, 2016. DOI: [10.1109/TNS.2015.2507138](https://doi.org/10.1109/TNS.2015.2507138).
- [35] D. Valenza, H. Iida, R. Plenteda, and R. T. Santoro, “Proposal of shutdown dose estimation method by Monte Carlo code,” *Fusion Engineering and Design*, vol. 55, no. 4, pp. 411–418, 2001, ISSN: 0920-3796. DOI: [10.1016/S0920-3796\(01\)00188-0](https://doi.org/10.1016/S0920-3796(01)00188-0).
- [36] L. Petrizzi, H. Iida, D. Valenza, and P. Batistoni, “Improvement and Benchmarking of the New Shutdown Dose Estimation Method by Monte Carlo Code,” in *Advanced Monte Carlo for Radiation Physics, Particle Transport Simulation and Applications*, A. Kling, F. J. C. Barão, M. Nakagawa, L. Távora, and P. Vaz, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 865–870, ISBN: 978-3-642-18211-2.
- [37] I. Palermo, R. Villari, and A. Ibarra, “Shutdown dose rate assessment with the Advanced D1S method for the european DCLL DEMO,” *Fusion Engineering and Design*, vol. 122, pp. 163–175, 2017, ISSN: 0920-3796. DOI: [10.1016/j.fusengdes.2017.08.021](https://doi.org/10.1016/j.fusengdes.2017.08.021).
- [38] B. Lee, “Analysis on the Detailed Heat Generation Rate for the HANARO,” Tech. Rep., 2008, KAERI/TR-3643/2008.

- [39] F. B. Brown, W. R. Martin, and R. D. Mosteller, *Monte Carlo - Advances and Challenges*, Interlaken, Switzerland, Sep. 2008. [Online]. Available: https://mcnp.lanl.gov/pdf_files/la-ur-08-5891.pdf.
- [40] D. Ilas, “Impact of HFIR LEU Conversion on Beryllium Reflector Degradation Factors,” Oak Ridge National Laboratory, Oak Ridge, TN, USA, ORNL/TM-2013/441, Oct. 2013. [Online]. Available: <https://www.osti.gov/biblio/1096323>.
- [41] E. E. Davidson, B. R. Betzler, D. Chandler, and G. Ilas, “Heat deposition analysis for the High Flux Isotope Reactor’s HEU and LEU core models,” *Nuclear Engineering and Design*, vol. 322, Aug. 2017. DOI: [10.1016/j.nucengdes.2017.06.040](https://doi.org/10.1016/j.nucengdes.2017.06.040).
- [42] L. Jurbandam, “Calculation of the Fission Q-value and Spatial Energy Deposition in the SAFARI-1 Nuclear Reactor,” Master’s thesis, University of the Witwatersrand, Johannesburg, South Africa, 2018.
- [43] R. Moore, B. Schnitzler, C. Wemple, R. Babcock, and D. Wessol, “MOCUP: MCNP-ORIGEN2 Coupled Utility Program,” Lockheed Martin Idaho Technologies/Idaho National Engineering Laboratory, Technical Report, 1995.
- [44] H. Trellue, “Development of Monteburns: A Code that Links MCNP and ORGEN2 in an Automated Fashion for Burnup Calculations,” Master’s thesis, University of New Mexico, Albuquerque, New Mexico, USA, Dec. 1998.
- [45] J. Leppänen, M. Pusa, T. Viitanen, V. Valtavirta, and T. Kaltiaisenaho, “The Serpent Monte Carlo code: Status, development and applications in 2013,” *Annals of Nuclear Energy*, vol. 82, pp. 142–150, 2015, Joint International Conference on Supercomputing in Nuclear Applications and Monte Carlo 2013, SNA + MC 2013. Pluri- and Trans-disciplinarity, Towards New Modeling and Numerical Simulation Paradigms, ISSN: 0306-4549. DOI: [10.1016/j.anucene.2014.08.024](https://doi.org/10.1016/j.anucene.2014.08.024).
- [46] Z. Xu and P. Hejzlar, “MCODE, Version 2.2: An MCNP-ORIGEN DEpletion Program,” Massachusetts Institute of Technology/Center for Advanced Nuclear Energy Systems, Technical Report MIT-NFC; TR-104, Dec. 2008.
- [47] V. Haeck, *Monte Carlo Depletion Calculation using VESTA 2.1*, Technical Presentation, IRSN, 2012.
- [48] M. Pusa, “Numerical methods for nuclear fuel burnup calculations,” English, ISBN: 978-951-38-7999-0 Issue: 32 Series: VTT Science, PhD Thesis, Aalto University, 2013.
- [49] J. Park, A. Khassenov, W. Kim, S. Choi, and D. Lee, “Comparative analysis of VERA depletion benchmark through consistent code-to-code comparison,” *Annals of Nuclear Energy (Oxford)*, pp. 385–398, 2019, Place: United Kingdom, ISSN: 0306-4549. [Online]. Available: http://inis.iaea.org/search/search.aspx?orig_q=RN:50079636.
- [50] A. G. Croff, “ORIGEN2: A revised and updated version of the Oak Ridge isotope generation and depletion code,” English, Oak Ridge National Laboratory, United States, Tech. Rep. ORNL-5621, TRN: 80-012377, Jul. 1980. DOI: [10.2172/5352089](https://doi.org/10.2172/5352089).
- [51] A. E. Isotalo and W. A. Wieselquist, “A method for including external feed in depletion calculations with CRAM and implementation into ORIGEN,” English, *Annals of Nuclear Energy (Oxford)*, May 2015, Place: United States, Publisher: Elsevier. DOI: [10.1016/j.anucene.2015.04.037](https://doi.org/10.1016/j.anucene.2015.04.037).
- [52] F. Marshall, “Advanced Test Reactor Capabilities and Future Operating Plans,” in *Proceedings of the Americas Nuclear Energy Symposium 2002*, Oct. 2002.

- [53] R. E. Fairhurst-Agosta, A. L. Carter, and J. L. Peterson-Droogh, “Development of the MCNP-ORIGEN Activation Automation tool,” in *PHYSOR 2022*, Pittsburgh, PA, USA: ANS, May 2022.
- [54] M. G. Na, W. S. Park, and D. H. Lim, “Detection and Diagnostics of Loss of Coolant Accidents Using Support Vector Machines,” *IEEE Transactions on Nuclear Science*, vol. 55, no. 1, pp. 628–636, 2008. DOI: [10.1109/TNS.2007.911136](https://doi.org/10.1109/TNS.2007.911136).
- [55] I. Bae, M. Na, Y. Lee, and G. Park, “Estimation of the Power Peaking Factor in a Nuclear Reactor Using Support Vector Machines and Uncertainty Analysis,” *NUCLEAR ENGINEERING AND TECHNOLOGY*, vol. 41, no. 9, Nov. 2009.
- [56] Y. Zeng, J. Liu, K. Sun, and L.-w. Hu, “Machine learning based system performance prediction model for reactor control,” en, *Annals of Nuclear Energy*, vol. 113, pp. 270–278, Nov. 2017, ISSN: 03064549. DOI: [10.1016/j.anucene.2017.11.014](https://doi.org/10.1016/j.anucene.2017.11.014).
- [57] H. A. Gohel, H. Upadhyay, L. Lagos, K. Cooper, and A. Sanzetenea, “Predictive maintenance architecture development for nuclear infrastructure using machine learning,” en, *Nuclear Engineering and Technology*, vol. 52, no. 7, pp. 1436–1442, Jul. 2020, ISSN: 17385733. DOI: [10.1016/j.net.2019.12.029](https://doi.org/10.1016/j.net.2019.12.029).
- [58] J. Blevins and G. Yang, “Machine learning enabled advanced manufacturing in nuclear engineering applications,” en, *Nuclear Engineering and Design*, vol. 367, p. 110817, Oct. 2020, ISSN: 00295493. DOI: [10.1016/j.nucengdes.2020.110817](https://doi.org/10.1016/j.nucengdes.2020.110817).
- [59] Z. Guo and R. E. Uhrig, “Use of Artificial Neural Networks to Analyze Nuclear Power Plant Performance,” *Nuclear Technology*, Jan. 1992. DOI: [10.13182/NT92-A34701](https://doi.org/10.13182/NT92-A34701).
- [60] A. Erdogan and G. Melih, “A PWR reload optimisation code (XCore) using artificial neural networks and genetic algorithms,” *Annals of Nuclear Energy*, 2003.
- [61] T. Andersson, C. Demazière, A. Nagy, U. Sandberg, N. S. Garis, and I. Pázsit, “Development and application of core diagnostics and monitoring for the Ringhals PWRS,” *Progress in Nuclear Energy*, vol. 43, no. 1, pp. 35–41, 2003, ISSN: 0149-1970. DOI: [10.1016/S0149-1970\(03\)00006-4](https://doi.org/10.1016/S0149-1970(03)00006-4).
- [62] J. L. Montes, J. L. François, J. J. Ortiz, C. Martín-del-Campo, and R. Perusquía, “Local power peaking factor estimation in nuclear fuel by artificial neural networks,” en, *Annals of Nuclear Energy*, vol. 36, no. 1, pp. 121–130, Jan. 2009, ISSN: 03064549. DOI: [10.1016/j.anucene.2008.09.011](https://doi.org/10.1016/j.anucene.2008.09.011).
- [63] M. Gomez Fernandez, A. Tokuhiro, K. Welter, and Q. Wu, “Nuclear energy system’s behavior and decision making using machine learning,” en, *Nuclear Engineering and Design*, vol. 324, pp. 27–34, Dec. 2017, ISSN: 00295493. DOI: [10.1016/j.nucengdes.2017.08.020](https://doi.org/10.1016/j.nucengdes.2017.08.020).
- [64] M. Kamuda, J. Zhao, and K. Huff, “A Comparison of Machine Learning Methods for Automated Gamma-Ray Spectroscopy,” *Nuclear Instruments and Methods in Physics Research, Section A*, Jul. 2018. DOI: [10.1016/j.nima.2018.10.063](https://doi.org/10.1016/j.nima.2018.10.063).
- [65] J. Wilson, “Machine Learning for Nuclear Fission Systems: Preliminary Investigation of an Autonomous Control System for the MGEP,” en, Bachelor Thesis, MIT, Boston, MA, Jun. 2019.
- [66] “Recurrent neural network based language model,” in *Eleventh annual conference of the international speech communication association*, 2010.

- [67] J. Yang and J. Kim, “An accident diagnosis algorithm using long short-term memory,” en, *Nuclear Engineering and Technology*, vol. 50, no. 4, pp. 582–588, May 2018, ISSN: 17385733. DOI: [10.1016/j.net.2018.03.010](https://doi.org/10.1016/j.net.2018.03.010).
- [68] D. Lee, P. H. Seong, and J. Kim, “Autonomous operation algorithm for safety systems of nuclear power plants by using long-short term memory and function-based hierarchical framework,” en, *Annals of Nuclear Energy*, vol. 119, pp. 287–299, Sep. 2018, ISSN: 03064549. DOI: [10.1016/j.anucene.2018.05.020](https://doi.org/10.1016/j.anucene.2018.05.020).
- [69] M. I. Radaideh, C. Pigg, T. Kozlowski, Y. Deng, and A. Qu, “Neural-based time series forecasting of loss of coolant accidents in nuclear power plants,” en, *Expert Systems with Applications*, vol. 160, p. 113 699, Dec. 2020, ISSN: 09574174. DOI: [10.1016/j.eswa.2020.113699](https://doi.org/10.1016/j.eswa.2020.113699).
- [70] A. SMAIZYS, E. NARKUNAS, and P. POSKAS, “Criticality and Dose Rate Modelling for Copper Disposal Canister with Spent RBMK1500 Nuclear Fuel,” in *Proceedings of the International Conference Underground Disposal Unit Design and Emplacement Processes for a Deep Geological Repository*, Prague, Czech Rep., 2008.
- [71] A. Mohammadi, M. Hassanzadeh, and M. Gharib, “Shielding calculation and criticality safety analysis of spent fuel transportation cask in research reactors,” *Applied Radiation and Isotopes*, vol. 108, pp. 129–132, 2016, ISSN: 0969-8043. DOI: [10.1016/j.apradiso.2015.12.045](https://doi.org/10.1016/j.apradiso.2015.12.045).
- [72] R. Fairhurst-Agosta and T. Kozlowski, “Decay heat calculations using MCNP and ORIGEN,” in *ANS Student Conference 2022*, Urbana-Champaign, IL: ANS, Apr. 2022.
- [73] ——, “Demonstration of Decay Heat Calculations using the MCNP-ORIGEN Activation Automation Tool,” in *International Conference on Engineering Computational Technology*, Montpellier, France, Aug. 2022.
- [74] ——, “Database Construction for Nuclear Heating Calculations,” in *ANS Winter Meeting 2022*, Phoenix, AZ: ANS, Nov. 2022.
- [75] J. Leppanen, *Burnup calculation methodology in Serpent*, Technical Presentation, Mar. 2012. [Online]. Available: http://montecarlo.vtt.fi/download/Serpent2_BU.pdf.
- [76] J. Bryant and N. Wilburn, “Handbook of software quality assurance techniques applicable to the nuclear industry,” NRC/Pacific Northwest Laboratory, USA, Technical Report NUREG/CR-4640, PNL-5784, Aug. 1987, p. 111.
- [77] ASME, “Quality Assurance Requirements for Nuclear Facility Applications,” Technical Report ASME NQA-1-2008, Mar. 2008.
- [78] A. Scopatz and K. Huff, *Effective Computation in Physics*, First Edition. O’Reilly Media, Inc, Jun. 2015, ISBN: 978-1-4919-0153-3. [Online]. Available: <https://www.oreilly.com/library/view/effective-computation-in/9781491901564/>.
- [79] R. Fairhurst-Agosta and T. Kozlowski, “Machine Learning Classification for Safety Analysis of Nuclear Research Reactor Experiments,” in *ANS Student Conference 2022*, Urbana-Champaign, IL, USA: ANS, Apr. 2022.
- [80] ——, “Analysis of Nuclear Research Reactor Experiments using Machine Learning,” in *ANS Annual Meeting 2022*, CA, USA: ANS, Apr. 2022.

- [81] A. Trkov, M. Herman, and D. A. Brown, “Endf-6 Formats Manual: Data Formats and Procedures for the Evaluated Nuclear Data Files ENDF/B-VI, ENDF/B-VII and ENDF/B-VIII,” Brookhaven National Laboratory, Upton, NY, USA, Technical Report BNL-203218-2018-INRE, 2018.
- [82] W. P. on Nuclear Criticality Safety, “International Criticality Safety Benchmark Evaluation Project,” Organisation for Economic Co-Operation and Development, Nuclear Energy Agency, Technical Report NEA-7231, 2014.
- [83] T. A. Tomberlin, “Advanced Test Reactor (ATR) Facility 10CFR830 Safety Basis Related to Facility Experiments,” English, Research Org.: Idaho National Lab. (INL), Idaho Falls, ID (United States), Jun. 2002. [Online]. Available: <https://www.osti.gov/biblio/911330>.
- [84] H. Q. Ho, T. Ishii, S. Nagasumi, M. Ono, Y. Shimazaki, E. Ishitsuka, M. Goto, I. L. Simanullang, N. Fujimoto, and K. Iigaki, “Calculation of shutdown gamma distribution in the high temperature engineering test reactor,” *Nuclear Engineering and Design*, vol. 396, p. 111913, 2022, ISSN: 0029-5493. DOI: [10.1016/j.nucengdes.2022.111913](https://doi.org/10.1016/j.nucengdes.2022.111913).
- [85] R. Abrefah, P. Essel, and H. Odoi, “Estimation of the dose rate of nuclear fuel of Ghana Research Reactor-1 (GHARR-1) using ORIGEN-S and MCNP 6,” *Progress in Nuclear Energy*, vol. 105, pp. 309–317, 2018, ISSN: 0149-1970. DOI: [10.1016/j.pnucene.2018.02.002](https://doi.org/10.1016/j.pnucene.2018.02.002).
- [86] K. Ambrožič and L. Snoj, “JSIR2S code for delayed radiation simulations: Validation against measurements at the JSI TRIGA reactor,” *Progress in Nuclear Energy*, vol. 129, p. 103498, 2020, ISSN: 0149-1970. DOI: [10.1016/j.pnucene.2020.103498](https://doi.org/10.1016/j.pnucene.2020.103498).
- [87] S. Bagheri, H. Khalafi, F. Faghihi, A. Ezzati, M. Keyvani, and H. Ghods, “Gamma dose rate determination of TRR irradiated fuel assemblies,” *Progress in Nuclear Energy*, vol. 142, p. 103950, 2021, ISSN: 0149-1970. DOI: [10.1016/j.pnucene.2021.103950](https://doi.org/10.1016/j.pnucene.2021.103950).
- [88] Suwoto, H. Adrial, A. Hamzah, Zuhair, S. Bakhri, and G. R. Sunaryo, “Neutron dose rate analysis on HTGR-10 reactor using Monte Carlo code,” *Journal of Physics: Conference Series*, vol. 962, no. 1, p. 012029, Feb. 2018. DOI: [10.1088/1742-6596/962/1/012029](https://doi.org/10.1088/1742-6596/962/1/012029).
- [89] S. Ueta, N. Sakaba, and K. Sawa, “Shielding technology for upper structure of HTTR,” *Annals of Nuclear Energy*, vol. 94, pp. 72–79, 2016, ISSN: 0306-4549. DOI: [10.1016/j.anucene.2016.02.021](https://doi.org/10.1016/j.anucene.2016.02.021).
- [90] A. Hamzah, Suwoto, and H. Adrial, “Preliminary analysis of dose rates distribution of experimental power reactor 10 MW using MCNP,” *Journal of Physics: Conference Series*, vol. 1198, no. 2, p. 022038, Apr. 2019. DOI: [10.1088/1742-6596/1198/2/022038](https://doi.org/10.1088/1742-6596/1198/2/022038).
- [91] A. G. Croff, “ORIGEN2: A revised and updated version of the Oak Ridge isotope generation and depletion code,” English, Oak Ridge National Laboratory, United States, Tech. Rep. ORNL-5621, TRN: 80-012377, Jul. 1980. DOI: [10.2172/5352089](https://doi.org/10.2172/5352089).
- [92] ICRP, “Conversion Coefficients for Radiological Protection Quantities for External Radiation Exposures,” ICRP, Technical Report ICRP 40(2-5), 2010.
- [93] DOE-NE and GIF, “A Technology Roadmap for Generation IV Nuclear Energy Systems,” en, Tech. Rep. GIF-002-00, 859029, Dec. 2002, GIF-002-00, 859029. DOI: [10.2172/859029](https://doi.org/10.2172/859029).
- [94] H. Gougar, “NGNP Program 2013 Status and Path Forward,” Idaho National Laboratory, Idaho Falls, ID, USA, Tech. Rep. INL/EXT-14-31035, Mar. 2014. DOI: [10.2172/1129109](https://doi.org/10.2172/1129109).

- [95] *Pinawa Demonstration Reactor*, <https://starcorenuclearpower.com/technology/>, Accessed: 2023-04-01.
- [96] *BWX Technologies selected to build Project Pele microreactor*, <https://www.world-nuclear-news.org/Articles/BWX-Technologies-selected-to-build-Project-Pele-mi>, Accessed: 2023-04-01.
- [97] C. Brooks, T. Kozlowski, C. Townsend, J. Stubbins, and R. Uddin, “New reactor on campus?” *American Nuclear Society Nuclear News*, pp. 24–31, Apr. 2022.
- [98] B. A. Brigham, K. I. Montoya, T. J. Gerczak, and E. S. Sooby, “Determination of oxidation rates and volatile oxidation products for htgr graphite matrix material exposed to steam atmospheres,” *Journal of Nuclear Materials*, vol. 557, p. 153 256, 2021, ISSN: 0022-3115. DOI: <https://doi.org/10.1016/j.jnucmat.2021.153256>.
- [99] R. Fairhurst-Agosta and T. Kozlowski, “Shutdown Dose Rate Calculations of a Gas-Cooled Microreactor,” in *Proceedings of the International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering - MC2023*, Niagara Falls, Canada: ANS, Aug. 2023.
- [100] “USNC Micro Modular Reactor (MMR TM Block 1) Technical Information,” Ultra Safe Nuclear Corporation, Technical Report, 2021. [Online]. Available: <https://www.usnc.com/mmr/>.
- [101] “Project Description for the Micro Modular Reactor TM Project at Chalk River,” Ultra Safe Nuclear Corporation, Technical Report, 2019. [Online]. Available: <https://www.usnc.com/mmr/>.
- [102] A. Hawari, “Development and Deployment Assessment of a Melt-Down Proof Modular Micro Reactor (MDP-MMR),” North Carolina State University, Raleigh, North Carolina, USA, Technical Report 14-6782, 1431209, Apr. 2018.
- [103] K. Yang, E. Kardoulaki, D. Zhao, A. Broussard, K. Metzger, J. T. White, M. R. Sivack, K. J. McClellan, E. J. Lahoda, and J. Lian, “Uranium nitride (un) pellets with controllable microstructure and phase – fabrication by spark plasma sintering and their thermal-mechanical and oxidation properties,” *Journal of Nuclear Materials*, vol. 557, p. 153 272, 2021, ISSN: 0022-3115. DOI: [10.1016/j.jnucmat.2021.153272](https://doi.org/10.1016/j.jnucmat.2021.153272).
- [104] A. T. Incorporated, *Reactor grade zirconium*, Technical Data Sheet, Pittsburgh, PA, 2015. [Online]. Available: https://www.atimaterials.com/Products/Documents/datasheets/zirconium/alloy/Zr_nuke_waste_disposal_v1.pdf.
- [105] M. L. Bleiberg, L. J. Jones, and B. Lustman, “Phase changes in pile-irradiated uranium-base alloys,” *Journal of Applied Physics*, vol. 27, no. 11, pp. 1270–1283, May 2004, ISSN: 0021-8979. DOI: [10.1063/1.1722250](https://doi.org/10.1063/1.1722250).
- [106] S. Wong, *Metal Alloys to support Nuclear Fusion Energy*, Web Article, 2023. [Online]. Available: <https://www.pnnl.gov/news-media/metal-alloys-support-nuclear-fusion-energy>.
- [107] K. G. Field, Y. Yamamoto, B. Pint, M. N. Gussev, and K. Terrani, “Accident tolerant FeCrAl fuel cladding: Current status towards commercialization,” English, Research Org.: Oak Ridge National Lab. (ORNL), Oak Ridge, TN (United States), Nov. 2017. DOI: [10.1007/978-3-030-04639-2_91](https://doi.org/10.1007/978-3-030-04639-2_91).
- [108] D. Gaston, C. Newman, G. Hansen, and D. Lebrun-Grandie, “MOOSE: A parallel computational framework for coupled systems of nonlinear equations,” *Nuclear Engineering and Design*, vol. 239, no. 10, pp. 1768–1778, Oct. 2009, ISSN: 0029-5493. DOI: [10.1016/j.nucengdes.2009.05.021](https://doi.org/10.1016/j.nucengdes.2009.05.021).

- [109] H. G. Weller, G. Tabor, H. Jasak, and C. Fureby, “A tensorial approach to computational continuum mechanics using object-oriented techniques,” en, *Computers in Physics*, vol. 12, no. 6, p. 620, 1998, ISSN: 08941866. DOI: [10.1063/1.168744](https://doi.org/10.1063/1.168744).
- [110] S. Longshaw, M. Borg, S. Ramisetti, J. Zhang, D. Lockerby, D. Emerson, and J. Reese, “mdFoam+: Advanced molecular dynamics in OpenFOAM,” en, *Computer Physics Communications*, vol. 224, pp. 1–21, Mar. 2018, ISSN: 00104655. DOI: [10.1016/j.cpc.2017.09.029](https://doi.org/10.1016/j.cpc.2017.09.029).
- [111] *Openfoam Repository*, <https://develop.openfoam.com/Development/openfoam>.
- [112] F. Chollet *et al.*, *Keras*, Publisher: GitHub, 2015. [Online]. Available: <https://github.com/fchollet/keras>.
- [113] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine Learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [114] L. Breiman, “Bagging predictors,” *Machine Learning*, vol. 24, no. 2, pp. 123–140, Aug. 1996, ISSN: 1573-0565. DOI: [10.1007/BF00058655](https://doi.org/10.1007/BF00058655).
- [115] M. Ravikumar, *Let’s Talk Bayesian Optimization*, Accessed: 2022-07, Nov. 2018. [Online]. Available: <https://mlconf.com/blog/lets-talk-bayesian-optimization/>.
- [116] T. Head, M. Kumar, H. Nahrstaedt, G. Louppe, and I. Shcherbatyi, *Scikit-optimize/scikit-optimize*, version v0.9.0, Oct. 2021. DOI: [10.5281/zenodo.5565057](https://doi.org/10.5281/zenodo.5565057).
- [117] F. Incropera, *Fundamentals of heat and mass transfer*, 6th ed. Hoboken N.J.: Wiley, 2006, ISBN: 978-0-471-45728-2.
- [118] P. Virtanen *et al.*, “SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python,” *Nature Methods*, vol. 17, pp. 261–272, 2020. DOI: [10.1038/s41592-019-0686-2](https://doi.org/10.1038/s41592-019-0686-2). [Online]. Available: <https://rdcu.be/b08Wh>.
- [119] W. Wiscombe and J. Evans, “Exponential-sum fitting of radiative transmission functions,” *Journal of Computational Physics*, vol. 24, no. 4, pp. 416–444, 1977, ISSN: 0021-9991. DOI: [10.1016/0021-9991\(77\)90031-6](https://doi.org/10.1016/0021-9991(77)90031-6).
- [120] D. G. Cantor and J. W. Evans, “On Approximation by Positive Sums of Powers,” *SIAM Journal on Applied Mathematics*, vol. 18, no. 2, pp. 380–388, 1970, ISSN: 00361399. [Online]. Available: <http://www.jstor.org/stable/2099472> (visited on 10/02/2023).
- [121] B. Coleman, *Fitting Exponential Decay Sums with Positive Coefficients*, Accessed: 2023-06, Mar. 2020. [Online]. Available: <https://randorithms.com/2020/03/08/exponential-sum-fits.html>.
- [122] C. Geuzaine and J.-F. Remacle, “Gmsh: A three-dimensional finite element mesh generator with built-in pre- and post-processing facilities.,” *International Journal for Numerical Methods in Engineering*, 2009. [Online]. Available: <https://gmsh.info/>.
- [123] D. J. Antonio, K. Shrestha, J. M. Harp, C. A. Adkins, Y. Zhang, J. Carmack, and K. Gofryk, “Thermal and transport properties of U₃Si₂,” *Journal of Nuclear Materials*, vol. 508, May 2018, ISSN: 0022-3115. DOI: [10.1016/j.jnucmat.2018.05.036](https://doi.org/10.1016/j.jnucmat.2018.05.036).
- [124] C. R. Harris *et al.*, “Array programming with NumPy,” *Nature*, vol. 585, no. 7825, pp. 357–362, Sep. 2020. DOI: [10.1038/s41586-020-2649-2](https://doi.org/10.1038/s41586-020-2649-2).

- [125] T. pandas development team, *Pandas-dev/pandas: Pandas*, version latest, Feb. 2020. doi: [10.5281/zenodo.3509134](https://doi.org/10.5281/zenodo.3509134).
- [126] J. D. Hunter, “Matplotlib: A 2D graphics environment,” *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007. doi: [10.1109/MCSE.2007.55](https://doi.org/10.1109/MCSE.2007.55).

Appendix A

MOAA execution

A.1 User Input File

The creation of a MOAA application starts with the definition of a UIF in *.xml* format. The UIF contains a set of required and optional settings for MOAA. The required settings are a list of irradiation cases, a list of decay times, and a list of cells. Within the irradiation cases, the user must specify an associated MCNP input file, an irradiation time, and an irradiation power.

Listing A.1 displays a minimal MOAA input file which includes a single irradiation case, a single decay time step, and a single cell. The irradiation case geometry and material definition are specified by the MCNP input file `test1.i` with an irradiation duration of 1 day and a reactor power of 100 MW. The irradiation period is followed by a decay of 1 day. The user-defined cell for the calculations is cell number 1 of the MCNP input file.

```
<moaa>
    <irradiation_cases>
        <case>
            <mcnp_input_file>test1.i</mcnp_input_file>
            <time>1.0</time>
            <power>100</power>
        </case>
    </irradiation_cases>
    <decay_times>
        <time>1</time>
    </decay_times>
    <cells>
        <cell>1</cell>
    </cells>
</moaa>
```

Listing A.1: Example of a minimalist MOAA UIF.

A.2 ATR case

The required settings within the irradiation cases include an MCNP input file, an irradiation time, an irradiation power divided into five lobe powers: NW, NE, C, SW, SE. Additionally, the user must provide the F7 tally identification number that will be used for adjusting the lobe powers.

Listing A.2 displays a typical UIF for an ATR irradiation experiment. This application includes a single irradiation case, two decay times, and 1800 cells (only four are shown). The irradiation case geometry and material definition are specified by the MCNP input file `ne71a.i1`, with an irradiation duration of 3 days, and a total reactor power of 120 MW, which is subdivided into five reactor lobes. The irradiation period is followed by a decay of 12 hours. The user-defined cells for the calculations are cells number 501101, 501102, 508459, and 508460 of the MCNP input file. Finally, the adjusted lobe power calculation relies on tally number 7.

```

<moaa>
    <atr>true</atr>
    <irradiation_cases>
        <case>
            <mcnp_input_file>ne71a.i1</mcnp_input_file>
            <time units="d">3</time>
            <atr_lobe_powers>
                <nw>23.0</nw>
                <ne>20.0</ne>
                <c>25.0</c>
                <sw>26.0</sw>
                <se>26.0</se>
            </atr_lobe_powers>
        </case>
    </irradiation_cases>
    <decay_times>
        <time units="s">1</time>
        <time units="h">12</time>
    </decay_times>
    <cells>
        <cell lobe="ne">501101</cell>
        <cell lobe="ne">501102</cell>
        ...
        <cell lobe="ne">508459</cell>
        <cell lobe="ne">508460</cell>
    </cells>
    <atr_lp_tally_id>7</atr_lp_tally_id>
</moaa>
```

Listing A.2: Example of a typical MOAA UIF for an ATR irradiation experiment.

A.3 Executing MOAA

The execution of MOAA requires the installation of several common dependencies, such as *NumPy* [124] and *Pandas* [125]. The recommended installation procedure is to create a Python virtual environment in `$MOAA-ENV` for their installation. The next step is the installation of the MOAA package.

After the installation, the user can choose to execute MOAA either in interactive or batch mode from the command line. The only required positional argument is the UIF: `moaa input.xml`. The environment path must include MCNP and SCALE executables for the successful run of the UIF. Additionally, the UIF directory must contain a subdirectory holding the MCNP input files defining each irradiation case. Running MOAA in batch mode uses a submission script for execution on a HPC cluster, as shown in Listing A.3.

```
#!/bin/bash
#PBS -l select=1:ncpus=48:mpiprocs=48
#PBS -N MOAA_RUN
#PBS -l walltime=12:00:00
#PBS -P projectname
#PBS -v DATAPATH

export DATAPATH=/path/to/mcnp_xs_libraries/
cat $PBS_NODEFILE
# Import any modules
module load mcnp
module load scale
# Activate MOAA virtual environment
source $MOAA-ENV/bin/activate
#
cd $PBS_O_WORKDIR
moaa moaa.xml
deactivate
```

Listing A.3: HPC job submission script.

The execution of MOAA may include several optional arguments that give the user greater flexibility. These arguments include the definition of a non-default output directory, a non-default log-file name, and the choice of different verbosity levels for the logged information.

During the execution of MOAA, the generation and execution of the SCALE input files, as well as the parsing of the SCALE output file, occurs concurrently. MOAA relies on the Python standard library *Asyncio*, which is the foundation for Python asynchronous frameworks, allowing for the creation and execution of co-routines. This feature allows for an overall reduction in the run times.

A.4 Simulation output

Upon successful execution of a MOAA simulation, a number of files are created in the UIF directory based on the user-defined settings. The most common files are an execution log-file `MOAA.log` and a `.xlsx` spreadsheet with the isotopic composition.

MOAA leverages the Python built-in logging module for informing the user on the progression of the simulation. The log-file contains all the logging messages produced during the execution, which may include warnings and errors.

The generated spreadsheets contain the isotopic composition at the end of the last irradiation step and after each decay step. The different tabs of the spreadsheet separate the results by irradiation cell. The MOAA calculation may produce additional *.xlsx* files containing the simulation results in the user-defined units of interest, such as Curies and Watts.

The choice of *.xlsx* format for the output files comes from the fact that most experiment analysts and clients are familiar with spreadsheet editors. The results in these files can be easily visualized with the editor's built-in plotting capabilities or with python using *Pandas* and *Matplotlib* [126]. An example of such visualization using the isotopic composition of the input file example from Section A.1 is presented in Figure A.1.

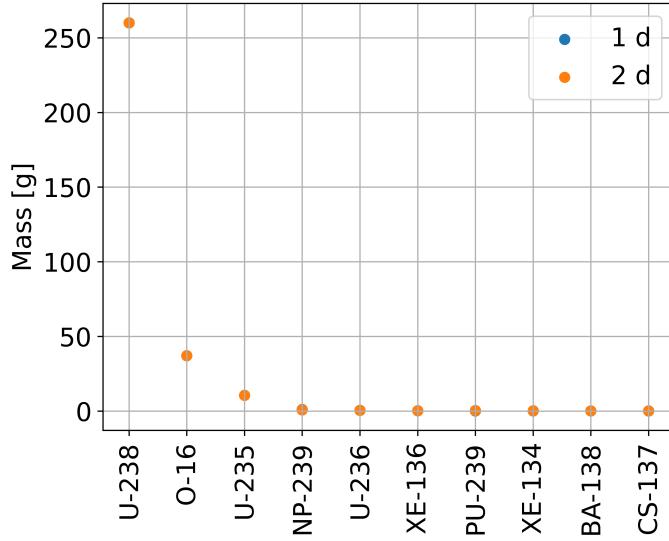


Figure A.1: Example of an output processing with *matplotlib*. Only the ten largest contributors to the total mass are displayed.

Furthermore, MOAA has the ability to combine the results for a group of user-defined cells. When several cells form a region of interest, MOAA combines the source term from all of them. This feature is helpful in cases wherein an experiment capsule is defined by multiple cells and materials, for example. This step allows for computing a source term for the whole capsule and thus easing the results post-processing.

Appendix B

Execution of Delayed Heating Calculation Workflow

The calculation workflow requires the MOAA UIF and the `mcnp` directory containing the MCNP input files for the burnup calculation, as shown in Figure B.1.

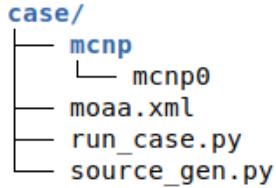


Figure B.1: Delayed heating case directory.

The delayed heating calculation requires two additional scripts: `run_case.py` and `source_gen.py`. The user defines some of the calculation parameters in `run_case.py`. Some of these parameters include the name of the MOAA UIF, path to the MCNP neutron- and photon-transport input files, geometry cell numbers where to calculate the delayed heating, and photon-transport batch size - i.e., number of photon sources to be included in each photon-transport input file. Section B.1 provides further details on the batch size. If neutron and photon geometries are identical, the user must provide number of particles, and number of active and inactive cycles for the neutron-transport calculation.

The execution of the workflow can be either interactive or batch mode from the command line by simply running `run_case.py`. An example of a submission script for execution on an HPC cluster can be found in Appendix A, where instead of running MOAA, the user would run `run_case.py`. Upon successful execution, a `.csv` is created summarizing the results. The information is organized into rows and columns, where each row represents a decay step after shutdown, while each column contains the energy deposited by charged particles (H_{ch}), the energy deposited by photons ($H_{\gamma,Tr}$), and the total delayed heat (H_T), to mention a few.

B.1 Batch Size

This work implements three different ways of calculating the delayed photon heating ($H_{\gamma,Tr}$):

First Method: Runs one photon transport simulation for each cell contributing to the gamma heating. For a system composed by N cells, the method obtains the contribution of each cell independently ($*F8_i$), and running N simulations calculates the total gamma heating:

$$H_{\gamma,Tr}[W] = 1.6022 \times 10^{-13} \times \sum_{i \in N} *F8_i \times S_i \quad (\text{B.1})$$

Second Method: Obtains all contributions to the gamma heating of all cells simultaneously ($*F8$), and then calculates the total gamma heating:

$$H_{\gamma,Tr}[W] = 1.6022 \times 10^{-13} \times *F8 \times \sum_{i \in N} S_i \quad (\text{B.2})$$

Third Method: Splits the calculation into M simulations ($M < N$). Each simulation includes N_i cells ($N_i < N$). The method calculates the contribution to the gamma heating of N_i cells simultaneously ($*F8_i$), and then calculates the total gamma heating:

$$H_{\gamma,Tr}[W] = 1.6022 \times 10^{-13} \times \sum_{i \in M} \left[*F8_i \times \sum_{j \in N_i} S_j \right] \quad (\text{B.3})$$

The first method runs N photon-transport simulations, and in terms of computational expense, it is the most expensive. However, this method allows to compare the magnitude of the different gamma heating contributors. Additionally, each individual fixed-source simulation is completed in a matter of seconds. Only when a system comprises a large number of cells, then, the computational expense becomes restrictive.

The second method runs only one photon-transport simulation, making it the least computationally intensive. However, this method is only applicable for systems with a small number of cells due to MCNP limitations. To sample the position of the particles in a cell, MCNP uses the enclosing volume rejection method. MCNP randomly samples the position within the enclosing volume, if the points fall within the cell, they are accepted.

To define the enclosing volume, MCNP uses the SI_i card, for which i is restricted to values between 1 and 999. Our method defines additional SI cards to sample the cell where the particle is born and the energy of the particle for each cell. If the enclosing volume is a box, it requires three additional SI cards for the position and one for the energy for each cell. Overall, MCNP can sample the photon particles in only 248 cells within one input file, if the enclosing volumes are boxes or cylinders. This reason motivated the development of the third method, which still allows to compute the total gamma heating with a considerably low computational expense.

Figure B.2 displays the experiment heat production over time. $H_{\gamma,Tr}$ was calculated for 761 and 4 independent photon-transport simulations. Both results are identical within error bars.

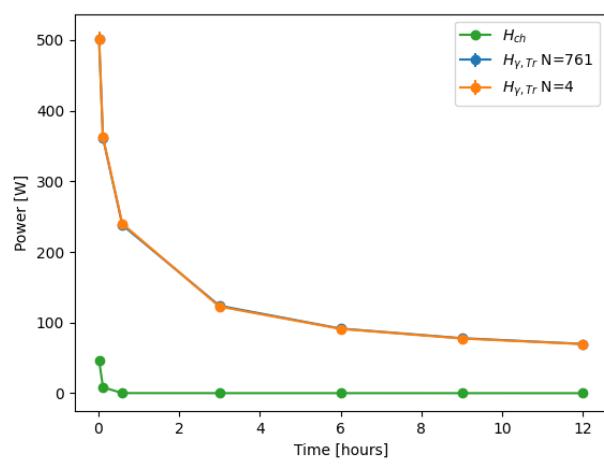


Figure B.2: Heating rate over time in the ATR A1 experiment position for an aluminum sample.

Appendix C

Serpent Gamma Source

This section presents an example of a `_gsrc.m` file produced by Serpent. The output file is divided into materials and their respective nuclides. For this work, the following columns contain the most valuable information. First column is the ZAI of the nuclide, third column is the total gamma emission rate of the nuclide, fifth column is the gamma emission energy, and seventh column is the emission rate cumulative fraction for the gamma emission energy specified in the fifth column. Figure C.1 displays the gamma source distribution for each nuclide specified in this example.

```
mat_m1 = [  
  
% —— Te-132 :  
  
521320 1.84967E+00 1.27871E+16 4.11726E-01 2.28160E-01 8.80000E-01 4.75762E-01  
521320 1.84967E+00 1.27871E+16 4.11726E-01 2.86120E-02 3.77008E-01 6.79586E-01  
521320 1.84967E+00 1.27871E+16 4.11726E-01 2.83170E-02 2.04754E-01 7.90284E-01  
521320 1.84967E+00 1.27871E+16 4.11726E-01 4.97200E-02 1.49600E-01 8.71164E-01  
521320 1.84967E+00 1.27871E+16 4.11726E-01 3.94000E-03 7.83222E-02 9.13508E-01  
521320 1.84967E+00 1.27871E+16 4.11726E-01 3.22950E-02 6.77639E-02 9.50144E-01  
521320 1.84967E+00 1.27871E+16 4.11726E-01 3.22390E-02 3.51007E-02 9.69120E-01  
521320 1.84967E+00 1.27871E+16 4.11726E-01 3.30470E-02 2.00692E-02 9.79970E-01  
521320 1.84967E+00 1.27871E+16 4.11726E-01 1.16300E-01 1.96240E-02 9.90580E-01  
521320 1.84967E+00 1.27871E+16 4.11726E-01 1.11760E-01 1.74240E-02 1.00000E+00  
  
% —— Sr-92 :  
  
380920 1.03234E+00 9.11055E+15 6.39262E-01 1.38393E+00 9.00000E-01 8.71809E-01  
380920 1.03234E+00 9.11055E+15 6.39262E-01 9.53310E-01 3.51900E-02 9.05897E-01  
380920 1.03234E+00 9.11055E+15 6.39262E-01 4.30490E-01 3.27600E-02 9.37631E-01  
380920 1.03234E+00 9.11055E+15 6.39262E-01 2.41560E-01 2.92500E-02 9.65965E-01  
380920 1.03234E+00 9.11055E+15 6.39262E-01 1.14235E+00 2.79000E-02 9.92991E-01  
380920 1.03234E+00 9.11055E+15 6.39262E-01 6.50800E-01 3.69000E-03 9.96565E-01  
380920 1.03234E+00 9.11055E+15 6.39262E-01 4.91270E-01 2.74500E-03 9.99224E-01
```

```
380920 1.03234E+00 9.11055E+15 6.39262E-01 8.92680E-01 8.01000E-04 1.00000E+00
```

```
% —— Zr-95 :
```

```
400950 9.91185E-01 4.03323E+15 8.43173E-01 7.56725E-01 5.43800E-01 5.48636E-01  
400950 9.91185E-01 4.03323E+15 8.43173E-01 7.24192E-01 4.42700E-01 9.95273E-01  
400950 9.91185E-01 4.03323E+15 8.43173E-01 1.65210E-02 4.37482E-03 9.99687E-01  
400950 9.91185E-01 4.03323E+15 8.43173E-01 2.17000E-03 3.10473E-04 1.00000E+00
```

```
% —— Pd-112 :
```

```
461120 3.07251E-01 4.66930E+13 9.98279E-01 1.85000E-02 2.72000E-01 8.85268E-01  
461120 3.07251E-01 4.66930E+13 9.98279E-01 2.98000E-03 3.52514E-02 1.00000E+00
```

```
% —— Ra-225 :
```

```
882250 4.36388E-01 4.94167E-01 1.00000E+00 4.00000E-02 3.00000E-01 6.87461E-01  
882250 4.36388E-01 4.94167E-01 1.00000E+00 1.27000E-02 1.36388E-01 1.00000E+00  
];
```

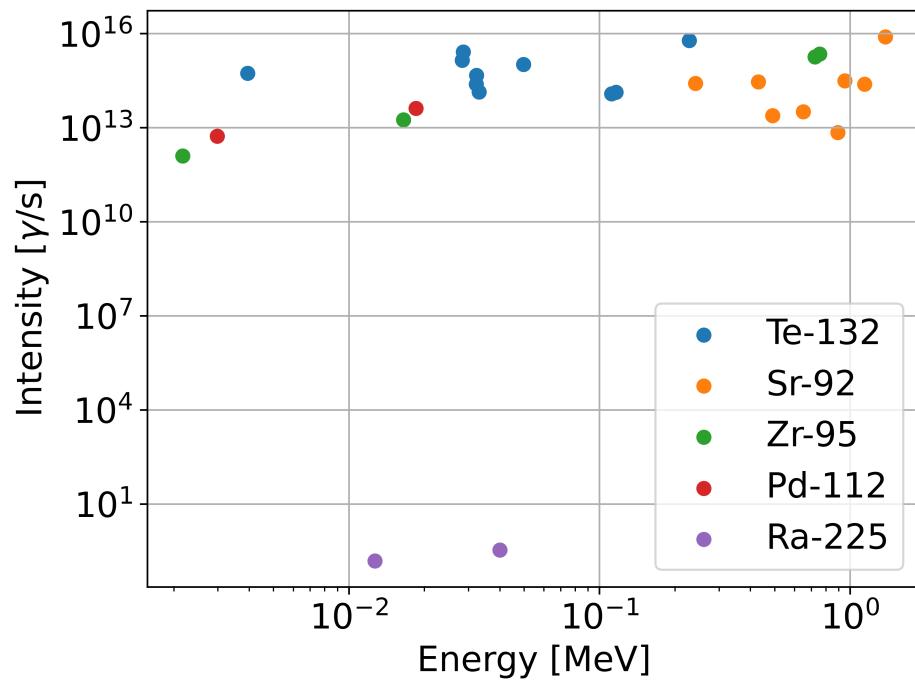


Figure C.1: Example of gamma source distribution produced by Serpent.

Appendix D

Experiment Material Properties for Neutronics Calculation

Table D.1: Materials included in the ATR experiment model.

Z	Symbol	M [g/mol]	ρ [g/cm ³]	A	a_i
1	h	1.0078	8.37480e-5	1	0.999885
1	h	2.0141	8.37480e-5	2	0.000115
2	he	3.016	1.66322e-4	3	0.00000134
2	he	4.0026	1.66322e-4	4	0.99999866
3	li	6.0151	5.34e-01	6	0.0759
3	li	7.016	5.34e-01	7	0.9241
4	be	9.0121831	1.848	9	1.0
5	b	10.0129	2.37	10	0.199
5	b	11.0093	2.37	11	0.801
6	c	12.0	2.26	12	0.99
6	c	13.0034	2.26	13	0.01
7	n	14.0031	1.16528e-3	14	0.99636
7	n	15.0001	1.16528e-3	15	0.00364
8	o	15.9949	1.33151e-3	16	0.99757
8	o	16.9991	1.33151e-3	17	0.00038
8	o	17.9992	1.33151e-3	18	0.00205
9	f	18.998403163	1.58029E-03	19	1.0
11	na	22.98976928	9.71e-1	23	1.0
12	mg	23.985	1.74	24	0.7899
12	mg	24.9858	1.74	25	0.1000
12	mg	25.9826	1.74	26	0.1101
13	al	26.9815385	2.6989	27	1.0
14	si	27.9769	2.33	28	0.92223
14	si	28.9765	2.33	29	0.04685

14	si	29.9738	2.33	30	0.03092
15	p	30.973761998	2.2	31	1.0
16	s	31.9721	2.0	32	0.9499
16	s	32.9715	2.0	33	0.0075
16	s	33.9679	2.0	34	0.0425
16	s	35.9671	2.0	36	0.0001
17	cl	34.9688	0.00299473	35	0.7576
17	cl	36.9659	0.00299473	37	0.2424
18	ar	35.9675	0.00166201	36	0.003336
18	ar	37.9627	0.00166201	38	0.000629
18	ar	39.9624	0.00166201	40	0.996035
19	k	38.9637	0.862	39	0.932581
19	k	39.964	0.862	40	0.000117
19	k	40.9618	0.862	41	0.067302
20	ca	39.9626	1.55	40	0.96941
20	ca	41.9586	1.55	42	0.00647
20	ca	42.9588	1.55	43	0.00135
20	ca	43.9555	1.55	44	0.02086
20	ca	45.9537	1.55	46	0.00004
20	ca	47.9525	1.55	48	0.00187
21	sc	44.955908	2.989	45	1.0
22	ti	45.9526	4.54	46	0.0825
22	ti	46.9518	4.54	47	0.0744
22	ti	47.9479	4.54	48	0.7372
22	ti	48.9479	4.54	49	0.0541
22	ti	49.9448	4.54	50	0.0518
23	v	49.9472	6.11	50	0.00250
23	v	50.944	6.11	51	0.99750
24	cr	49.946	7.18	50	0.04345
24	cr	51.9405	7.18	52	0.83789
24	cr	52.9407	7.18	53	0.09501
24	cr	53.9389	7.18	54	0.02365
25	mn	54.938044	7.26	55	1.0
26	fe	53.9396	7.874	54	0.05845
26	fe	55.9349	7.874	56	0.91754
26	fe	56.9354	7.874	57	0.02119
26	fe	57.9333	7.874	58	0.00282
27	co	58.933195	8.9	59	1.0
28	ni	57.9353	8.902	58	0.68077
28	ni	59.9308	8.902	60	0.26223
28	ni	60.9311	8.902	61	0.011399

28	ni	61.9283	8.902	62	0.036346
28	ni	63.928	8.902	64	0.009255
29	cu	62.9296	8.96	63	0.6915
29	cu	64.9278	8.96	65	0.3085
30	zn	63.9291	7.133	64	0.4917
30	zn	65.926	7.133	66	0.2773
30	zn	66.9271	7.133	67	0.0404
30	zn	67.9248	7.133	68	0.1845
30	zn	69.9253	7.133	70	0.0061
31	ga	68.9256	5.904	69	0.60108
31	ga	70.9247	5.904	71	0.39892
32	ge	69.9242	5.323	70	0.2057
32	ge	71.9221	5.323	72	0.2745
32	ge	72.9235	5.323	73	0.0775
32	ge	73.9212	5.323	74	0.3650
32	ge	75.9214	5.323	76	0.0773
33	as	74.921595	5.73	75	1.0
34	se	73.9225	4.5	74	0.0089
34	se	75.9192	4.5	76	0.0937
34	se	76.9199	4.5	77	0.0763
34	se	77.9173	4.5	78	0.2377
34	se	79.9165	4.5	80	0.4961
34	se	81.9167	4.5	82	0.0873
35	br	78.9183	0.00707218	79	0.5069
35	br	80.9163	0.00707218	81	0.4931
36	kr	77.9204	0.00347832	78	0.00355
36	kr	79.9164	0.00347832	80	0.02286
36	kr	81.9135	0.00347832	82	0.11593
36	kr	82.9141	0.00347832	83	0.11500
36	kr	83.9115	0.00347832	84	0.56987
36	kr	85.9106	0.00347832	86	0.17279
37	rb	84.9118	1.532	85	0.7217
37	rb	86.9092	1.532	87	0.2783
38	sr	83.9134	2.54	84	0.0056
38	sr	85.9093	2.54	86	0.0986
38	sr	86.9089	2.54	87	0.0700
38	sr	87.9056	2.54	88	0.8258
39	y	88.90584	4.469	89	1.0
40	zr	89.9047	6.506	90	0.5145
40	zr	90.9056	6.506	91	0.1122
40	zr	91.905	6.506	92	0.1715

40	zr	93.9063	6.506	94	0.1738
40	zr	95.9083	6.506	96	0.0280
41	nb	92.90637	8.57	93	1.0
42	mo	91.9068	10.22	92	0.1453
42	mo	93.9051	10.22	94	0.0915
42	mo	94.9058	10.22	95	0.1584
42	mo	95.9047	10.22	96	0.1667
42	mo	96.906	10.22	97	0.0960
42	mo	97.9054	10.22	98	0.2439
42	mo	99.9075	10.22	100	0.0982
44	ru	95.9076	12.41	96	0.0554
44	ru	97.9053	12.41	98	0.0187
44	ru	98.9059	12.41	99	0.1276
44	ru	99.9042	12.41	100	0.1260
44	ru	100.9056	12.41	101	0.1706
44	ru	101.9044	12.41	102	0.3155
44	ru	103.9054	12.41	104	0.1862
45	rh	102.90550	12.41	103	1.0
46	pd	101.9056	12.02	102	0.0102
46	pd	103.904	12.02	104	0.1114
46	pd	104.9051	12.02	105	0.2233
46	pd	105.9035	12.02	106	0.2733
46	pd	107.9039	12.02	108	0.2646
46	pd	109.9052	12.02	110	0.1172
47	ag	106.9051	10.5	107	0.51839
47	ag	108.9047	10.5	109	0.48161
48	cd	105.9065	8.65	106	0.0125
48	cd	107.9042	8.65	108	0.0089
48	cd	109.903	8.65	110	0.1249
48	cd	110.9042	8.65	111	0.1280
48	cd	111.9028	8.65	112	0.2413
48	cd	112.9044	8.65	113	0.1222
48	cd	113.9034	8.65	114	0.2873
48	cd	115.9048	8.65	116	0.0749
49	in	112.9041	7.31	113	0.0429
49	in	114.9039	7.31	115	0.9571
50	sn	111.9048	5.75	112	0.0097
50	sn	113.9028	5.75	114	0.0066
50	sn	114.9033	5.75	115	0.0034
50	sn	115.9017	5.75	116	0.1454
50	sn	116.9029	5.75	117	0.0768

50	sn	117.9016	5.75	118	0.2422
50	sn	118.9033	5.75	119	0.0859
50	sn	119.9022	5.75	120	0.3258
50	sn	121.9034	5.75	122	0.0463
50	sn	123.9053	5.75	124	0.0579
51	sb	120.9038	6.691	121	0.5721
51	sb	122.9042	6.691	123	0.4279
52	te	119.904	6.24	120	0.0009
52	te	121.903	6.24	122	0.0255
52	te	122.9043	6.24	123	0.0089
52	te	123.9028	6.24	124	0.0474
52	te	124.9044	6.24	125	0.0707
52	te	125.9033	6.24	126	0.1884
52	te	127.9045	6.24	128	0.3174
52	te	129.9062	6.24	130	0.3408
53	i	126.90447	4.93	127	1.0
54	xe	123.9059	0.00548536	124	0.000952
54	xe	125.9043	0.00548536	126	0.000890
54	xe	127.9035	0.00548536	128	0.019102
54	xe	128.9048	0.00548536	129	0.264006
54	xe	129.9035	0.00548536	130	0.040710
54	xe	130.9051	0.00548536	131	0.212324
54	xe	131.9041	0.00548536	132	0.269086
54	xe	133.9054	0.00548536	134	0.104357
54	xe	135.9072	0.00548536	136	0.088573
55	cs	132.90545196	1.873	133	1.0
56	ba	129.9063	3.5	130	0.00106
56	ba	131.9051	3.5	132	0.00101
56	ba	133.9045	3.5	134	0.02417
56	ba	134.9057	3.5	135	0.06592
56	ba	135.9046	3.5	136	0.07854
56	ba	136.9058	3.5	137	0.11232
56	ba	137.9052	3.5	138	0.71698
57	la	137.9071	6.154	138	0.0008881
57	la	138.9064	6.154	139	0.9991119
58	ce	135.9072	6.657	136	0.00185
58	ce	137.906	6.657	138	0.00251
58	ce	139.9054	6.657	140	0.88450
58	ce	141.9092	6.657	142	0.11114
59	pr	140.90766	6.71	141	1.0
60	nd	141.9077	6.9	142	0.27152

60	nd	142.9098	6.9	143	0.12174
60	nd	143.9101	6.9	144	0.23798
60	nd	144.9126	6.9	145	0.08293
60	nd	145.9131	6.9	146	0.17189
60	nd	147.9169	6.9	148	0.05756
60	nd	149.9209	6.9	150	0.05638
62	sm	143.912	7.46	144	0.0307
62	sm	146.9149	7.46	147	0.1499
62	sm	147.9148	7.46	148	0.1124
62	sm	148.9172	7.46	149	0.1382
62	sm	149.9173	7.46	150	0.0738
62	sm	151.9197	7.46	152	0.2675
62	sm	153.9222	7.46	154	0.2275
63	eu	150.9198	5.243	151	0.4781
63	eu	152.9212	5.243	153	0.5219
64	gd	151.9198	7.9004	152	0.0020
64	gd	153.9209	7.9004	154	0.0218
64	gd	154.9226	7.9004	155	0.1480
64	gd	155.9221	7.9004	156	0.2047
64	gd	156.924	7.9004	157	0.1565
64	gd	157.9241	7.9004	158	0.2484
64	gd	159.927	7.9004	160	0.2186
65	tb	158.92535	8.229	159	1.0
66	dy	155.9243	8.55	156	0.00056
66	dy	157.9244	8.55	158	0.00095
66	dy	159.9252	8.55	160	0.02329
66	dy	160.9269	8.55	161	0.18889
66	dy	161.9268	8.55	162	0.25475
66	dy	162.9287	8.55	163	0.24896
66	dy	163.9292	8.55	164	0.28260
67	ho	164.93033	8.795	165	1.0
68	er	161.9288	9.066	162	0.00139
68	er	163.9292	9.066	164	0.01601
68	er	165.9303	9.066	166	0.33503
68	er	166.9321	9.066	167	0.22869
68	er	167.9324	9.066	168	0.26978
68	er	169.9355	9.066	170	0.14910
69	tm	168.93422	9.321	169	1.0
71	lu	174.9408	9.84	175	0.97401
71	lu	175.9427	9.84	176	0.02599
72	hf	173.94	13.31	174	0.0016

72	hf	175.9414	13.31	176	0.0526
72	hf	176.9432	13.31	177	0.1860
72	hf	177.9437	13.31	178	0.2728
72	hf	178.9458	13.31	179	0.1362
72	hf	179.9465	13.31	180	0.3508
73	ta	179.9475	16.654	180	0.0001201
73	ta	180.948	16.654	181	0.9998799
74	w	179.9467	19.3	180	0.0012
74	w	181.9482	19.3	182	0.2650
74	w	182.9502	19.3	183	0.1431
74	w	183.9509	19.3	184	0.3064
74	w	185.9544	19.3	186	0.2843
75	re	184.953	21.02	185	0.3740
75	re	186.9557	21.02	187	0.6260
77	ir	190.9606	22.42	191	0.373
77	ir	192.9629	22.42	193	0.627
79	au	196.966569	19.32	197	1.0
80	hg	195.9658	13.546	196	0.0015
80	hg	197.9668	13.546	198	0.0997
80	hg	198.9683	13.546	199	0.1687
80	hg	199.9683	13.546	200	0.2310
80	hg	200.9703	13.546	201	0.1318
80	hg	201.9706	13.546	202	0.2986
80	hg	203.9735	13.546	204	0.0687
81	tl	202.9723	11.72	203	0.2952
81	tl	204.9744	11.72	205	0.7048
82	pb	203.973	11.35	204	0.014
82	pb	205.9745	11.35	206	0.241
82	pb	206.9759	11.35	207	0.221
82	pb	207.9767	11.35	208	0.524
83	bi	208.98040	9.747	209	1.0
90	th	232.0381	11.72	232	1.0
91	pa	231.03588	15.37	231	1.0
92	u	235.0439	18.95	235	0.0072
92	u	238.0508	18.95	238	0.9928
94	pu	239.0522	19.84	239	0.90490
94	pu	240.0538	19.84	240	0.08580
94	pu	241.0569	19.84	241	0.00890
94	pu	242.0587	19.84	242	0.00040

Appendix E

Experiment Material Properties for Thermal-Fluids Model

Table E.1: Materials included in the ATR experiment model.

Material	Z	Symbol	[kg/m ³]	k [W/m/K]	c _p [J/kg/K]
aluminum	13	al	2700.0	167	896
americium	95	am	11700.0	12.552	126
antimony	51	sb	6696.0	18.41	207
arsenic_grey	33	as	5730.0	4.184	331
barium	56	ba	3500.0	33.472	188
berkelium	97	bk	12000.0	12.552	126
beryllium	4	be	1848.0	184.1	1841
bismuth	83	bi	9750.0	8.368	121
boron	5	b	2500.0	2.092	1130
bromine	35	br	3100.0	0.122	460
cadmium	48	cd	8650.0	92.05	230
calcium	20	ca	1550.0	125.52	657
californium	98	cf	12000.0	12.552	126
carbon_graphite	6	c	2250.0	167.36	707
cerium	58	ce	6790.0	10.878	193
cesium	55	cs	1870.0	41.84	230
cobalt	27	co	8860.0	84.935	423
copper	29	cu	8940.0	397.48	385
curium	96	cm	7000.0	12.552	126
einsteinium	99	es	12000.0	12.552	126
erbium	68	er	9060.0	9.623	168
europium	63	eu	5300.0	12.552	176
fermium	100	fm	12000.0	12.552	126
francium	87	fr	12000.0	12.552	142

gadolinium	64	gd	7870.0	8.786	230
gallium	31	ga	5907.0	33.472	372
germanium	32	ge	5325.0	58.576	322
gold	79	au	19300.0	317.98	128
hafnium	72	hf	13200.0	22.175	142
holmium	67	ho	8800.0	10.042	165
indium	49	in	7300.0	23.43	239
iodine_solid	53	i	4930.0	0.004	218
iridium	77	ir	22500.0	146.44	130
iron_0to3000degC	26	fe	7870.0	71.965	448
lanthanum	57	la	6200.0	13.807	193
lawrencium	103	lr	12000.0	12.552	126
lead	82	pb	11350.0	34.309	130
lithium	3	li	530.0	71.128	3515
magnesium	12	mg	1740.0	150.62	1004
manganese	25	mn	7400.0	6.694	481
mendelevium	101	md	12000.0	12.552	126
mercury_liquid	80	hg	13500.0	8.368	138
molybdenum	42	mo	10200.0	133.89	247
neptunium	93	np	18000.0	12.552	126
nickel	28	ni	8900.0	87.864	444
niobium	41	nb	8570.0	50.208	268
nobelium	102	no	12000.0	12.552	126
palladium	46	pd	11400.0	76.149	243
plutonium	94	pu	19200.0	8.201	134
polonium	84	po	9320.0	1.674	121
potassium	19	k	862.0	97.069	753
praseodymium	59	pr	6800.0	11.715	197
promethium	61	pm	7200.0	12.552	188
radium	88	ra	5000.0	12.552	113
rhenium	75	re	21000.0	60.668	138
rhodium	45	rh	12400.0	150.62	243
rubidium	37	rb	1500.0	50.208	377
ruthenium	44	ru	12200.0	104.6	234
samarium	62	sm	7500.0	12.552	180
scandium	21	sc	3000.0	12.552	552
selenium_grey	34	se	4790.0	0.753	322
silicon	14	si	2330.0	125.52	703
silver	47	ag	10500.0	426.77	236
sodium	11	na	970.0	133.89	1213
strontium	38	sr	2600.0	83.68	301

sulfur	16	s	2000.0	0.276	711
tantalum	73	ta	16600.0	71.128	138
technetium	43	tc	11500.0	12.552	243
tellurium	52	te	6240.0	3.347	201
terbium	65	tb	8250.0	8.368	182
thallium	81	tl	11850.0	38.911	130
thorium	90	th	11680.0	38.493	117
thulium	69	th	9300.0	8.368	160
tin	50	sn	7310.0	61.086	226
titanium	22	ti	4500.0	20.92	523
tungsten	74	w	19300.0	196.65	134
uranium	92	u	19100.0	26.443	117
vanadium	23	v	6100.0	30.794	460
ytterbium	70	yb	7000.0	8.368	149
yttrium	39	y	4470.0	12.552	293
zinc	30	zn	7140.0	111.71	389
zirconium	40	zr	6500.0	31.38	285