

© 2019 Andrei Rykhlevskii

FUEL PROCESSING SIMULATION TOOL FOR LIQUID-FUELED NUCLEAR REACTORS

BY

ANDREI RYKHLEVSKII

PRELIMINARY EXAMINATION

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, 2019

Urbana, Illinois

Committee:

Assistant Professor Kathryn Huff, Advisor
Associate Professor Tomasz Kozlowski
Professor James Stubbins
Professor Luke Olson

ABSTRACT

In the search for new ways to generate carbon-free, reliable base-load power, interest in advanced nuclear energy technologies, particularly Molten Salt Reactors (MSRs), has resurged with multiple new companies pursuing MSR commercialization. To further develop these MSR concepts, researchers need simulation tools for analyzing liquid-fueled MSR depletion and fuel processing. Current MSR modeling efforts in the literature usually assume ideal (e.g., 100% of neutron poison being removed) rather than realistically constrained removal efficiency. This work proposes to create a Python package, SaltProc, which will implement realistically constrained extraction efficiency of fission product based on physical models of fuel processing components appearing in various MSR systems. To demonstrate the capabilities of SaltProc, the Transatomic Power (TAP) concept will be simulated to capture the evolution of fuel salt composition during reactor operation with a realistic, physics-driven model of an online fuel reprocessing system. SaltProc will also be applied to determine the feasibility and safety of load following for the TAP system. This proposal outlines extensions to this work, including investigating safety parameters (temperature coefficients, shutdown margin, axial power offset) evolution over a 60-year reactor operation lifetime and during short-term load-following transients.

TABLE OF CONTENTS

LIST OF SYMBOLS	iv
CHAPTER 1 INTRODUCTION	1
1.1 Motivation	1
1.2 Fuel burnup and online reprocessing	3
1.3 Background Summary	10
CHAPTER 2 ONLINE REPROCESSING MODELING AND SAFETY ANALYSIS	12
2.1 Fuel salt reprocessing overview	12
2.2 Serpent overview	21
2.3 Proposed simulation tool design and capabilities	22
2.4 Preliminary results	27
CHAPTER 3 CODE DEMONSTRATION AND VALIDATION	33
3.1 Transatomic Power Molten Salt Reactor concept	33
3.2 TAP fuel salt reprocessing system	41
3.3 Preliminary results	45
CHAPTER 4 SAFETY ANALYSIS	55
4.1 Safety and operational parameters	55
4.2 Preliminary results	62
CHAPTER 5 FUTURE WORK AND PROPOSED SIMULATIONS	64
5.1 Summary	64
5.2 Stage 1: Basic online reprocessing demonstration	66
5.3 Stage 2: SaltProc v1.0 demonstration and validation for the TAP	66
5.4 Stage 3: Variable xenon extraction rate	68
5.5 Stage 4: Prototype design for the Xe removal system	70
5.6 Stage 5: TAP Safety Analysis	71
5.7 Conclusions	71
REFERENCES	73

LIST OF SYMBOLS

Σ_f Macroscopic fission cross section (cm^{-1})

ϕ Neutron flux ($\frac{neutron}{cm^2 \times s}$)

γ_i Fission yield of i^{th} nuclide

CHAPTER 1

INTRODUCTION

1.1 Motivation

Humankind has only a few ways to generate reliable, nonintermittent baseload power: fossil fuels, hydro-power, geothermal power, and nuclear energy. Because of increasing global climate change concerns, sources with negligible CO₂ footprints are crucial measures for global temperature control. Thus, from an environmental viewpoint, hydro and nuclear power are preferable ways to generate reliable power. Nevertheless, the potential for a hydro-power is strictly limited by local geographical conditions; hence, the only option left is nuclear power. Nuclear power plants provided 10% of global electricity supply in 2018 [1]. Moreover, nuclear share in energy generation is projected to stay constant through 2040, while electricity demand will increase by 30% [2].

The Generation IV International Forum (GIF) chose MSRs among the six advanced reactor concepts for further research and development. MSRs offer significant improvements “in the four broad areas of sustainability, economics, safety and reliability, and proliferation resistance and physical protection” [3]. To achieve the goals formulated by the GIF, MSRs simplify the reactor core and improve inherent safety by using liquid coolant which is also a fuel¹. In a thermal spectrum MSR, liquid fuel consists of carrier salt (i.e. LiF, LiF-BeF₂ or LiF-NaF-KF) and fluorides of fissile and/or fertile materials (i.e. UF₄, PuF₃ and/or ThF₄). This fuel circulates in a loop-type primary circuit [4]. This innovation leads to immediate advantages over traditional, solid-fueled, reactors. These include near-atmospheric pressure in the primary loop, relatively high coolant temperature, outstanding neutron economy, a high level of inherent safety, reduced fuel preprocessing, the ability to continuously remove

¹Herein MSRs are assumed to be reactors with liquid fuel which simultaneously serves as coolant.

fission products and add fissile and/or fertile elements without shutdown [5]. The possibility of continuously removing neutron poisons increases the potential fuel burnup and thus improves the resource utilization of MSRs. Finally, the MSR also could be employed for transmutation of spent fuel from current Light Water Reactors (LWRs) [6].

Recently, interest in MSRs has resurged, with multiple new companies pursuing commercialization of MSR designs². China’s MSR program was initiated in 2011 and promises to startup a 2MW_{th} liquid-fueled test MSR in 2020, a 10MW_{th} demonstration reactor in 2025, and a gigawatt-level commercial reactor in 2050 [7]. The European Union funds the Safety Assessment of the Molten Salt Fast Reactor (SAMOFAR) project, in which several European research institutes and universities are developing various molten salt reactor prototypes such as the Molten Salt Fast Reactor (MSFR) [8] and the Molten Salt Actinide Recycler and Transmuter (MOSART) [9]. To advance these MSR concepts, particularly concerning their strategies for online reprocessing and refueling, we need computational analysis methods capturing their unique reactor physics, fuel reprocessing mechanics, and chemistry.

The main objective of the proposed work is to develop the online reprocessing simulation package, SaltProc, which couples with the continuous-energy Monte Carlo depletion calculation code, Serpent 2 [10], for liquid-fueled MSR depletion simulations. Most of existing MSR depletion simulators usually assume 100% efficiency of the neutron poison removal process (see Chapter 1). The ultimate objective of this effort is to develop a generic open-source tool capable of simulating a wide range of liquid-fueled systems — including two-fluid and multi-region designs — and to validate it against existing modeling efforts. Moreover, SaltProc enables poison extraction simulation based on a realistic physics-based fuel processing model.

This document outlines the motivation, preliminary work, and future work proposed towards developing a simulation tool for analyzing fuel depletion in a liquid-fueled MSRs. Chapter 1 serves as a literature review, providing background on fuel burnup, online fuel reprocessing approaches, safety parameter evolution during reactor operation, and how these concepts have been applied to a wide range of MSRs in the literature. Chapter 2 covers

²Examples include liquid-fueled MSR designs from Terrapower, Terrestrial, ThorCon, Flibe, Copenhagen Atomics, Elysium, etc.

modeling online reprocessing details and proposed computation tool architecture. Chapter 3 explains the verification and validation (V&V) method, demonstration cases, and safety parameter evolution. Specifically, these demonstration and verification efforts will focus on the TAP MSR because it is well analyzed in the literature. Chapter 4 gives the safety parameter overview and outlines the plan for analyzing these parameters' evolution during TAP reactor lifetime. Finally, remaining future work and expected contributions to the nuclear community are summarized in Chapter 5.

1.2 Fuel burnup and online reprocessing

All liquid-fueled MSR designs involve varying levels of online fuel processing. Minimally, volatile gaseous fission products (e.g., Kr, Xe) escape from the fuel salt during routine reactor operation and must be captured. Additional systems might be used to enhance the removal of those elements. Most designs also call for the removal of rare earth metals from the core since these metals act as neutron poisons. Some designs suggest a more elaborate list of elements to process (figure 1.1), including the temporary removal of protactinium from the salt or other regulation of the actinide inventory in the fuel salt [11]. Fresh fuel salt with dissolved fissile and/or fertile material (e.g., ^{233}U , ^{232}Th , low-enriched uranium (LEU), a transuranic vector from LWR spent nuclear fuel (SNF)) make up the salt mass loss caused by poison removal and conserves the total mass in the primary loop.

Most liquid-fueled nuclear reactor concepts adopt nonintermittent separations and feeds: the core material is circulated to or from the core at all times (continuously) or specific intervals (batch-wise). In contrast, in a solid-fueled reactor, fission products and actinides remain within the initial fuel material during and after the operation until reprocessing. The ability to perform online fuel salt reprocessing improves the potential neutronics performance of liquid-fueled reactors. First, liquid-fueled reactors can operate with a relatively small excess reactivity because fissile material is continuously being added to the core. Second, continuously removing fission products, including strong absorbers (poisons) should significantly

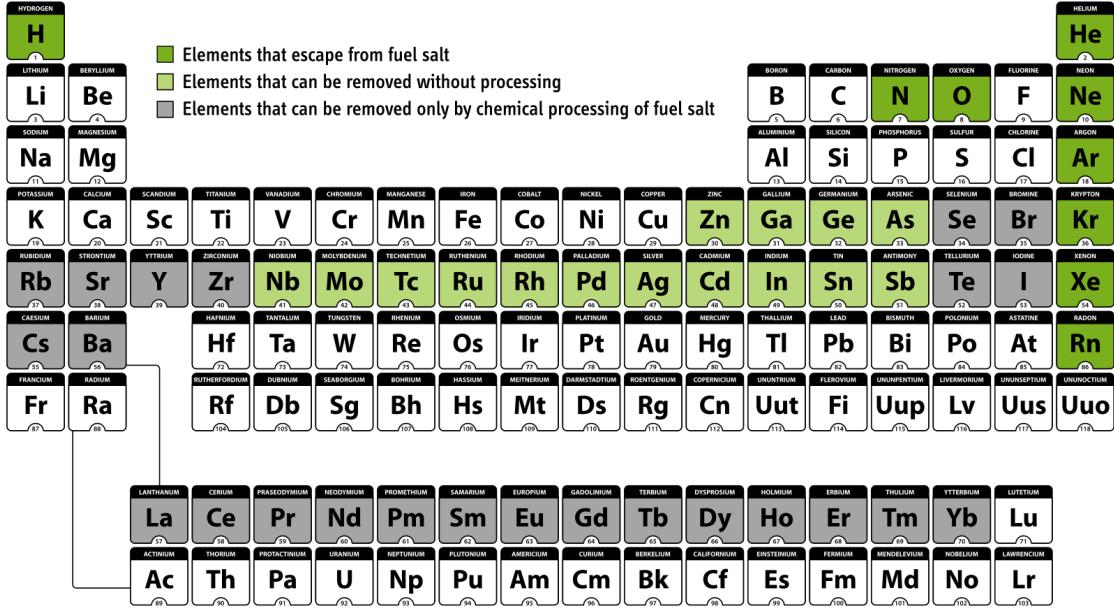


Figure 1.1: Processing options for MSR fuels (figure reproduced from Ahmed *et al.* [11]).

improve fuel utilization and decrease parasitic neutron absorption. Finally, for a breeder³ excess of fissile material might be continuously extracted from the core and used to start up new reactors. Nevertheless, the removal of each element from the liquid fuel salt presents a unique challenge in terms of chemical separation, storage, and disposal of the separated materials.

Continuous fuel salt reprocessing prevents the usage of most contemporary nuclear reactor fuel burnup software. To handle the material flows and potential online removal and feed of liquid-fueled systems, early MSR simulation methods at Oak Ridge National Laboratory (ORNL), which integrated neutronic and fuel cycle codes (i.e., Reactor Optimum Design (ROD) [12]) into operational plant tools (i.e., Multiregion Processing Plant (MRPP) [13]) for MSR fuel reprocessing system design. A summary of recent efforts is listed in table 1.1.

Two main online reprocessing simulation approaches are commonly used in the literature: continuous and batch-wise. In the batch-wise approach, the burnup simulation stops at a given time and restarts with a new liquid fuel composition (after removal of discarded materials and addition of fissile/fertile materials).

³conversion ratio (CR) \equiv fissile generated/fissile consumed: if CR < 1, the reactor is a “converter”; CR \equiv 1, an “isobreeder”; CR > 1, a “breeder.”

Table 1.1: Tools and methods for liquid-fueled MSRs fuel salt depletion analysis.

	Nuttin <i>et al.</i> , 2005 [14]	Aufiero <i>et al.</i> , 2013 [15]	Betzler <i>et al.</i> , 2018 [16]	Proposed work
Neutron transport software	MCNP	Serpent 2	SCALE6.2	Serpent 2
Neutron transport method		Monte Carlo continuous energy	Deterministic discrete ordinates	Monte Carlo continuous energy
Burnup software	REM	Serpent 2	ORIGEN-S	Serpent 2
Geometry model	unit cell	full-core 3D	unit cell	full-core 3D
FP removal/feed	continuous	continuous	batch-wise	batch-wise
Separation efficiency		fixed, must be defined by user before simulation		variable of many parameters
Fuel reprocessing plant		single component, “black” box model		realistic multi-component model
Reactivity control	continuous adjustment of fissile material injection	batch injection of fissile material		periodical adjustment of geometry and fissile material injection

Accounting for continuous removal or addition of material presents a greater challenge since it requires adding a term to the Bateman equations. In SCALE/TRITON, ORIGEN [17] solves a set of the Bateman equations using one-group averaged fluxes and cross sections obtained from a transport calculation. The Bateman equations describe the rate of change of each isotope, i , due to neutron induced reactions and decay processes [18]:

$$\frac{dN_i}{dt} = \sum_{m=1}^M l_{im} \lambda_m N_m + \phi \sum_{m=1}^M f_{im} \sigma_m N_m - (\lambda_i + \phi \sigma_i + r_i - f_i) N_i + F_i \quad i \in [1, M] \quad (1.1)$$

(1) (2) (3) (4) (5) (6)

where

N_i = atom density of nuclide i

M = number of nuclides

l_{im} = fraction of decays of nuclide m that result in formation of nuclide i

λ_i = radioactive decay constant of nuclide i

ϕ = neutron flux, averaged over position and energy

f_{im} = fraction of neutron absorption by nuclide m leading to the formation of nuclide i

σ_m = average neutron absorption cross section of nuclide m

r_i = continuous removal rate of nuclide i from the system

f_i = continuous feed rate of nuclide i

F_i = production rate of nuclide i directly from fission

The four terms on the right-hand side of the equation represent:

- (1) production of species i as a result of the decay of all the nuclides present;
- (2) production of species i as a result of neutron capture by all nuclides present;
- (3) loss of nuclide i through its own decay;
- (4) loss of nuclide i as a result of neutron capture;
- (5) loss of nuclide i through continuous removal from the system;
- (6) gain of nuclide i as a result of continuous feed to the system.

Recently, Nuttin *et al.* developed in-house depletion code REM which directly couples with MCNP [19] to simulate fuel salt material evolution in a simplified Molten Salt Breeder Reactor (MSBR)-like reactor. That work directly integrated the Bateman differential equations using neutron flux from the MCNP, tracking all the isotopes available in the data library, and control reactivity to maintain reactor critical [14].

In a similar vein, Aufiero *et al.* extended Serpent 2 for continuous reprocessing simulations by explicitly introducing “reprocessing” time constants into the system of Bateman equations and adding effective decay and transmutation terms for each nuclide [15]. The developed extension directly accounts for the effects of online fuel reprocessing on depletion calculations and features a reactivity control algorithm. The extended version of Serpent 2 was assessed

against a dedicated version of the deterministic ERANOS-based EQL3D procedure in [20] and applied to analyze the MSFR fuel salt isotopic evolution.

ORNL researchers have developed ChemTriton, a Python script for SCALE/TRITON which uses the batch-wise approach to simulate a continuous reprocessing and refill for either single or multiple fluid designs. ChemTriton models salt treatment, separations, discharge, and refill using a unit-cell MSR SCALE/TRITON depletion simulation over small time steps to simulate continuous reprocessing and deplete the fuel salt [21, 16].

Most of the existing tools represented fuel salt reprocessing plant as an invariable “black box” model which removes target elements all at once with a fixed efficiency, determined by the user before starting the depletion simulation. Typically, such a “black box” model is characterized by vector of removing elements and their extraction efficiencies:

$$\begin{bmatrix} N_0^{in} \\ \vdots \\ N_e^{in} \\ \vdots \\ N_E^{in} \end{bmatrix} \times \begin{bmatrix} \epsilon_0 \\ \vdots \\ \epsilon_e \\ \vdots \\ \epsilon_E \end{bmatrix} = \begin{bmatrix} N_0^{out} \\ \vdots \\ N_e^{out} \\ \vdots \\ N_E^{out} \end{bmatrix} \quad (1.2)$$

where $N^{in/out}$ is the number density of atoms and ϵ is the extraction efficiency for all elements e in $(0, E)$. The main issues related with static “black box” model assumptions in the literature include:

Time-independent separation efficiency vector. Realistically, long-term reactor operation will require a time-dependent extraction efficiency vector.

The separation efficiency is independent of the reactor operational parameters. In reality the extraction efficiency depends on temperature, power level, current fuel salt isotopic composition, and material mass flow rate.

All reprocessing plant components are treated as a single “black box” component. However, the fuel salt in a reprocessing plant undergoes many separate components

(e.g., helium bubbling, nickel mesh filter, etc.) which target specific elements. Some of these components can be connected in series, parallel, or series-parallel. The “black box” model (only single process) requires massive pre-simulation analytic work from the user to calculate lumped separation efficiency vector before a simulation is run and cannot be adjusted during the simulation. Additionally, treating the processing system as a single “black box” may lose dynamics. Finally, the waste stream from each component cannot be tracked separately, which is necessary for fuel reprocessing system optimization.

Some of the tools listed in table 1.1 used major approximations that may lead to inaccurate fuel evolution predictions, and others are not available for external users. This work proposes an open-source simulation package, SaltProc, which expands the capability of the continuous-energy Monte Carlo Burnup calculation code, Serpent 2, for simulation liquid-fueled MSR operation.

1.2.1 Operational and safety parameters evolution

In contrast with conventional solid-fueled reactors in which in-core fuel residence time is 4-5 years⁴, the initial fuel salt batch stays in the MSR reactor primary loop during the whole lifetime. Therefore, the fuel salt accumulates FPs not captured by fuel reprocessing system as well as transuranic elements⁵. Continuous fuel salt composition evolution has a significant influence on the neutron energy spectrum and, consequently, affects the reactor behavior, necessitating additional safety analysis.

Nuttin *et al.* studied evolution of a key safety parameter, the temperature reactivity feedback coefficient, estimating it for the MSBR at start-up and at equilibrium. The temperature coefficient of reactivity quantifies reactivity changes due to temperature increase in the core and was calculated in that work as:

$$\alpha = \frac{k_{1200} - k_{900}}{\delta T} \quad (1.3)$$

⁴For the most common 18-month cycle, during refueling personnel removing 1/3 of the fuel assemblies, re-arranging other assemblies, and loading fresh fuel into the core. Thus, each fuel assembly is kept in the core at most $3 \times 18 = 54$ months.

⁵The chemical elements with atomic numbers greater than uranium (92).

where

k_{900}, k_{1200} = the multiplication coefficients at 900K and 1200K, respectively

$$\delta T = 1200K - 900K.$$

That work showed that the fuel temperature coefficient (FTC) at start-up and at equilibrium is -1.5 and $-1.0 \text{pcm}/K$, respectively. Percent mille (*pcm*) is the unit of reactivity equal to 10^{-5} of k_{eff} . Nuttin *et al.* also reported a positive and time-invariant total temperature coefficient ($+0.8 \text{pcm}/K$) [14]. Recently, Park and colleagues expanded that approach to a full-core high-fidelity MSBR model and estimated safety parameters evolution over 20 years of operation [22]. These calculations showed relatively large negative total temperature coefficient during 20 years of the reactor operation; the coefficient magnitude weakens from -3.21 to $-1.41 \text{pcm}/K$ at start-up and at equilibrium composition, respectively. Additionally, that work reported control rod worth deterioration from 2099pcm to 1970pcm due to neutron spectrum hardening during reactor operation.

More recently, Betzler *et al.* [23] reported key safety parameters evolution for the TAP MSR: the fuel reactivity coefficient at Beginning of Life (BOL) and 15 years from BOL is negative and decreasing slowly over the reactor lifetime; the moderator reactivity coefficient is small and positive at BOL and became negative after 15 years of operation. Overall, thermal feedback seems to be stronger in the TAP reactor and deteriorates insignificantly during the reactor operation. Notably, the authors ignored material density change with temperature to simplify temperature coefficients calculation; thus, only Doppler broadening was taken into account. Finally, the researchers reported the total worth of all control rods in the TAP core at start-up only.

The evolution of control rod worth in the TAP has not been reported in the literature before. The proposed work will illuminate the evolution of major safety parameters (fuel, moderator and total temperature coefficient, void reactivity coefficient, control rod worth) for the TAP MSR at various moments during the reactor operation. Additionally, the impact of neutron poison accumulation (e.g., ^{135}Xe) in the fuel salt during short-term transients (i.e., load following) on safety characteristics will be investigated.

1.3 Background Summary

State-of-the-Art software packages for depletion analysis and evolution of safety parameters in liquid-fueled MSR are reviewed in this Chapter. Based on this summary, I have identified a few possible directions for the improvement of MSR tools:

Reproducibility/Availability. Serpent is the only contemporary nuclear contemporary nuclear reactor physics software which can perform depletion calculations that can take into account online fuel salt reprocessing regimes. However, this built-in online reprocessing routine is undocumented and the discussion forum for Serpent users is the only useful source of information at the moment. Other mentioned tools are under the closed-source license or available for internal users only. These issues can be a barrier to reuse of research software and to reproduce scientific results. Thus, a new, open-source, reproducible tool for fuel processing simulation would assist in the production of reproducible research in the area of liquid-fueled reactor modeling.

Realistic fuel reprocessing system model. Major approximations in fuel reprocessing parameters deteriorate fuel salt composition predictions since the evolution of safety parameter accuracy is strongly dependent on fuel salt composition. A realistic fuel reprocessing system model will allow reprocessing component parameter optimization, increase fidelity of fuel and waste stream composition calculations, and advance reprocessing system design.

Variable extraction efficiency. Most of the research efforts in the literature assumed ideal 100% extraction efficiency of all removed elements, which stayed constant during the whole reactor lifetime. But realistically the efficiency is time-dependent and changes with respect to operational parameters: temperature, power level, salt composition, etc. Thus, the ability to set up dynamic separation efficiency must be added in MSR simulation tools to advance depletion calculations.

Reactivity control. Reconfigurable moderator configuration in the TAP core presents a challenge because of the core geometry changes with respect to time. The reactivity control

module which adjusts the core geometry to maintain criticality would be a great capability for simulating a new, more advanced MSR concepts and short-term transients.

Safety characteristics evolution during reactor operation. The MSR fuel salt accumulates FP and transuranic elements which significantly shift neutron energy spectrum. Neutron energy hardening might worsen the core safety during operation. The impact of the fuel salt evolution on the MSR safety parameters must be carefully investigated and reported.

The proposed work will hopefully overcome these issues and demonstrate the tool capabilities for promising TAP MSR concept.

CHAPTER 2

ONLINE REPROCESSING MODELING AND SAFETY ANALYSIS

2.1 Fuel salt reprocessing overview

Removing specific chemical elements from a molten salt is a complicated task that requires intelligent design (e.g., chemical separations equipment design, fuel salt flows to equipment). This section contains a brief overview of a generic MSR fuel salt reprocessing system. Modeling such a system is the focus of the proposed work.

2.1.1 Gas separation system

Gaseous fission products (e.g., Kr, Xe) must be removed from the fuel salt to avoid reactor poisoning, especially during startup and power maneuvering. This is particularly true for ^{135}Xe , with its extensive neutron capture cross section ($\approx 10^6 \dots 10^7 \text{ b}$ in a thermal energy range). ^{135}Xe is produced directly from fission in about 0.3% of ^{235}U fissions ($\gamma_{^{135}\text{Xe}}$), but an even larger fraction of ^{135}Xe is produced by the decay of ^{135}I and ^{135}Te (table 2.1). ^{135}I and ^{135}Te yields from fission are $\gamma_{^{135}\text{I}} = 3.6\%$ and $\gamma_{^{135}\text{Te}} = 2.5\%$, respectively. Thus, total ^{135}Xe production from fission is about 6.4% of fissions (of ^{235}U), most of this is from ^{135}I and ^{135}Te decay. Noble gases (e.g. tritium, xenon, and krypton) can be removed from the fuel salt as follows:

- (a) a bubble generator injects helium bubbles in the salt stream;
- (b) noble gases migrate promptly to the helium bubbles because of their extreme insolubility in the salt [25];
- (c) and a gas separator discharges the fission-product-rich bubbles from the salt to the

off-gas system.

Diagram 2.1 shows the key pathways for xenon production, accumulation, and removal in a typical MSR.

Table 2.1: ^{135}Xe production sources and principal rate constants involved (reproduced from Kedl *et al.* [26]).

^{135}Xe gain mechanism	Principal rate parameters involved
Direct from fission yield $\gamma_{^{135}\text{Xe}} = 0.003$	$\Sigma_f \gamma_{^{135}\text{Xe}} \phi$ (for ^{235}U fission)
^{135}I decay yield $\gamma_{^{135}\text{Xe}} = 0.036$, it decays to ^{135}Xe with $\tau_{1/2} = 6.68 \text{ h}$	$\Sigma_f \gamma_{^{135}\text{I}} \phi$ (for ^{235}U fission)
^{135}Te decay yield $\gamma_{^{135}\text{Xe}} = 0.025$, it decays to ^{135}I with $\tau_{1/2} = 19 \text{ s}$	$\Sigma_f \gamma_{^{135}\text{Te}} \phi$ (for ^{235}U fission)

Figure 2.2 shows the principal design of the MSBR gas separation system. Helium bubbles of a specific size are introduced in a salt stream via the primary pump bowl. These bubbles absorb noble gases before being separated from the salt by a gas separator. ORNL suggested that the MSBR off-gas system would inject $d = 0.508\text{mm}$ helium bubbles in the pump bowl, redirect 10% of the fuel salt flow through a bubble separator to remove the bubbles, and then return the flow back into the pump suction. Robertson *et al.* reported that helium bubble size was approximately 25% of the throat width (blue circle on figure 2.3) and was independent of the gas flow rate [25]. Consequently, it is possible to regulate the helium bubble size by changing the throat width in the bubble generator.

To realistically model the gas separation system, a mathematical model describing noble gas extraction efficiency dynamics during reactor operation is required. Particularly, a model of xenon extraction efficiency as a function of sparger design parameters is needed to accurately model ^{135}Xe removal in a fuel salt depletion simulation. The gain and loss terms for ^{135}Xe dissolved in the fuel salt are listed in Tables 2.1 and 2.2. The removal efficiency for the xenon in the pump bowl was measured during Molten Salt Reactor Experiment (MSRE) operation, but the technical report ORNL-4069 by Kedl-Houtzeel only stated its range (from 50% to 100%) and concluded, “It is probably a complex parameter like the circulating-void

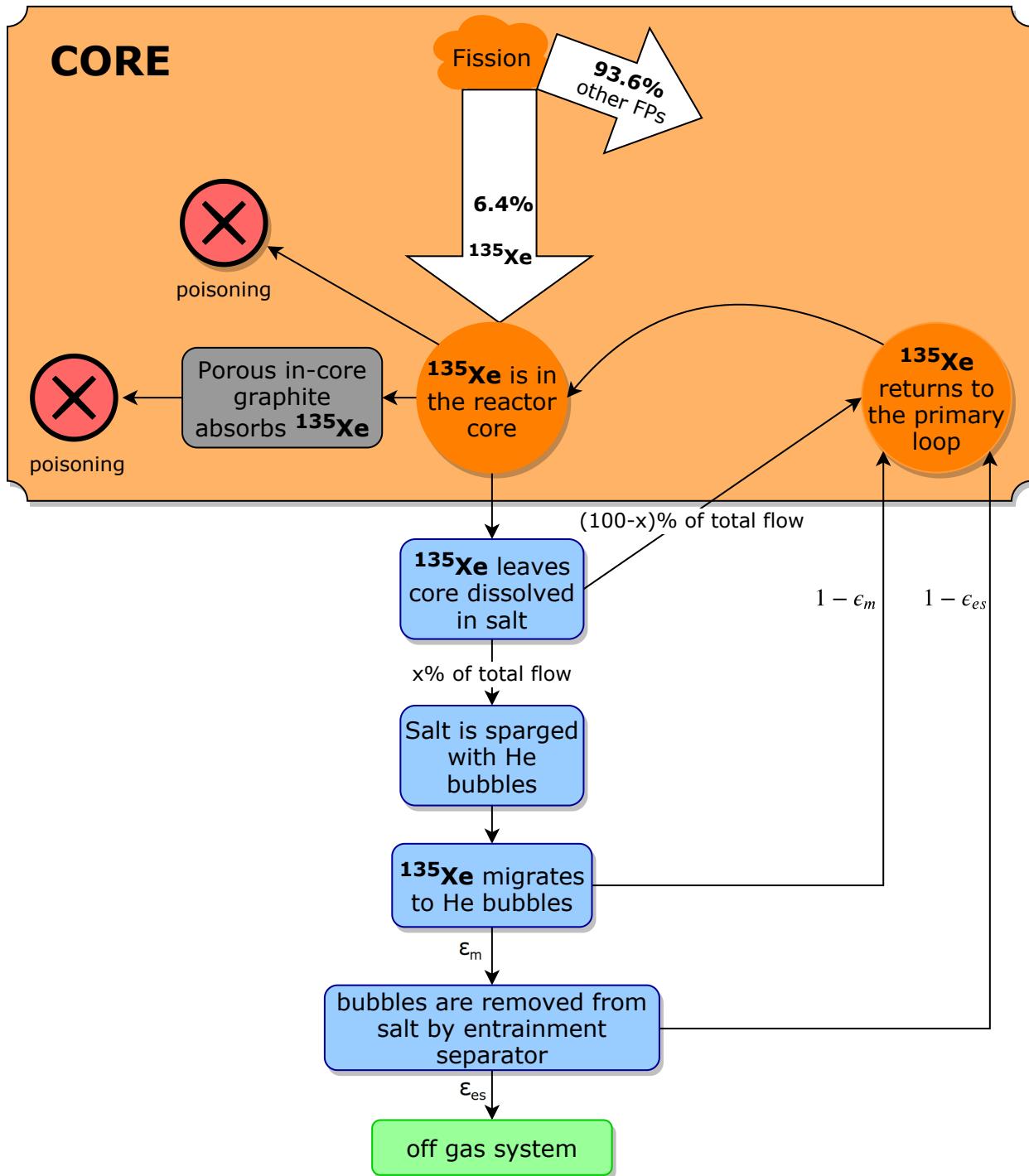


Figure 2.1: Schematic of ^{135}Xe circulation in a generic MSR. x is the fraction of fuel salt flow from the pump discharge redirected to the gas separation system, while ϵ_m and ϵ_{es} are the efficiencies of migration (of ^{135}Xe to the helium bubbles in the sparger) and separation (of gas in the entrainment separator). The orange color represents the fuel salt in the primary loop, the blue color represents the gas separation system, and the gray color is moderator in the core. Fission yields assume ^{235}U fission only.

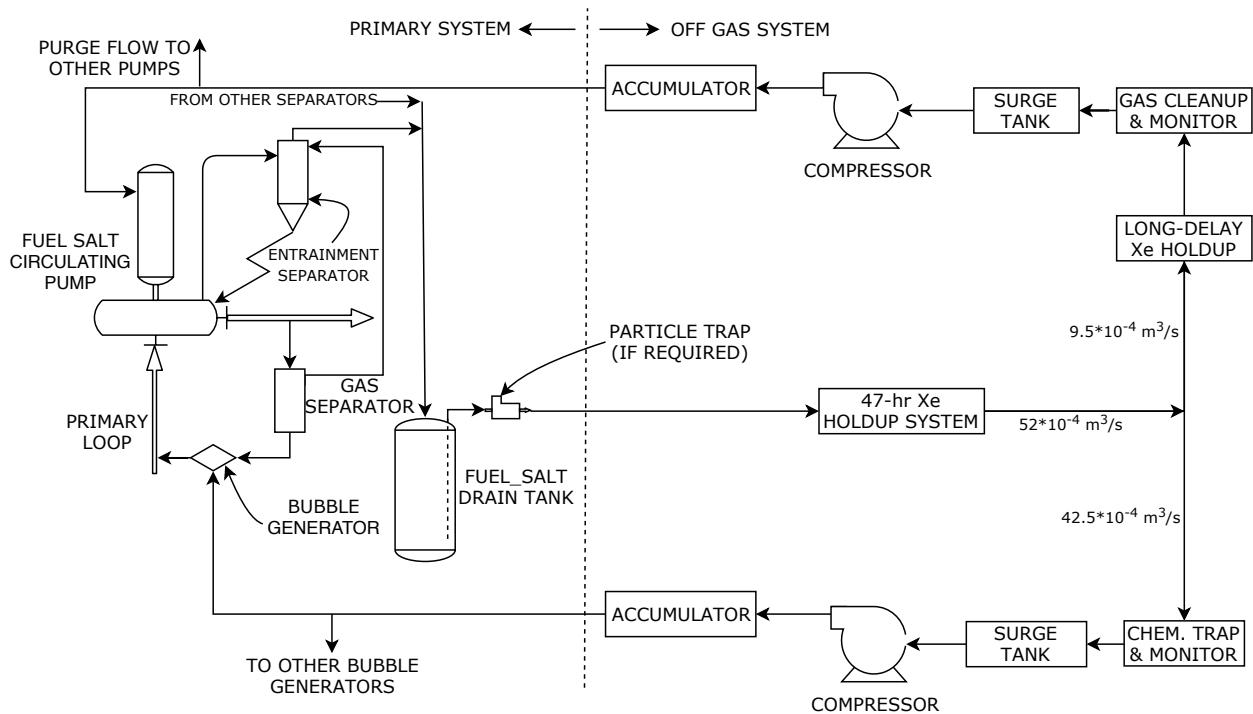


Figure 2.2: Schematic flow diagram of the MSBR gas separation system (figure reproduced from Robertson *et al.* [25]).

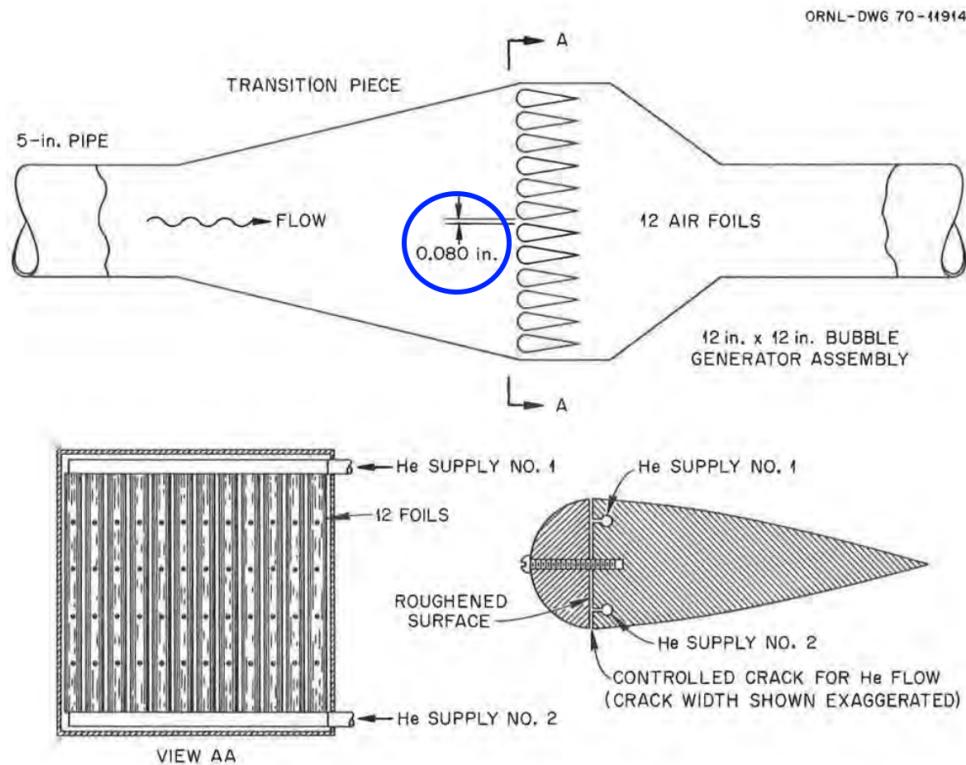


Figure 2.3: Preliminary concept of MSBR bubble generator (figure reproduced from Robertson *et al.* [25]). The blue circle shows throat width, which determines bubble size.

fraction and depends on many reactor operational variables.” [26]. ^{135}Xe burnup and decay rates are well known.

Peebles *et al.* in ORNL-TM-2245 has reported xenon removal efficiency (ϵ_{Xe}) in a gas separation system as a function of many parameters [28]:

$$\epsilon_{\text{Xe}} = \frac{1 - e^{-\beta}}{1 + \alpha} \quad (2.1)$$

where

$$\alpha = \frac{RTQ_{\text{salt}}}{HQ_{\text{He}}} \quad (2.2)$$

$$\beta = \frac{K_L a A_C L (1 + \alpha)}{Q_{\text{salt}}} \quad (2.3)$$

R = universal gas constant

T = salt temperature

Q_{salt} = volumetric salt flow rate

Q_{He} = volumetric helium flow rate

H = Henry's law constant for solute gas

a = gas-liquid interfacial area

A_C = contactor cross section

L = contactor length

Table 2.2: ^{135}Xe loss terms and principal rate constants involved (reproduced from Kedl *et al.* [26]).

^{135}Xe loss mechanism	Principal rate parameters involved
Decay of dissolved ^{135}Xe ($\tau_{1/2} = 9.1$ h)	Decay constant (λ)
^{135}Xe burnup dissolved xenon-135 burnup as it passes through core	Neutron flux (ϕ)
^{135}Xe migrated to helium bubbles	Removal efficiency (ϵ_m)
^{135}Xe transferred into circulating He bubbles; this xenon will eventually be burnup, decay, or stripped via bubble separator	Mass transfer coefficient (h), decay constant (λ), neutron flux (ϕ), bubble removal efficiency (ϵ_{es})

K_L = liquid phase mass transfer coefficient.

Most of the input parameters for that correlation are obvious and easy to obtain from the system component design. The mass transfer coefficient for transferring xenon into helium bubbles (K_L) can be estimated experimentally, but published information is currently insufficient to inform an accurate mathematical model appropriate for Computational Fluid Dynamics (CFD). Thus, Peebles *et al.* reported the mass transfer coefficient correlation for the MSBR salt (LiF-BeF₂-ThF₄-UF₄) but for a limited case. While it is out of the scope of this work to accurately estimate mass transfer coefficient, this work seeks to provide a tool which will allow the user to specify any mathematical model for a separation efficiency.

Equation 2.1 would apply to other noble gases (e.g., Kr) but the mass transfer coefficients would be different. Current effort at the University of Illinois at Urbana-Champaign namely, “Enabling Load Following Capability in the Transatomic Power MSR” [29], has a goal to determine mass transfer coefficients for various gaseous fission products using experiments, enabling CFD and multiphysics simulations of such reactors. As a result, the obtained mathematical model for gas removal efficiency will be used to inform a realistic physics-based fuel reprocessing model in the proposed SaltProc tool.

2.1.2 Fuel chemical processing facility

In addition to noble gases, the fuel salt reprocessing system must extract other FPs such as noble metals, semi-noble metals, and lanthanides. These absorb fewer neutrons than ¹³⁵Xe, but their removal is crucial to guarantee normal operation. Some fraction of noble and semi-noble solid fission products plate out onto the internal surfaces of the primary loop equipment, complicating their removal [30]. Meanwhile, lanthanides have relatively high solubility in the carrier salt and must be removed by chemical extraction.

In thorium-fueled MSR designs, ²³²Th in the fuel salt absorbs thermal neutrons and produces ²³³Pa which then decays into the fissile ²³³U (figure 2.4. Protactinium presents a challenge, since it has a large absorption cross section in the thermal energy spectrum. Accordingly, ²³³Pa is continuously removed from the fuel salt into a protactinium decay tank to

allow ^{233}Pa to decay to ^{233}U without poisoning the reactor. This feature allows the thorium-fueled MSR to avoid neutron losses to protactinium, keeps fission products to a very low level, and increases the efficiency of ^{233}U breeding.

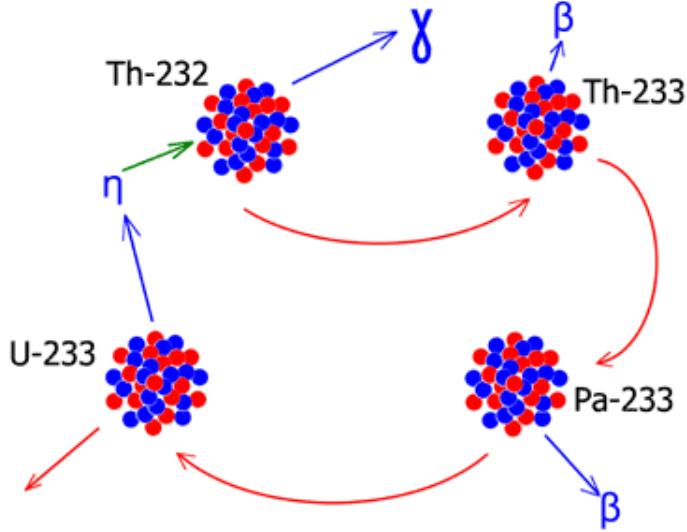


Figure 2.4: Production of ^{233}U from ^{232}Th .

Many authors report that a liquid-liquid reductive extraction process is the best option for removing protactinium and soluble fission products from molten fluoride salts [27, 31, 32]. In that process, the protactinium or lanthanides can be selectively stripped from the salt into liquid bismuth due to different chemical potentials. Moreover, the MSRE experience indicated that the extraction could be carried out rapidly and continuously [33].

The principal scheme of the MSBR reprocessing facility concept is shown in Figure 2.5. The fuel salt is first temporarily stored for cooling and decay of the shortest-lived fission products, then it is directed to the primary fluorinator. There, most of the uranium is removed by fluorination to UF_6 . After that, the salt is routed to an extraction column where it is combined with a mixture containing metallic bismuth, lithium, and thorium reductants. The remaining uranium and protactinium is reductively extracted to a bismuth solution, leaving a salt that only contains fission products dissolved in carrier salt (base composition $\text{LiF}-\text{BeF}_2-\text{ThF}_4$). The salt then goes through a reduction column where UF_6 is reduced to UF_4 preparing it for return to the reactor. BeF_2 and ThF_4 are also added, and all residual bismuth is removed from the salt. After a final cleanup step and valence

adjustment, the purified salt returns to the reactor [34, 35].

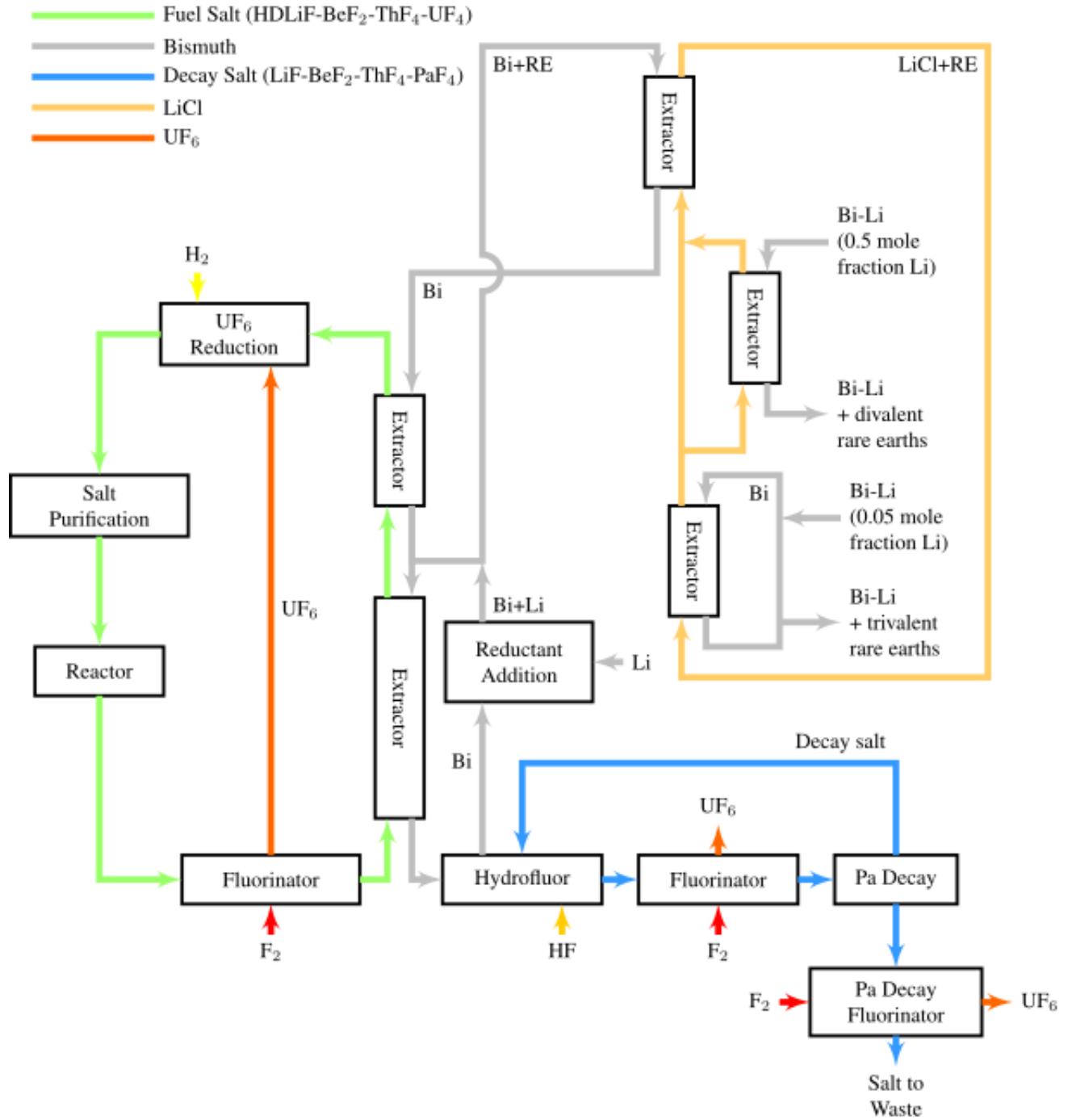


Figure 2.5: Simplified block diagram of chemical processing scheme for single-fluid MSBR (reproduced from Sorensen [35]). *RE* represents the rare earth elements extracted from the salt.

The bismuth accommodating some uranium and protactinium is routed to a hydrofluori-

nation column where metallic solutes in the bismuth are oxidized into their fluoride forms in the presence of a decay salt¹. The decay salt, containing UF_4 , PaF_4 , and ThF_4 , passes into a decay tank where ^{233}Pa is decays to ^{233}U . The uranium generated by protactinium decay is removed through fluorination to UF_6 and directed to the reduction column to refuel the purified fuel salt. A hydrofluorinator and a fluorinator can remove approximately 95% of the uranium from the stream [25].

The fully processed salt, on its way back to the reactor, has uranium added from the protactinium decay tank at the rate required to maintain or adjust the uranium concentration in the reactor (and, consequently, control the reactivity). Adding fissile material is performed by sparging the salt with UF_6 and hydrogen to produce UF_4 in the salt and HF gas [25].

After these separation steps, the fuel salt stream from the protactinium isolation system contains only traces of protactinium and uranium but contains practically all of the rare earths. A fraction of this salt stream is redirected to a reductive extraction process for removing rare earths. The principal scheme of a rare earth removal system is shown in Figure 2.6. A molten salt flow which contains rare earth fluorides is fed to the center of an extraction column. The salt flows countercurrent to a liquid bismuth stream which contains thorium and lithium. In the upper part of the column, the rare earths are reduced and transferred to the downflowing liquid metal stream. Below the feed point, the rare earth concentration is increased in the salt and metal streams in order to produce a concentration high enough for disposal [27].

While it is out of the scope of proposed work to derive the accurate chemistry-based mathematical formula for FPs separation efficiency, this work seeks to provide a flexible tool that will be able to simulate chemical processes in significant detail with regard to key system design parameters.

¹The decay salt contains UF_4 , PaF_4 , ThF_4 and fission products. Uranium produced after ^{233}Pa decay is extracted and directed back into the reactor. Decay salt is the precursor for the waste salt as it was periodically discarded every 220 days.

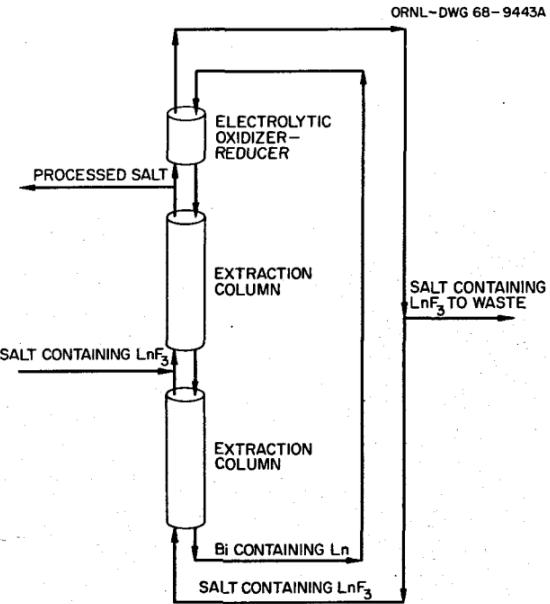


Figure 2.6: Rare earth removal from a fuel salt by reductive extraction (figure reproduced from Briggs *et al.* [27]).

2.2 Serpent overview

Serpent is a continuous-energy Monte Carlo neutronics software capable of solving the neutron transport problem by tracking individual neutrons within the problem geometry and using stochastic method to determine chain of events for each neutron [10]. Serpent has been under active development at the VTT Technical Research Centre of Finland since 2004, where it was initially conceived as a tool to simplify group constant generation in a high-fidelity Monte Carlo environment. Serpent is now a widely used transport code, used by more than 500 registered individuals in 155 organizations located in 37 countries around the world. The burnup calculation capability in Serpent is based on built-in calculation routines, without using any external solvers. A restart feature enables fuel shuffling simulation or applying any modifications to the input by dividing the calculation into several parts, which is crucial for online reprocessing simulations.

The latest version, Serpent 2, supports advanced geometries and has advanced burnup capabilities, including online refueling capabilities which are necessary for neutronic computations of pebble-bed reactors and liquid-fueled MSRs [15]. Unfortunately, built-in online refueling features are still under active development and unavailable to ordinary users. Also,

multi-physics simulations using Serpent 2 have been demonstrated, including calculations with thermal-hydraulics, CFD and fuel performance codes [36].

Serpent 2 can be effectively run in parallel on computer clusters and multi-core workstations. Parallelization is handled by thread-based OpenMP, which enables all processors to use shared memory space. Calculations can be divided into several nodes by distributed-memory Message Passing Interface (MPI) parallelization. Serpent 2 is an improvement upon Serpent 1, and contains a complete redesign of memory management using hybrid OpenMP [37] + MPI parallelization. This hybrid parallelization is important in depletion calculations using computer clusters with multiple nodes, and allows to achieve significant speed-up in depletion calculations on computer clusters with more than 4,000 cores [10].

All calculations herein were performed using Serpent 2 version 2.1.31 on Blue Waters XE6 nodes. For cross section generation, the JEFF-3.1.2 nuclear data library was employed based on entirely open cross section data [38].

2.3 Proposed simulation tool design and capabilities

The first version of the SaltProc Python tool for calculating MSR fuel composition evolution, taking into account an online reprocessing system was developed in 2018 as a part of the M.S. thesis preceding this proposal [39, 40]. The tool was designed to expand Serpent 2 depletion capabilities for modeling liquid-fueled MSRs with online fuel reprocessing system. SaltProc v0.1 uses HDF5 [41] to store data and uses the PyNE Nuclear Engineering Toolkit [42] for Serpent 2 output file parsing and nuclide naming. SaltProc v0.1 is an open-source Python package that uses a batch-wise approach to simulate continuous feeds and removals in MSRs.

SaltProc v0.1 only allows 100% separation efficiency for either specific elements or groups of elements at the end of the specific “cycle time”². Capabilities of the developed tool, working with the Monte Carlo software Serpent 2, were demonstrated using the full-core MSBR design for a simplified case with ideal removal efficiency (100% of mass for target elements

²The MSBR program defined “cycle time” as the time required to remove 100% of a target nuclide from a fuel salt [25].

removed) [43]. The SaltProc v0.1 architecture and the principal structure was not designed for flexible implementation of sophisticated online reprocessing systems, including realistic variable extraction efficiencies. Proposed improvements, discussed below, will correct this.

For the proposed work, SaltProc v0.1 will be completely refactored using Object-Oriented Programming (OOP) to create a comprehensive generic tool to realistically model complex MSR fuel reprocessing systems while taking into account variable extraction efficiencies, time-dependent core geometry, and the mass balance between the core and the reprocessing plant.

2.3.1 Proposed software architecture

The SaltProc v1.0 Python toolkit will couple directly with Serpent 2 input and output files, to couple the reprocessing system to depletion calculation. Python 3 OOP standard features will be used to create a flexible, user-friendly tool with great potential for further improvement and collaboration. Figure 2.7 shows the proposed SaltProc v1.0 class structure which includes 4 main classes:

Depcode. *Depcode* class contains attributes and methods for reading the user's input file for the depletion software, initial material (e.g., fuel and/or fertile salt) composition, principal parameters for burnup simulation (e.g., neutron population and number of cycles for Monte Carlo neutron transport), and running the depletion code.

Simulation. *Simulation* class runs Serpent depletion step, creates and writes HDF5 database, tracks time and converts isotopic composition vector nuclide names from Serpent to human-readable format.

MaterialFlow. Each *MaterialFlow* object represents the material flowing between *Process* objects (figure 2.8). All instances of this class contain an isotopic composition vector stored in PyNE Material object, mass flow rate, temperature, density, volume, and void fraction. Existing PyNE Material capabilities convert the units of the isotopic composition vector (e.g., from the atomic density provided by Serpent to a mass fraction or absolute mass in

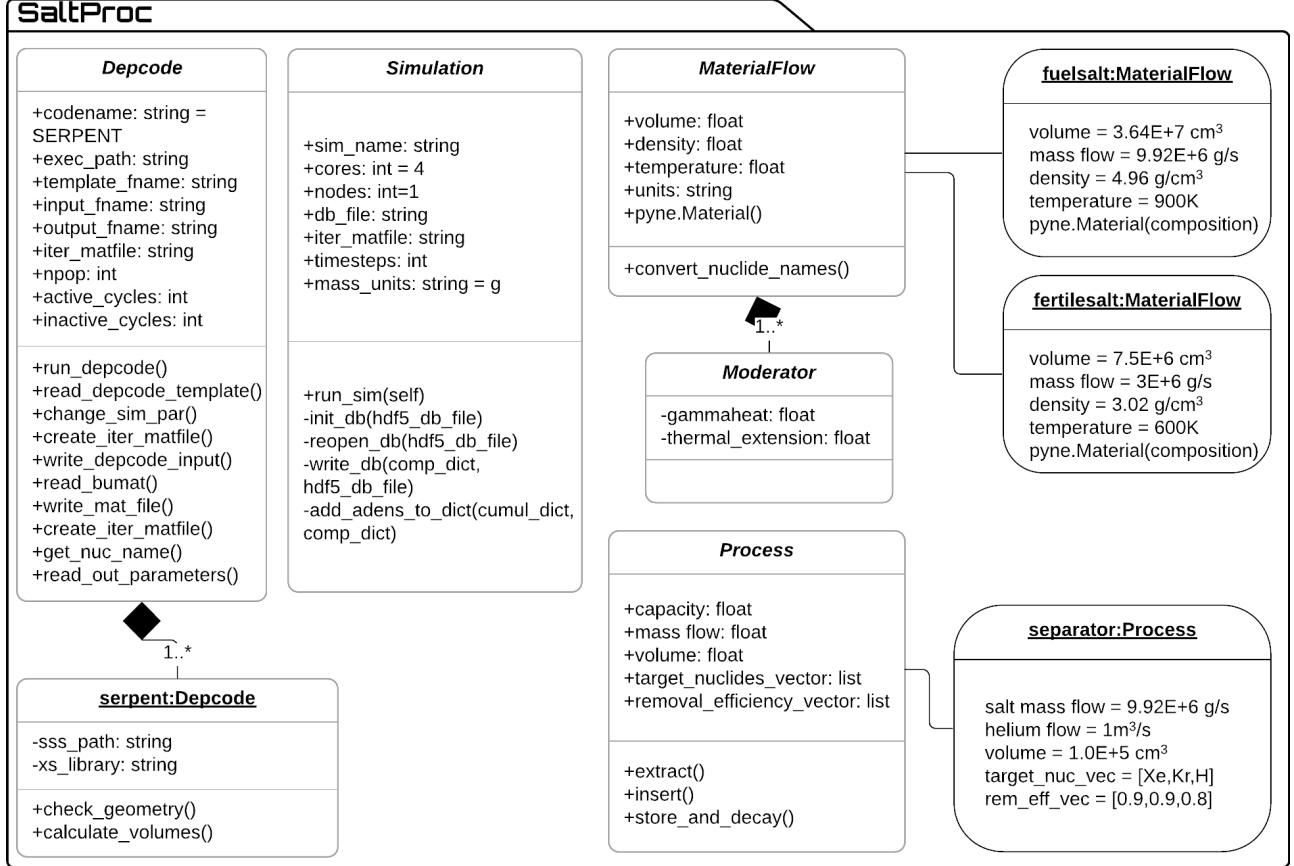


Figure 2.7: SaltProc v1.0 python package class diagram in UML notation with examples of object instances.

desired units) and decay the material (i.e., model the MSBR protactinium decay tank). The main idea of the *MaterialFlow* object is to pass detailed information about the salt starting at the MSR vessel outlet throughout reprocessing components (*Processes*), which modify the *MaterialFlow* object before depleting the material in the next Serpent burnup step.

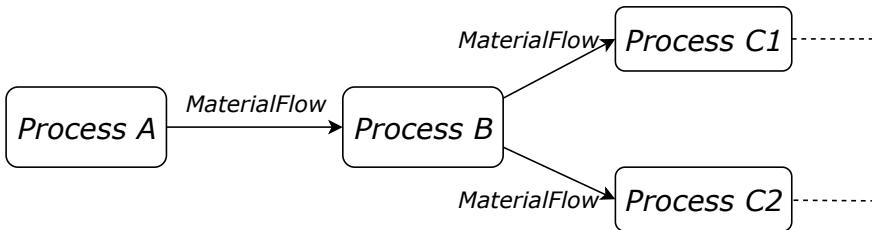


Figure 2.8: Schematic for passing material data between fuel processing system components.

Process. Each *Process* object represents a realistic fuel processing step characterized by its throughput rate, volumetric capacity, extraction efficiency for each target element (can be a function of many parameters), waste streams, and other process-specific parameters. Feed *Process* injects fresh fuel salt *MaterialFlow* directly into the reactor core (e.g., adding fissile material with a specific mass flow rate to *MaterialFlow* after performing all removals).

The proposed class structure provides outstanding flexibility in simulating various MSR fuel processing system designs. A library of various *MaterialFlow* (e.g., fuel salt flow, fertile salt flow, refueling salt flow) and *Process* (e.g., helium sparging facility, gas separator, lanthanide removal component) objects will be created to allow a user to quickly create a model of a desired reprocessing scheme. At runtime, the user will connect *Process* objects in series or parallel with *MaterialFlow* objects to form a comprehensive reprocessing system. The user will also be able to create custom objects with desired attributes and methods and contribute back to the code package using GitHub (<https://github.com/arfc/saltproc>).

2.3.2 Tentative flowchart

Figure 2.9 illustrates the proposed online reprocessing simulation algorithm coupling SaltProc v1.0 and Serpent. To perform a depletion step, SaltProc v1.0 reads a user-defined Serpent template file. This file contains input parameters such as geometry, material, isotopic composition, neutron population, criticality cycles, total heating power, and boundary conditions. SaltProc v1.0 fills in the template file and runs Serpent single-step depletion. After the depletion calculation, SaltProc v1.0 reads the depleted fuel composition file into the *MaterialFlow* object (*core_outlet* in figure 2.9). This object contains an isotopic composition vector, total volume of material, total mass, mass flow rate, density, temperature, void fraction, etc. For the simplest reprocessing case, when all fuel processing components are located in-line (100% of total material flow goes through a chain of separation components), the *core_outlet* object is flowing sequentially between *Processes* and each *Process* is removing a mass fraction of target elements with specified extraction efficiency. Afterward, the removed material mass is compensated by fresh fuel salt to maintain the salt inventory in

a primary loop. Finally, resulting isotopic composition after reprocessing is stored in HDF5 database and dumped in a new composition file for the next Serpent depletion run. SaltProc v1.0 also stores in database isotopic composition before reprocessing and waste stream from each fuel processing component.

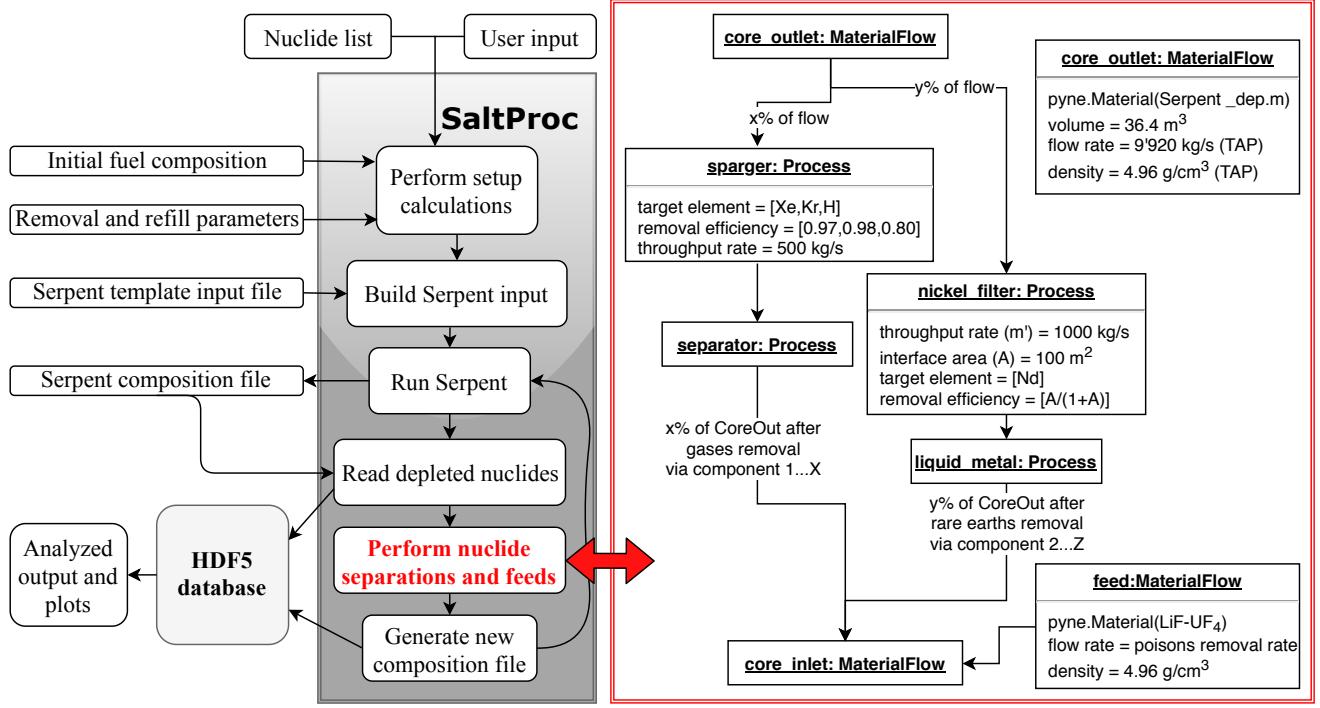


Figure 2.9: Tentative generic flow chart for SaltProc v1.0 python package.

For a more general case with multiple concurrent extraction processes, a separate *MaterialFlow* object is created for each branch with a user-defined mass flow rate (e.g., 90% of total mass flow rate flows via left branch and 10% throughout a right branch). The total mass and isotopic composition vector for each *MaterialFlow* object is calculated as a fraction of incoming *core_outlet* flow. Then each *MaterialFlow* object is passed via a cascade of *Processes* to separate selected chemical elements with specific efficiency. Finally, the left-hand-side branch *MaterialFlow* object is merged with the right-hand-side and similarly to the previous case, fresh fuel salt feed compensates the loss of mass in separation facilities and keep fuel salt mass in a primary loop constant.

The class diagram (Figure 2.7) allows to model the operation of a complex, multi-zone, multi-fluid MSR and is sufficiently general to represent myriad reactor systems. The refac-

tored version of SaltProc will only store and edit the isotopic composition of the fuel stream, which makes it a flexible tool to model any geometry: an infinite medium, a unit cell, a multi-zone simplified assembly, or a full core. This flexibility allows the user to perform simulations of varying fidelity and computational intensity. SaltProc v1.0 is an open-source tool (but a user needs Serpent 2.1.31 installed to use SaltProc v1.0), available on Github. It leverage unit and continuous tests crucial for sustainable development [44]. It will also have documentation generated through Sphinx, a documentation generator, for ease of use [45]. In summary, the development approach of SaltProc v1.0 is focused on producing a generic, flexible and expandable tool to give the Serpent 2 Monte Carlo code the ability to conduct advanced in-reactor fuel cycle analysis as well as simulate many online refueling and fuel reprocessing systems.

2.4 Preliminary results

Developed as a part of my master thesis, the first version of the tool only was able to leverage ideal removals (e.g., 100% of target isotope mass extracted). The capabilities of SaltProc were demonstrated for a full-core model of the MSBR model [46, 43]. Subsection 2.4.1 summarized that preliminary work and results obtained from applying the previous version of SaltProc to the MSBR core and reprocessing plant.

2.4.1 MSBR online reprocessing analysis

The MSBR vessel has a diameter of 680 cm and a height of 610 cm. It contains a molten fluoride fuel-salt mixture that generates heat in the active core region and transports that heat to the primary heat exchanger by way of the primary salt pump. In the active core region, the fuel salt flows through channels in moderating and reflecting graphite blocks. Figure 2.10 shows the configuration of the MSBR vessel, including the “fission” (zone I) and “breeding” (zone II) regions inside the vessel. The core has two radial zones bounded by a solid cylindrical graphite reflector and the vessel wall. The central zone, zone I, in which 13% of the volume is fuel salt and 87% graphite, is composed of 1,320 graphite cells,

2 graphite control rods, and 2 safety³ rods. The under-moderated zone, zone II, with 37% of fuel salt, and the radial reflector, surrounds the zone I core region and serves to diminish neutron leakage. Zones I and II are surrounded radially and axially by fuel salt (figure 2.11). This space for fuel is necessary for injection and flow of molten salt.

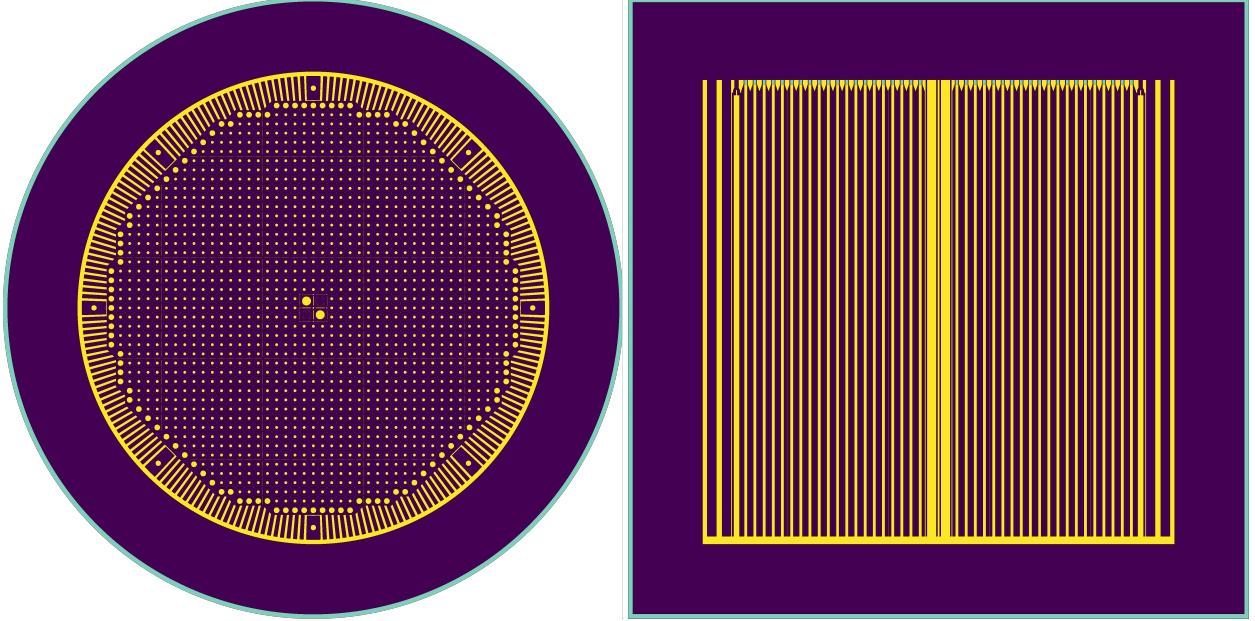


Figure 2.10: XY (left) and XZ (right) views of Serpent MSBR model (figure reproduced from Rykhlevskii *et al.* [46]).

As mentioned in section 2.1.2, the MSBR design requires online reprocessing to remove neutron gaseous FPs (Xe, Kr), noble metals (e.g., Se, Nb, Mo) and ^{233}Pa every 20 seconds. Table 2.3 summarizes a full list of nuclides and the cycle time used for modeling salt treatment and separations [25]. The removal rates vary among nuclides in this reactor concept and dictate the necessary resolution of depletion calculations. If the depletion time intervals are very short, an enormous number of depletion steps are required to obtain the equilibrium composition. On the other hand, if the depletion calculation time interval is too long, the impact of short-lived fission products is not captured. To compromise, a 3- day time interval was selected for depletion calculations to correlate with the removal interval of ^{233}Pa , and ^{232}Th was continuously added to maintain the initial mass fraction of ^{232}Th .

Figures 2.12 and 2.13 show the effective multiplication factors obtained using SaltProc

³These rods needed for emergency shutdown only.

Table 2.3: The cycle times for protactinium and fission products removal from the MSBR (reproduced from Robertson *et al.* [25]).

Processing group	Nuclides	Cycle time (at full power)
Rare earths	Y, La, Ce, Pr, Nd, Pm, Sm, Gd	50 days
	Eu	500 days
Noble metals	Se, Nb, Mo, Tc, Ru, Rh, Pd, Ag, Sb, Te	20 sec
Seminoble metals	Zr, Cd, In, Sn	200 days
Gases	Kr, Xe	20 sec
Volatile fluorides	Br, I	60 days
Discard	Rb, Sr, Cs, Ba	3435 days
Protactinium	^{233}Pa	3 days
Higher nuclides	^{237}Np , ^{242}Pu	16 years

v0.1 and Serpent. The effective multiplication factors were calculated after removing fission products listed in Table 2.3 and adding the fertile material at the end of cycle time (3 days for this work). The effective multiplication factor fluctuates significantly as a result of the batch-wise nature of this online reprocessing strategy.

First, Serpent calculates the effective multiplication factor for the beginning of the cycle (there is fresh fuel composition at the first step). Next, it computes the new fuel salt com-

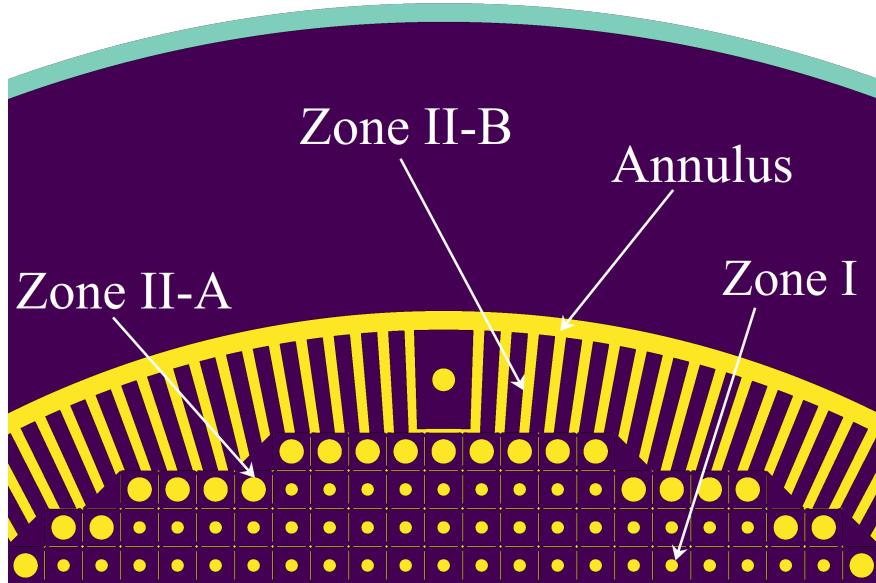


Figure 2.11: Detailed view of MSBR two zone model. Yellow represents fuel salt, purple represents graphite, and aqua represents the reactor vessel. Figure reproduced from Rykhlevskii *et al.* [46].

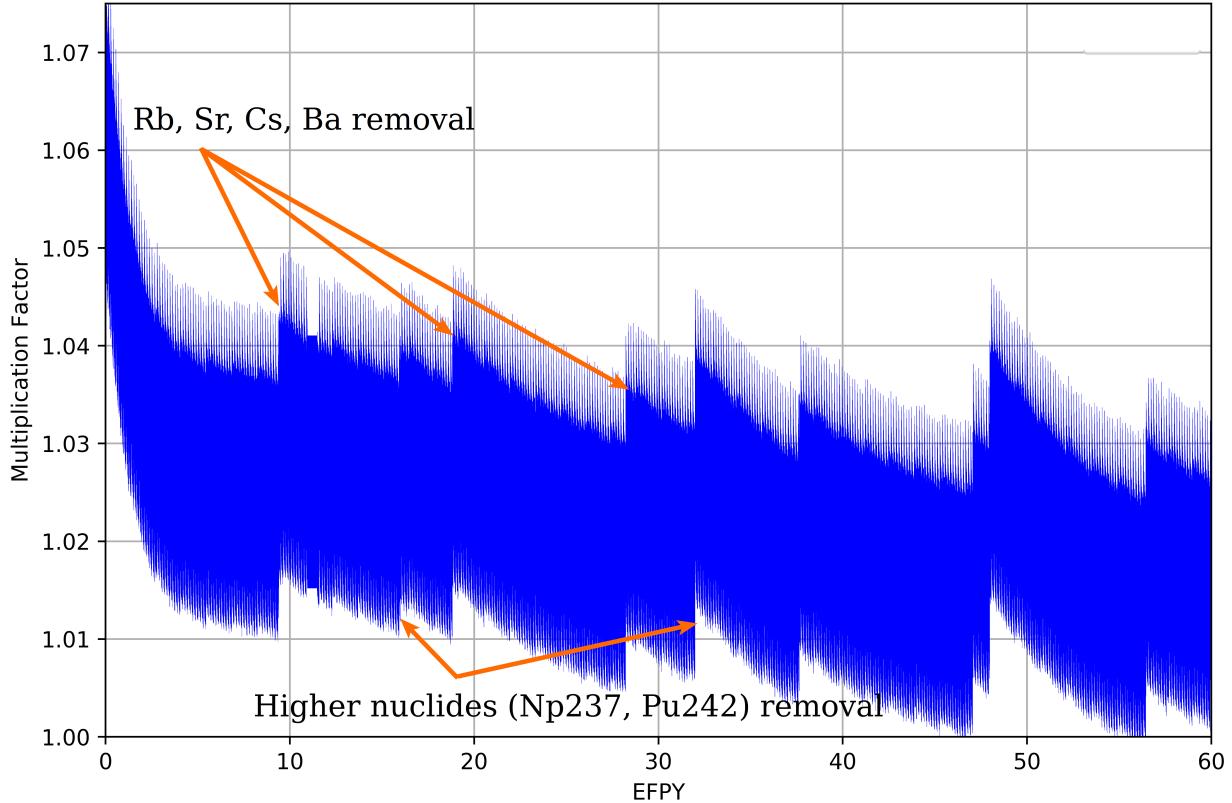


Figure 2.12: Effective multiplication factor dynamics for the full-core MSBR model over a 60-year reactor operation lifetime (reproduced from Rykhlevskii *et al.* [43]).

position at the end of a 3-day depletion. The corresponding effective multiplication factor is much smaller than the previous one. Finally, Serpent calculates k_{eff} for the depleted composition after applying feeds and removals. The k_{eff} increases accordingly since major reactor poisons (e.g. Xe, Kr) are removed, while fresh fissile material (^{233}U) from the protactinium decay tank is added.

Additionally, the presence of rubidium, strontium, cesium, and barium in the core are disadvantageous to reactor physics. Overall, the effective multiplication factor gradually decreases from 1.075 to ≈ 1.02 at equilibrium after approximately 6 years of irradiation.

Loading initial fuel salt composition into the MSBR core leads to a supercritical configuration (Figure 2.14). After reactor startup, the effective multiplication factor for the case with volatile gas and noble metal removal is approximately 7500 pcm higher than for the case with no fission product removal. This significant impact on the reactor core is achieved due

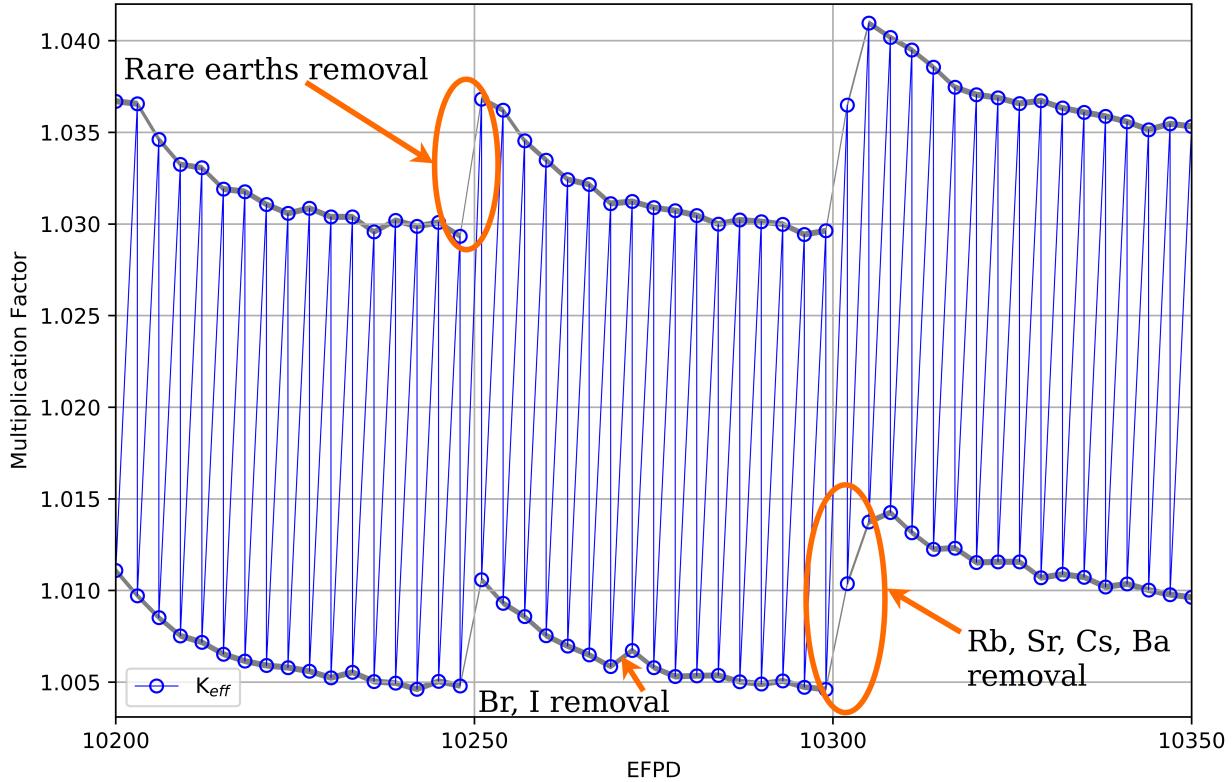


Figure 2.13: Zoomed effective multiplication factor for 150-EFPD time interval (reproduced from Rykhlevskii *et al.* [43]).

to immediate removal (20 sec cycle time) of elements with a high absorption cross section (e.g., Xe, Kr, Mo). The effect of rare earth element removal was considerable a few months after startup and reached approximately 5500 pcm after 10 years of operation. The rare earth elements were removed at a slower rate (50-day cycle time). Moreover, Figure 2.14 demonstrates that batch-wise removal of strong absorbers every 3 days did not necessarily lead to fluctuation in results but rare earth element removal every 50 days caused an approximately 600 pcm jump in reactivity.

The effective multiplication factor of the core reduces gradually over operation time because the fissile material (^{233}U) continuously depletes from the fuel salt due to fission while fission products simultaneously accumulate in the fuel salt. Eventually, without fission product removal, the reactivity decreases to the subcritical state after approximately 500 and 1300 days of operation for cases with no removal and volatile gas & noble metal removal, respectively. The time when the simulated core reaches subcriticality ($k_{eff} < 1.0$) for the

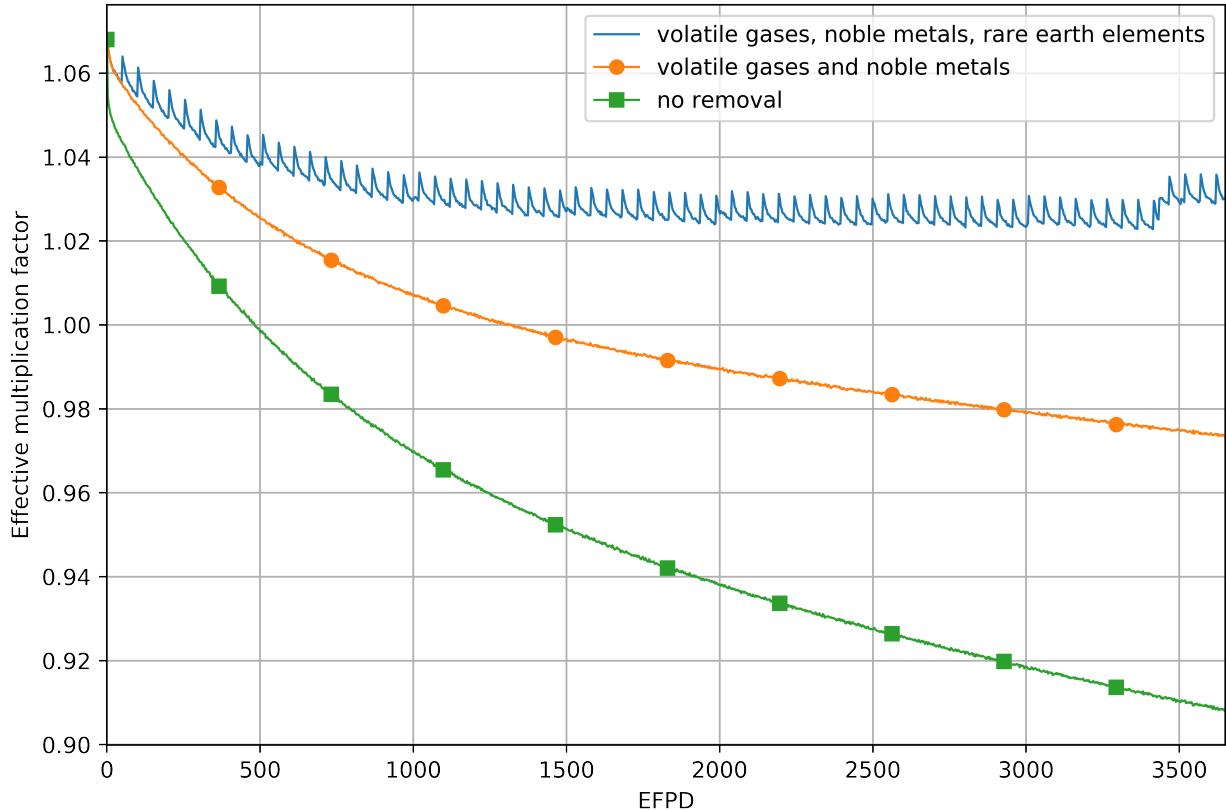


Figure 2.14: Calculated effective multiplication factor for the full-core MSBR model with removal of various fission product groups over 10 years of operation (reproduced from Rykhlevskii *et al.* [43]).

full-core model) is called the core lifetime. Therefore, removing fission products provides significant neutronic benefit and enables a longer core lifetime.

These preliminary results have demonstrated SaltProc's capability to find the equilibrium fuel salt composition (where equilibrium is defined as when the number densities of major isotopes vary by less than 1% over several years). Additionally these results showed the benefits of continuous fission product removal for a thermal MSR design.

CHAPTER 3

CODE DEMONSTRATION AND VALIDATION

After refactoring, redesign, and implementation of the new capabilities mentioned in Chapter 3, SaltProc v1.0 will be tested for the TAP MSR. The TAP concept was selected because it is well analyzed in the literature[49, 23]; thus, cross-code verification with ChemTRITON/SCALE is possible [23]. The demonstration will be performed for two timescales:

Long-term. The reactor lifetime-long (e.g., 40 years) depletion simulation will be performed with moderate time resolution (e.g., 30-day depletion step). The results obtained with SaltProc v1.0 will be compared with recent efforts discussed in Chapter 2, more specifically with Betzler *et al.* [23]. That validation effort will help to ensure that SaltProc v1.0 solution is correct for the case with ideal extraction efficiency.

Short-term (transient). The 7-day-long depletion simulation with changing, load following core power will be performed with the fine time resolution (e.g., 5-min depletion step). The depletion calculation for the TAP in load following regime would capture the effects of xenon poisoning and evaluate the benefit of using an online gas removal system.

Additionally, a compatible *.json* database with input examples of various fuel reprocessing system configurations for use with SaltProc v1.0 will be released to encourage research efforts in online reprocessing simulations for various MSR designs.

3.1 Transatomic Power Molten Salt Reactor concept

The TAP concept is a 1250 MW_{th} MSR with a LiF-based uranium fuel salt [47]. This concept uses configurable zirconium hydride rods as the moderator while most MSR designs

typically propose high-density reactor graphite. Zirconium hydride offers a much higher neutron moderating density than graphite: much less zirconium hydride volume is needed to achieve a thermal energy spectrum similar to one obtained with graphite moderator. Moreover, zirconium hydride has a much longer lifespan in extreme operational conditions (high temperature, large neutron flux, chemically aggressive salt) than reactor graphite. Finally, zirconium hydride is a nonporous material and holds up fewer neutron poisons (e.g., xenon, krypton) than does high-density reactor graphite.

In this section, the design characteristics and reprocessing plant design are based on information presented in the TAP white papers [47, 48] and ORNL technical reports [49, 23].

3.1.1 TAP design description

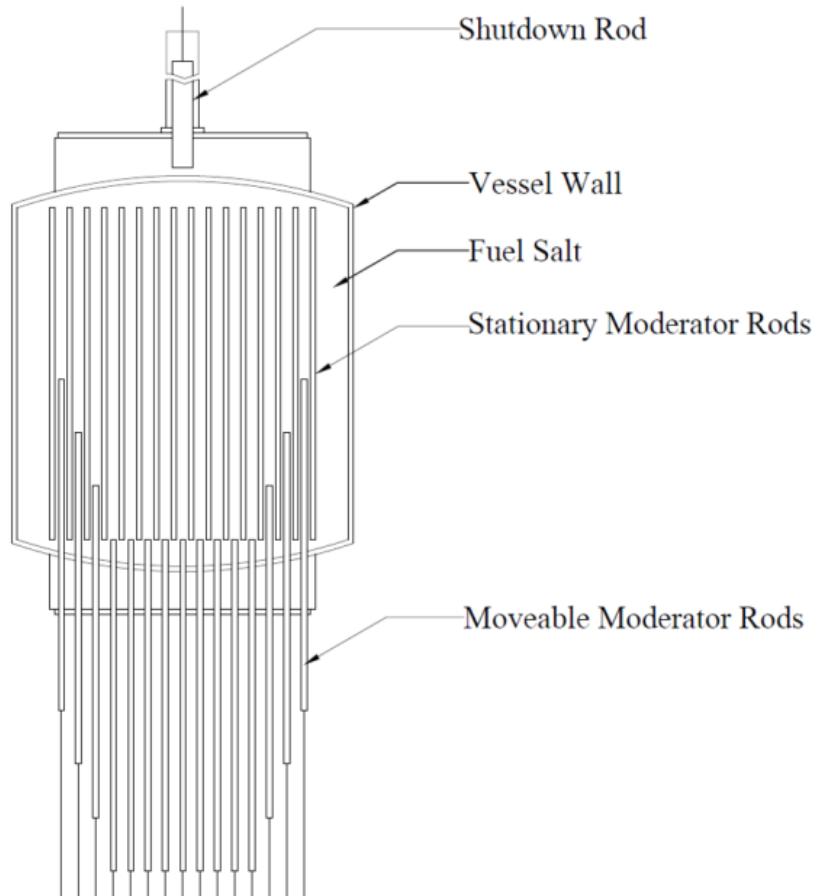


Figure 3.1: The TAP MSR schematic view showing movable moderator rod bundles and shutdown rod (figure reproduced from Transatomic Power White Paper [47]).

The TAP design (figure 3.1) is very similar to original MSRE design developed by ORNL [4] but has two major innovations: the fuel salt composition and the moderator. The MSRE's LiF-BeF₂-ZrF₄-UF₄ salt has been substituted with LiF-UF₄ salt which allows for an increase in the uranium concentration within the fuel salt from 0.9 to 27.5% while maintaining a relatively low melting point (490°C compared with 434°C for the original MSRE's salt) [49]. The graphite has a very high thermal scattering cross section which makes it a perfect moderator but has a few major drawbacks:

- (a) the low lethargy gain per collision requires a large volume of moderator to be present to reach criticality, which leads to a larger core and obstructs the core power density;
- (b) even special reactor-grade graphite has relatively high porosity, consequently, it holds gaseous FPs (e.g., tritium, xenon) in pores;
- (c) the reactor graphite lifespan in a commercial reactor is approximately 10 years [25].

As previously mentioned, to resolve these issues, the TAP concept uses zirconium hydride instead, allowing for a more compact core and a significant increase in power density. These two innovative design choices, together with a configurable moderator (the moderator-to-fuel ratio can be changed during regular maintenance shutdown), facilitate the deployment of this conceptual design in the current commercially available 5% enriched LEU fuel cycle.

The TAP MSR primary loop contains the reactor core volume (including the zirconium hydride moderator rods with silicon carbide cladding), pumps, and primary heat exchanger. Pumps circulate the LiF-(Act)F₄ fuel salt through the primary loop. The pumps, vessels, tanks, and piping are made of a nickel-based alloy (similar to Hastelloy-N¹), which is highly resistant to corrosion in various molten salt environments. Inside the reactor vessel, near the zirconium hydride moderator rods, the fuel salt is in a critical configuration and generates heat. Table 3.1 contains details of the TAP system design which are taken from technical white paper [47] and a neutronics overview [48] as well as ORNL analysis of the TAP design [49, 23].

¹Hastelloy-N is very common in MSR concepts now, having been developed at ORNL in the MSRE program that started in the 1950s.

Table 3.1: Summary of principal data for the TAP MSR (reproduced from [47, 23]).

Thermal power	1250 MW _{th}
Electric power	520 MW _e
Gross thermal efficiency	44%
Outlet temperature	620°C
Fuel salt components	LiF-UF ₄
Fuel salt composition	72.5-27.5 mole%
Uranium enrichment	5% ²³⁵ U
Moderator	Zirconium Hydride (ZrH _{1.66}) rods (with silicon carbide cladding)
Neutron spectrum	thermal/epithermal

3.1.2 TAP core design

In the TAP core (figure 3.2), fuel salt flows around moderator assemblies consisting of lattices of zirconium hydride rods clad in a corrosion-resistant silicone carbide (figure 3.1). The TAP reactor pressure vessel is a cylinder with an inner radius 150 cm, height 350 cm, and wall thickness 5 cm made of a nickel-based alloy.

The salt volume fraction in the core is parameter similar to wide-used moderator-to-fuel ratio and can be defined as:

$$SVF = \frac{V_F}{V_F + V_M} = \frac{1}{1 + V_M/V_F} \quad (3.1)$$

where

V_F = the fuel volume

V_M = the moderator volume

V_M/V_F = the moderator-to-fuel salt ratio.

The salt volume fraction in the core can be varied during operation to shift the spectrum from intermediate to thermal energies (from BOL to End of Life (EOL), respectively) to maximize fuel burnup. In practice, salt volume fraction can be varied by inserting fixed-sized moderator rods via the bottom of the reactor vessel (for safety considerations), similarly to moving the control rods in a Boiling Water Reactor (BWR), as shown in Figure 3.1. For

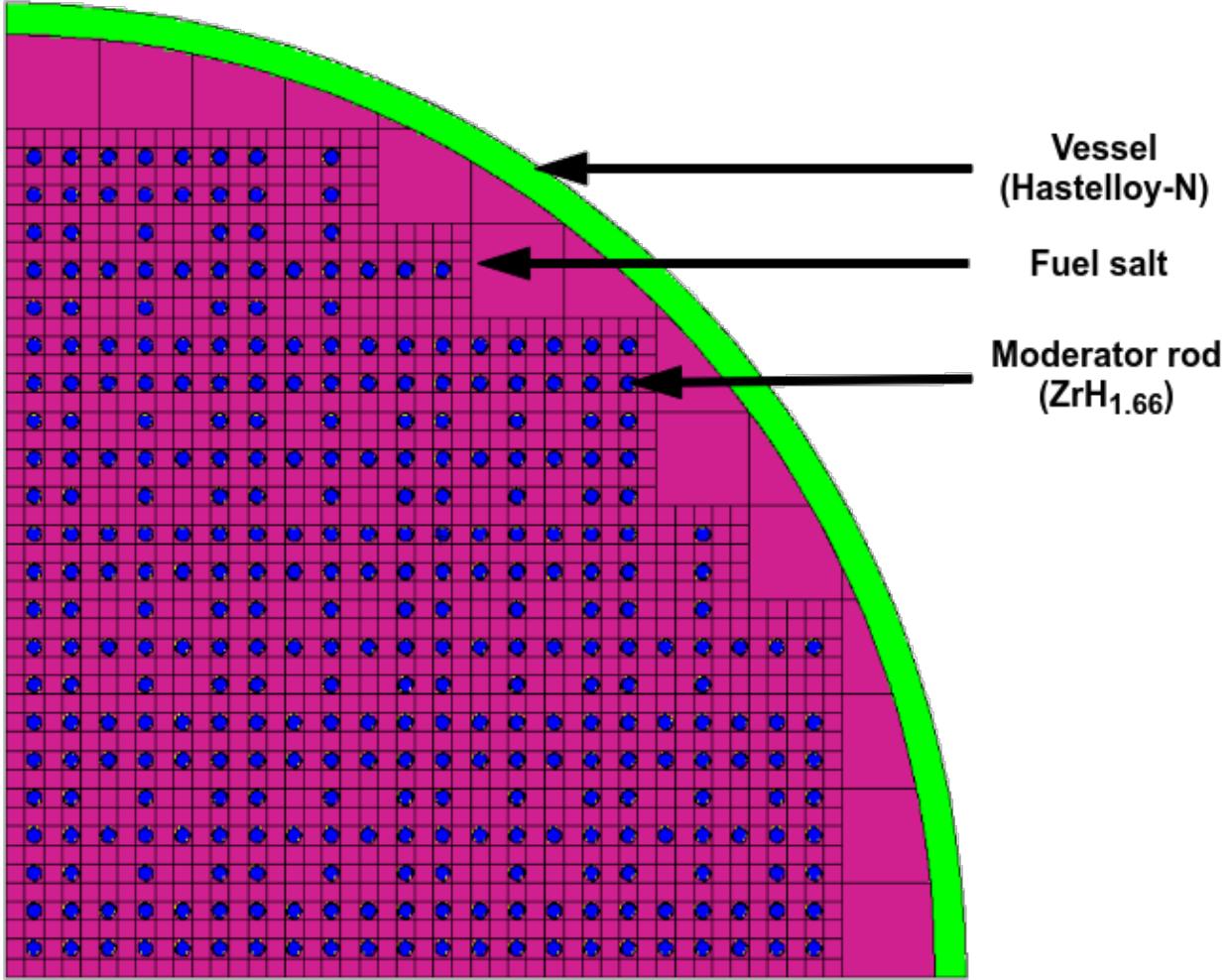


Figure 3.2: The TAP MSR schematic core view showing moderator rods (figure reproduced from ORNL/TM-2017/475 [23]).

the TAP reactor, EOL occurs when the maximum number of moderator rods are inserted into the core, and a further injection of fresh fuel salt does not alter criticality. Unmoderated salt is flowing in the annulus between the core, and the vessel wall provides for a potential reduction in fast neutron flux at the vessel structural material [48].

3.1.3 Serpent 2 full-core model

Nest and lattice geometry types as well as universe transformation capabilities of Serpent [50] are employed to represent TAP core. Figure 3.3 shows the *XY* section of whole-core configuration at the expected reactor operational level when all control rods are fully withdrawn. Figures 3.4 and 3.5 show a longitudinal section of the reactor. This model contains

the moderator rods with silicon carbide cladding, pressure vessel, and inlet and outlet plena (Table 3.2). Fuel salt flows around rectangular moderator assemblies consisting of lattices of small-diameter zirconium hydride rods in a corrosion-resistant material. The salt volume fraction for model herein is 0.907268, which means the modeled core is under-moderated and has an intermediate, epithermal spectrum.

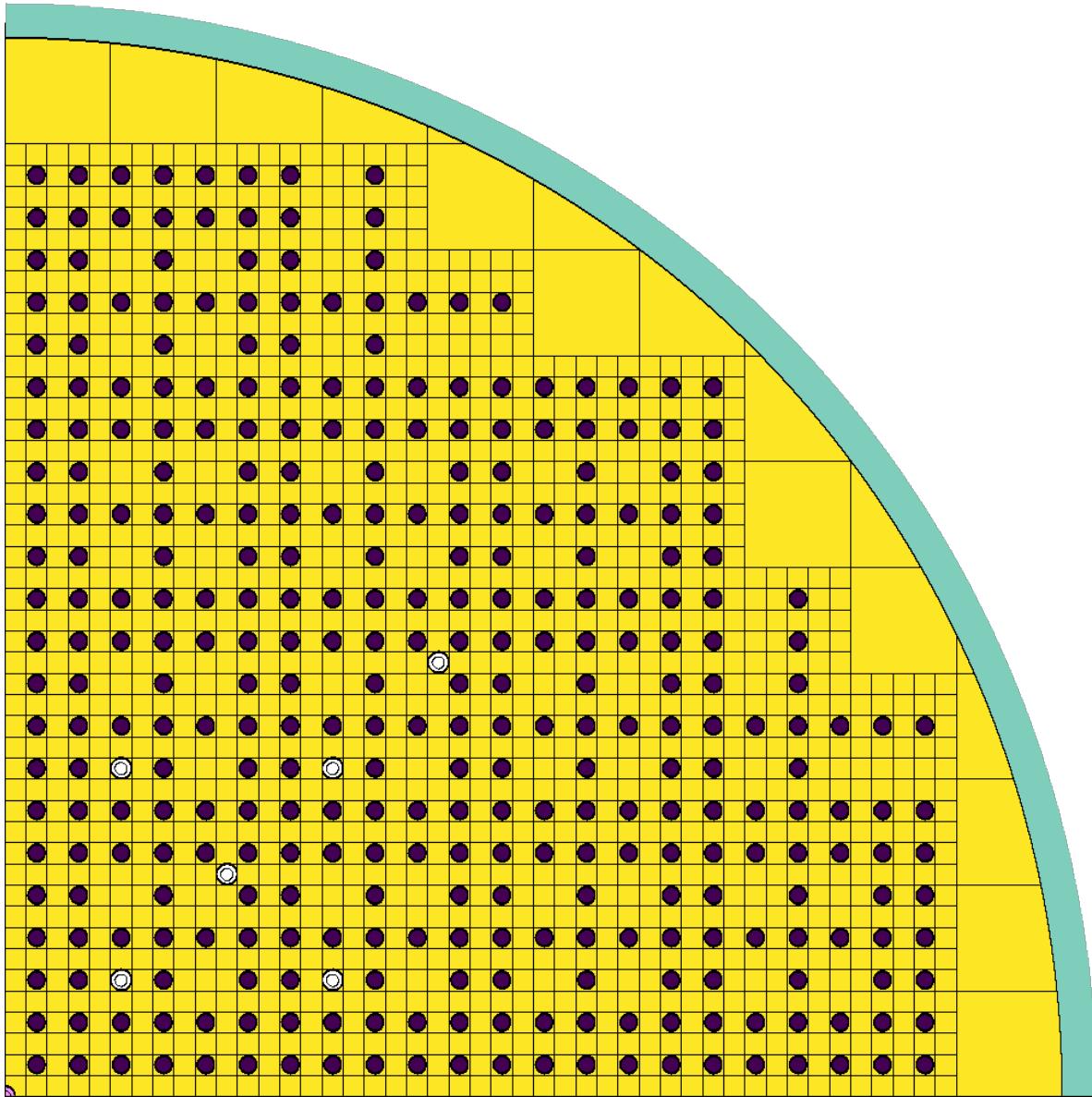


Figure 3.3: An XY section of the TAP model at horizontal midplane with fully withdrawn control rods at BOL (salt volume fraction is equal 0.907268). The violet color represents zirconium hydride, and the yellow represents fuel salt. The blue color shows Hastelloy-N, a material used for the vessel wall, and the white color is the air.

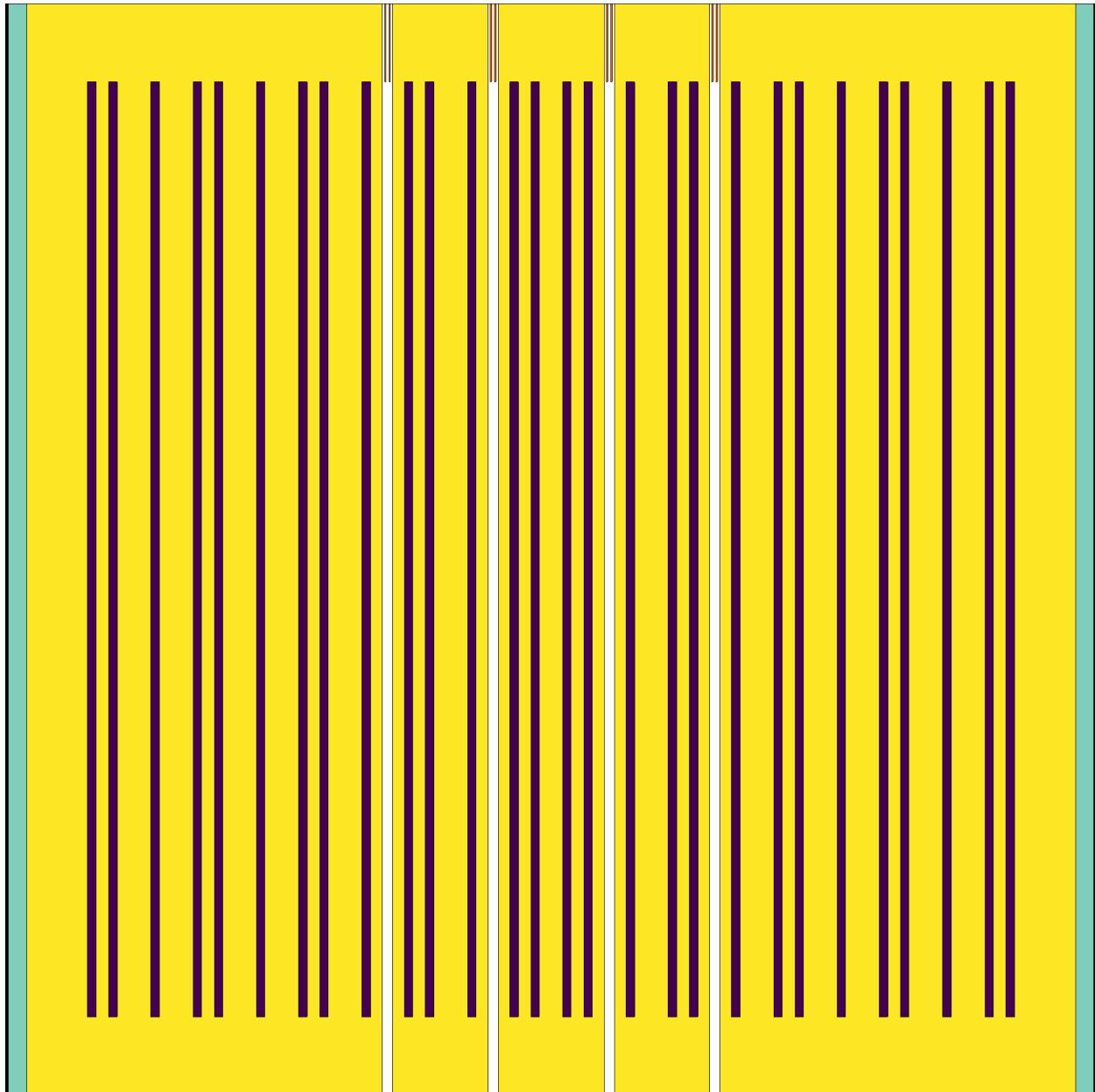


Figure 3.4: An XZ section of the TAP model.

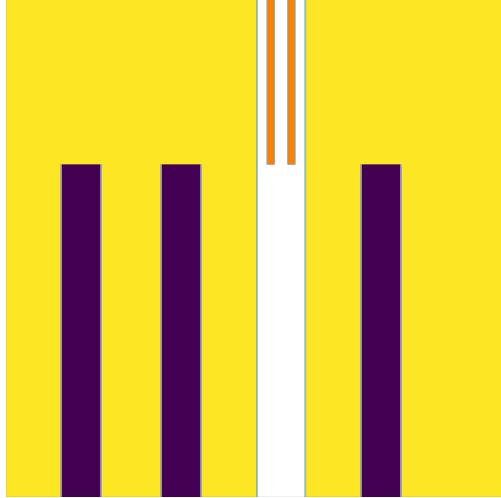


Figure 3.5: Zoomed XZ section of the top of the moderator rods and guide tubes in the TAP model. The orange color shows 70-30% Gd_2O_3 - Al_2O_3 ceramic absorbers used for control rods.

To represent the reactivity control system, the model has:

- (a) control rod guide tubes made of nickel-based alloy;
- (b) control rods represented as hollow 70-30% Gd_2O_3 - Al_2O_3 cylinders with a thin Hastelloy-N coating [23];
- (c) air inside guide tubes and control rods;

The control rod design is comprised of a cluster of 25 rods that provide a total reactivity worth of $1110 \pm 9.7 \text{pcm}$.

The control rod cluster is modeled using the **TRANS** Serpent 2 feature, which allows the user to easily change the control rod position during the simulation. Herein I assumed that all control rods are fully withdrawn from the core (figure 3.5), but control rod positions may vary in future investigations. In this report, all figures of the core were generated using the built-in Serpent plotter.

3.2 TAP fuel salt reprocessing system

The TAP nuclear island contains the FP removal system. Gaseous FPs are continuously removed using an off-gas system while liquid and solid FPs are extracted via a chemical processing system. As these byproducts are gradually removed, a small quantity of fresh fuel salt is regularly added to the primary loop. This process conserves a constant fuel salt mass and keeps the reactor critical. In contrast with the MSBR reprocessing system, the TAP design does not need a protactinium separation and isolation system because it operates in a uranium-based single-stage fuel cycle. The authors of the TAP concept suggested three distinct fission product removal methods [48]:

Off-Gas System: The off-gas system removes gaseous fission products such as krypton and xenon, which are then compressed and stored temporarily until they have decayed to the background radiation level. Trace amounts of tritium are also removed and bottled in a liquid form via the same process. Also, the off-gas system directly removes a small fraction of the noble metals.

Metal Plate-Out/Filtration: A nickel mesh filter removes noble and semi-noble metal solid fission products as they plate out onto internal surface of the filter.

Table 3.2: Geometric parameters for the full-core 3D model of the TAP (reproduced from Betzler *et al.* [23]).

Component	Parameter	Value	Unit
Moderator rod	Cladding thickness	0.10	cm
	Radius	1.15	cm
	Length	3.0	m
	Pitch	3.0	cm
Moderator assembly	Array	5×5	rods×rods
	Pitch	15.0	cm
Core	Assemblies	268	assemblies/core
	Inner radius	1.5	m
	Plenum height	25.0	cm
	Vessel wall thickness	5.0	cm

Liquid Metal Extraction: Lanthanides and other non-noble metals stay dissolved in the fuel salt. They generally have a lower capture cross section and thus absorb fewer neutrons than ^{135}Xe , but their extraction is essential to ensure normal operation. In the TAP reactor, lanthanide removal is accomplished via a liquid-metal/molten salt extraction process similar to that developed for MSBR by ORNL [25]. The process converts the dissolved lanthanides into a well-understood oxide waste form, similar to that of LWR SNF. This oxide waste comes out of the TAP reprocessing plant in ceramic granules and can be sintered into another convenient form for storage [47].

Figure 3.6 shows the principal design of the TAP primary loop, including an off-gas system, nickel mesh filter, and lanthanide chemical extraction facility. Similarly to the MSBR, an off-gas system is also based on a simple process of helium sparging through fuel salt with consequent gas bubbles removed before returning the fuel salt to the core. Nevertheless, one crucial difference must be noted: the MSBR gas separation system suggested helium injection and subsequent transport of the voids throughout the primary loop, including the core for at least ten full loops [25]. It is a significant concern for safe, stable operation because the increase of void fraction in the fuel salt when it enters back to the core would cause unpredictable reactivity change. This drawback can be overcome by using an effective gas separator for stripping helium/xenon bubbles before returning the salt to a primary loop (Figure 3.6, blue block).

Noble and semi-noble metal solid fission products tend to plate out onto metal surfaces including piping, heat exchanger tubes, reactor vessel inner surface, etc. Previous research by ORNL [25] reported that about 50% of noble and semi-noble metals would plate out inside MSBR systems without any special treatment. To improve the extraction efficiency of these fission products, the TAP concept suggested employing a nickel mesh filter located in a bypass stream in the primary loop (Figure 3.6, orange block). The main idea of this filter is to create a maze with large metal (nickel) surface area. The fuel salt is flowing throughout the filter and noble metals plate-out on the internal filter surface.

This Liquid Metal Extraction process for the TAP concept has been adopted from the MSBR. The MSRE demonstrated a liquid-liquid extraction process for removing rare earths and lanthanides from fuel salt and estimated efficiency of this process. Removal efficiency

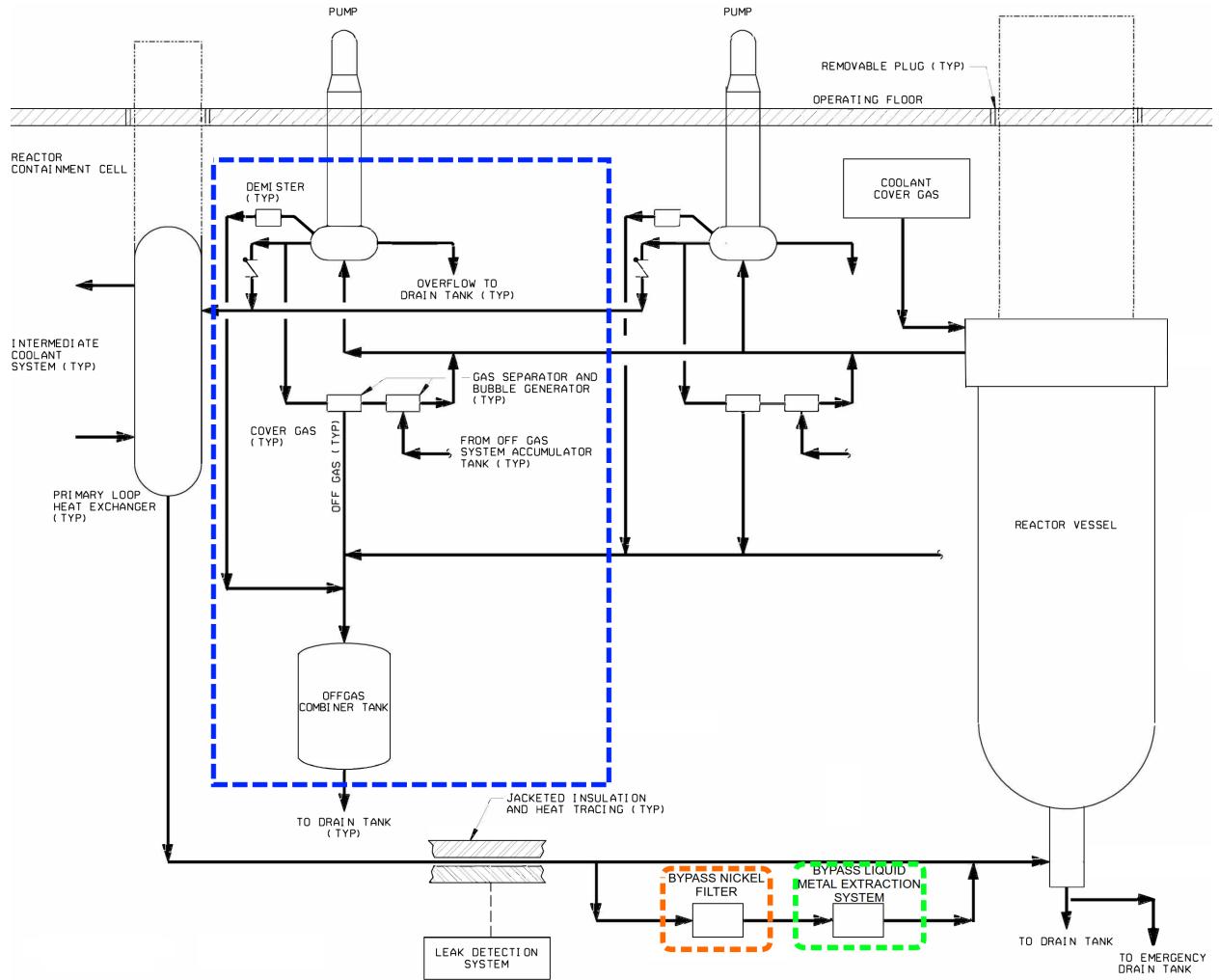


Figure 3.6: Simplified TAP primary loop design including off-gas system (blue), nickel filter (orange) and liquid metal extraction system (green) (reproduced from [51]).

(ϵ_{RE}) of this process is the function of salt mass flow rate, liquid bismuth mass flow rate, interfacial areas between salt and metal, and mass transfer coefficient for each noble metal species. The most recent research for LiF salt (for the MSFR concept) reported the following form of extraction efficiency correlation [52]:

$$\begin{aligned}\epsilon_{RE} &= \frac{1}{1 + 10^\lambda} \\ &= \frac{1}{1 + 10^{f(A, \dot{m}_{Bi}, \dot{m}_{salt}, N, K)}}\end{aligned}\quad (3.2)$$

where

A = metal-to-salt interface area

\dot{m}_{Bi} = bismuth mass flow rate

\dot{m}_{salt} = salt mass flow rate

N = number of stages

K = liquid phase mass transfer coefficient

Correlations are different for various lanthanides and can be determined from experimental data and/or existing analytical models [53, 54, 52].

In fact, due to similarities in reprocessing schemes, the TAP project reported almost the same set of elements for removal and similar effective cycle times as suggested for MSBR (Table 3.3). The TAP neutronics whitepaper specifies additional low-probability fission products and gases that should be removed during operation. These elements are categorized into the previously defined processing groups, but the removal rates of most of these elements (i.e., all except for hydrogen) are very low.

Details of gas removal and fuel reprocessing systems have historically been conceptual. Accordingly, liquid-fueled system designs including the TAP concept usually assume ideal (rather than realistically constrained) removal efficiencies for reactor performance simulations. For the proposed work, a realistic online reprocessing system and reactor model will be created to capture the dynamics of fuel composition evolution during reactor operation.

Gas removal efficiency will be represented in that model as a variable, described using mathematical correlation from Chapter 2 (see Equation 2.1). For the other FPs, a fixed², non-ideal extraction efficiency based on cycle time from Table 3.3 will be used in the fuel reprocessing model.

Table 3.3: The effective cycle times for fission products removal from the TAP reactor (reproduced from [55] and [48]).

Processing group	Nuclides	Removal Rate (s^{-1})	Cycle time (at full power)
<i>Elements removed in MSBR concept and adopted for the TAP [25]</i>			
Volatile gases	Xe, Kr	5.00E-2	20 sec
Noble metals	Se, Nb, Mo, Tc, Ru, Rh, Pd, Ag, Sb, Te	5.00E-2	20 sec
Seminoble metals	Zr, Cd, In, Sn	5.79E-8	200 days
Volatile fluorides	Br, I	1.93E-7	60 days
Rare earths	Y, La, Ce, Pr, Nd, Pm, Sm, Gd Eu	2.31E-7 2.32E-8	50 days 500 days
Discard	Rb, Sr, Cs, Ba	3.37E-9	3435 days
<i>Additional elements removed [48, 55]</i>			
Volatile gases	H	5.00E-2	20 sec
Noble metals	Ti, V, Cr, Cu	3.37E-9	3435 days
Seminoble metals	Mn, Fe, Co, Ni, Zn, Ga, Ge, As	3.37E-9	3435 days
Rare earths	Sc	3.37E-9	3435 days
Discard	Ca	3.37E-9	3435 days

3.3 Preliminary results

An extended version of the SaltProc online reprocessing simulation package was demonstrated for the TAP MSR with static core geometry and 5% LEU in startup fuel salt composition. Three fueling scenarios were considered:

- (a) no FP removal or feed (Serpent only);

²Published information about dynamics of extraction efficiency during reactor operation for seminoble metals, volatile fluorides, and rare earths is insufficient to inform a variable removal efficiency.

- (b) a 5% LEU online feed;
- (c) a 19.79% LEU online feed.

The neutron population per cycle and the number of active/inactive cycles were chosen to obtain a balance between minimizing uncertainty for a transport problem (28 pcm for effective multiplication factor) and simultaneously minimizing computational time.

3.3.1 TAP long-term demonstration case

The simulations already conducted in this work thoroughly analyzed the original TAP reprocessing system design (figure 3.6) and neutron poisons removal rates (table 3.3) to determine a suitable reprocessing scheme for SaltProc v1.0 demonstration (figure 3.7). That demonstration case assumed fixed, non-ideal (< 100%) removal efficiencies and static geometry with constant salt volume fraction during 13 years of operation.

The gas removal components (the sparger and entrainment separator) are located in-line because estimated full loop time for the fuel salt is about 18 seconds and approximately equal to the cycle time (table 3.3). To remove all volatile gases every 20 seconds, the gas removal system must operate with 100% of the core throughout flow rate (in-line gas removal system). For the demonstration case herein to achieve required cycle time, the simulations herein assumed that xenon, krypton, and hydrogen extraction efficiencies for the sparger and entrainment separator are equal to 60% and 97%, respectively.

The nickel filter in the TAP concept is designed to extract noble metals and volatile fluorides. Similarly to volatile gases, noble metals must be removed every 20 seconds and, hence, the filter should operate in-line also. The nickel filter removes a wide range of elements with various efficiencies (table 3.3).

Lanthanides and other non-noble metals generally have a lower capture cross section and absorb fewer neutrons than gases and noble metals. These elements can be removed via a liquid-metal/molten salt extraction process with relatively low removal rates (cycle time > 50 days). This is accomplished by directing a small fraction of the salt mass flow leaving the nickel mesh filter (10%). That fraction then flows to the liquid-metal/molten salt com-

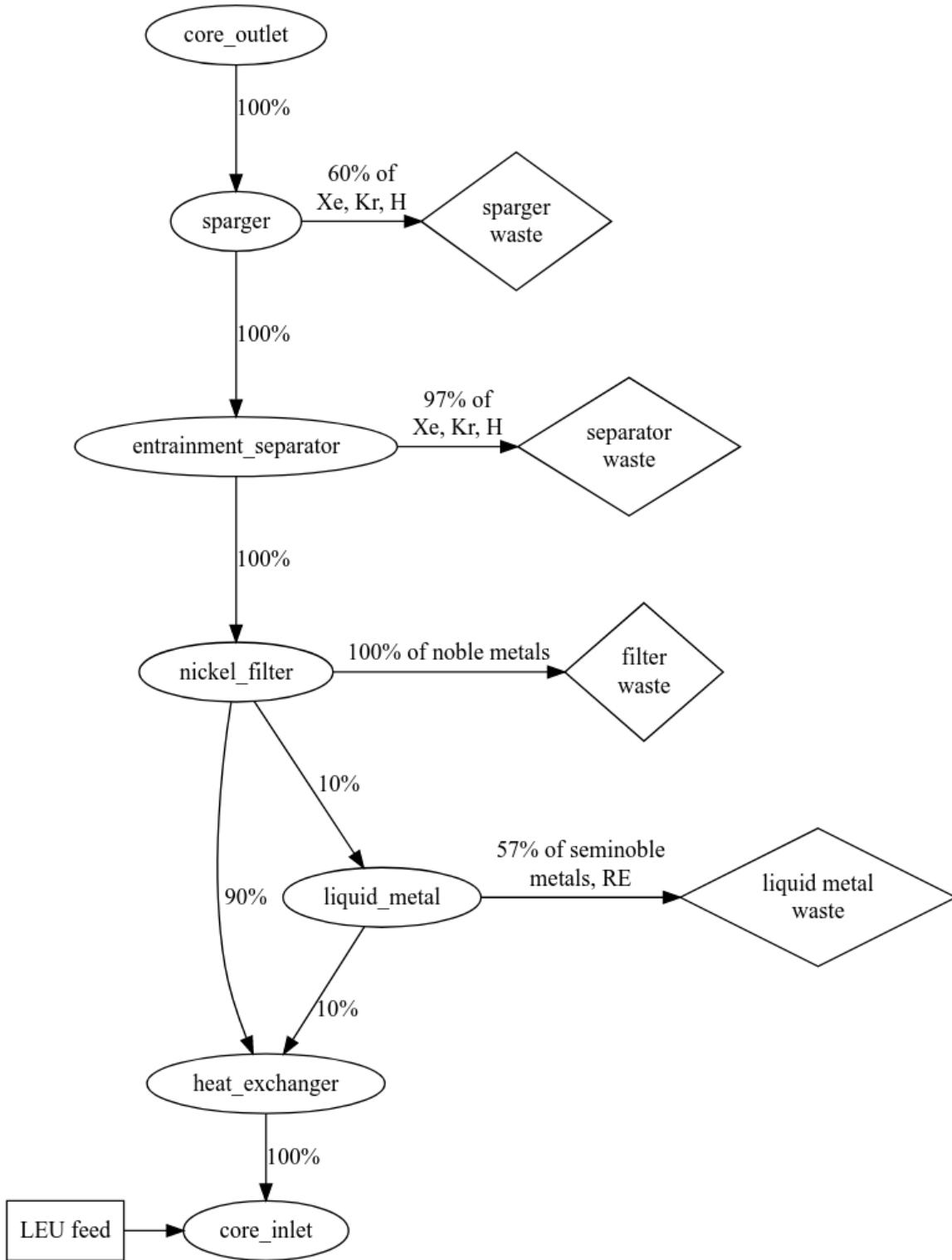


Figure 3.7: TAP reprocessing scheme flowchart used for demonstration of extended SaltProc. Arrows represent material flow; percents represent fraction of total mass flow rates; ellipses represent fuel reprocessing system components; diamonds represent waste streams; the box shows refuel material flow.

ponent of the reprocessing system, where lanthanides are removed with a specific extraction efficiency to match the required cycle time (table 3.3). The rest 90% of the salt mass flow is directed from the nickel filter to heat exchanger without performing any fuel salt treatment.

The removal rates vary among nuclides in this reactor concept, which dictate the necessary resolution of depletion calculations. To compromise, a 3-day time was selected based on a timestep refinement study by Betzler *et al.* [23].

3.3.2 Effective multiplication factor

Figures 3.8, 3.9, and 3.10 demonstrate the effective multiplication factors obtained using SaltProc v1.0 and Serpent. SaltProc obtained the effective multiplication factors after removing fission products and adding feed material at the end of each depletion step (3 days for this work). The k_{eff} fluctuates significantly as a result of the batch-wise nature of this online reprocessing strategy.

Loading initial fuel salt composition with 5% LEU into the TAP core leads to a supercritical configuration with an excess of reactivity about 1900pcm (figure 3.8). Without performing any fuel salt reprocessing the core became subcritical after 30 days of operation (figure 3.9). I obtained this result using Serpent on its own, without introducing any FP extraction and refueling. For the beginning of the TAP reactor lifetime, uranium enrichment in the feed has a minor effect because a tiny amount of poisons was produced (<1kg/day) and, hence, a small mass of fresh salt was injected. Notably, the core went subcritical after 42 days of operation either with LEU 5% or LEU 19.79% feed.

The TAP core is never reached equilibrium fuel salt composition without performing fuel salt reprocessing and refueling. For the fueling scenarios with 5% and 19.79% LEU feed, the reactor achieved the equilibrium state after 10 years of operation. Overall, the effective multiplication factor gradually decreases from 1.018 to 0.88 for the 19.79% LEU feed and 0.86 for the 5% LEU feed, which indicates problems with operating this nuclear reactor design. The proposed continuation of this work will to overcome this issue by adding dynamic salt volume fraction functionality to SaltProc v1.0.

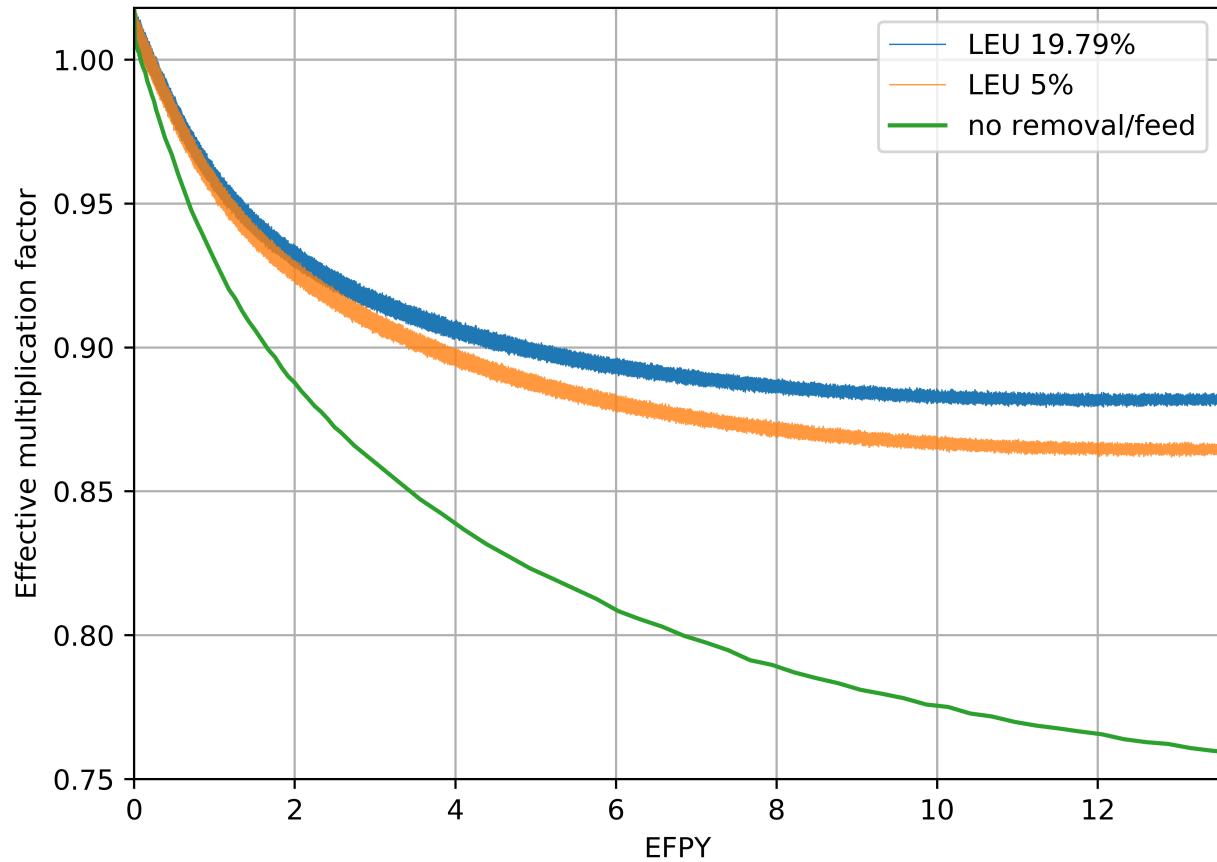


Figure 3.8: Effective multiplication factor dynamics for full-core TAP model for different fueling scenarios over a 13-year reactor operation. Confidence interval $\pm\sigma = 28\text{pcm}$ is shaded.

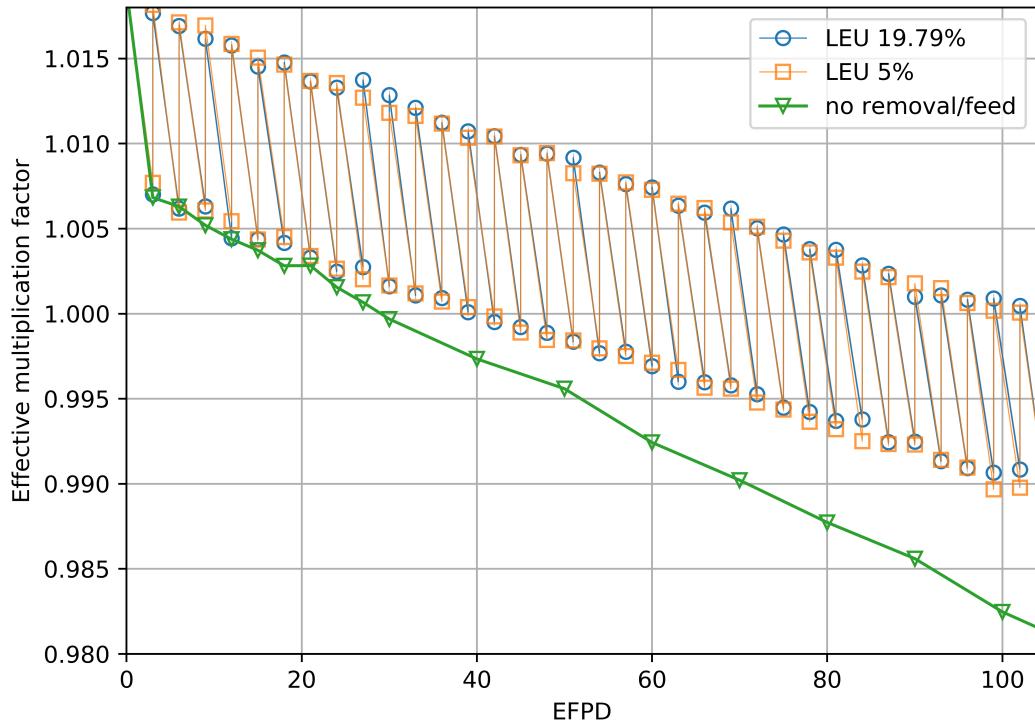


Figure 3.9: Zoomed effective multiplication factor for the first 104 EFPD after startup.

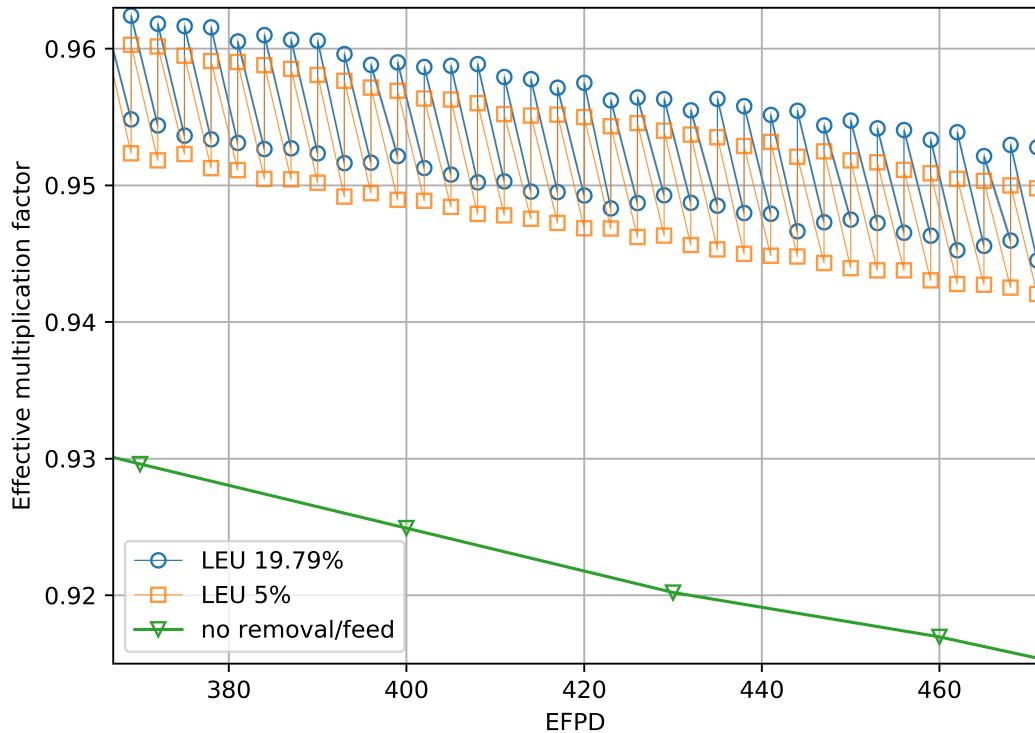


Figure 3.10: Zoomed effective multiplication factor for the time interval from 367 to 471 EFPD after startup.

3.3.3 Neutron spectrum

Figure 3.11 shows the normalized neutron flux spectrum for the full-core TAP core model in the energy range from 10^{-8} to 15 MeV. The neutron energy spectrum at equilibrium is slightly harder than at startup due to plutonium and other strong absorbers accumulating in the core during reactor operation. The TAP reactor spectrum is significantly harder than in a typical LWR and is in a good agreement with ORNL report [23].

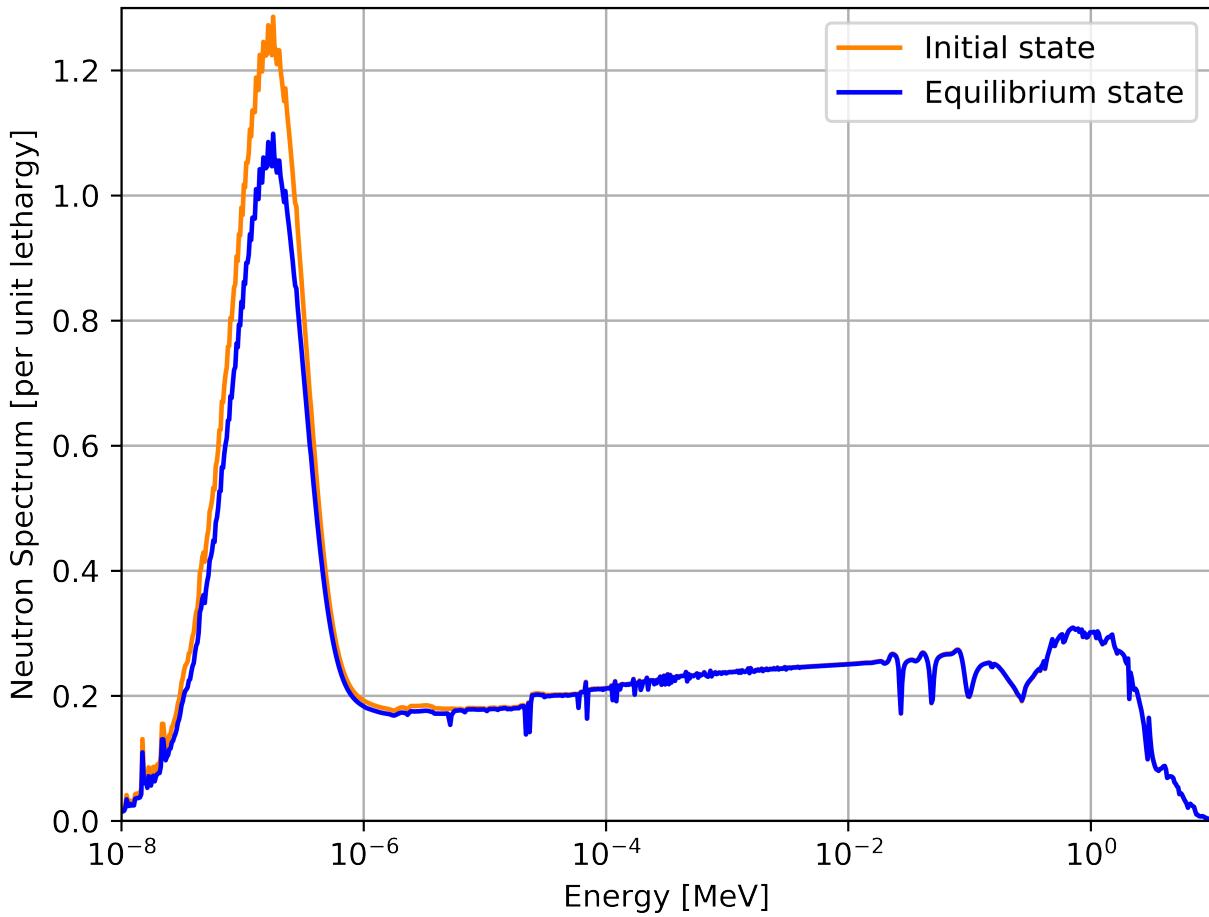


Figure 3.11: The neutron flux energy spectrum normalized by unit lethargy for initial and equilibrium fuel salt composition.

3.3.4 Fuel salt composition

Figure 3.12 shows the absolute mass of major heavy isotopes which have a strong influence on the reactor core physics. The masses of ^{236}U , ^{238}U , ^{239}Pu , ^{240}Pu , and ^{241}Pu in the fuel salt change insignificantly after approximately 10 years of operation, which matches the stabilization time for the effective multiplication factor. Hence, the quasi-equilibrium state was reached after 10 years of reactor operation. Moreover, the TAP core bred approximately the same amount of fissile ^{239}Pu ($\approx 2\text{t}$) as corresponds to the initial fissile material (^{235}U) load. A significant amount of non-fissile plutonium builds up during operation and accounts for 50% of the plutonium after 13 years of operation. Overall, the rate of breeding fissile ^{239}Pu from ^{238}U even in this relatively hard neutron spectrum is not large enough to compensate for the negative effects of strong absorber accumulation or to keep the reactor critical.

I checked correctness of SaltProc v1.0 by comparing the masses of important isotopes for load-following operation (^{135}Xe , ^{135}I) to the expected masses after each depletion step (figure 3.13). For ^{135}Xe , the expected mass was calculated as follows:

$$m_{post} = m_{pre} \times (1 - \epsilon_s) \times (1 - \epsilon_{es}) \quad (3.3)$$

where

m_{post} = mass of the isotope after applying removals and feeds

m_{pre} = mass of the isotope right before reprocessing

ϵ_s = sparger extraction efficiency

ϵ_{es} = entrainment separator extraction efficiency.

For iodine, the approach is similar, but the extraction efficiency of iodine in the nickel filter is only 5%. Figure 3.13 shows that SaltProc v1.0 extraction module correctly removes target isotopes with specified extraction efficiency: SaltProc calculations match the expected mass.

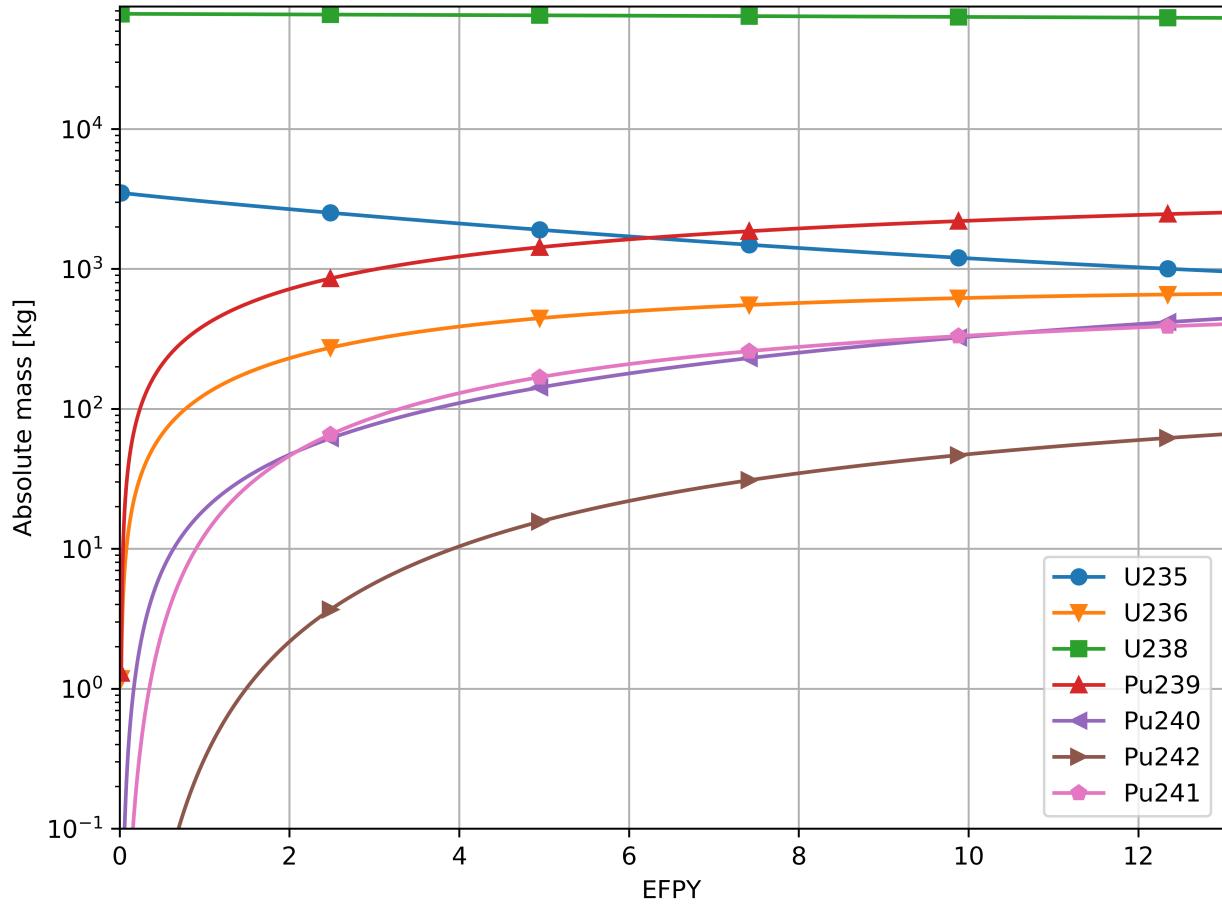


Figure 3.12: Mass of major nuclides during 13 years of reactor operation with 19.79% LEU feed.

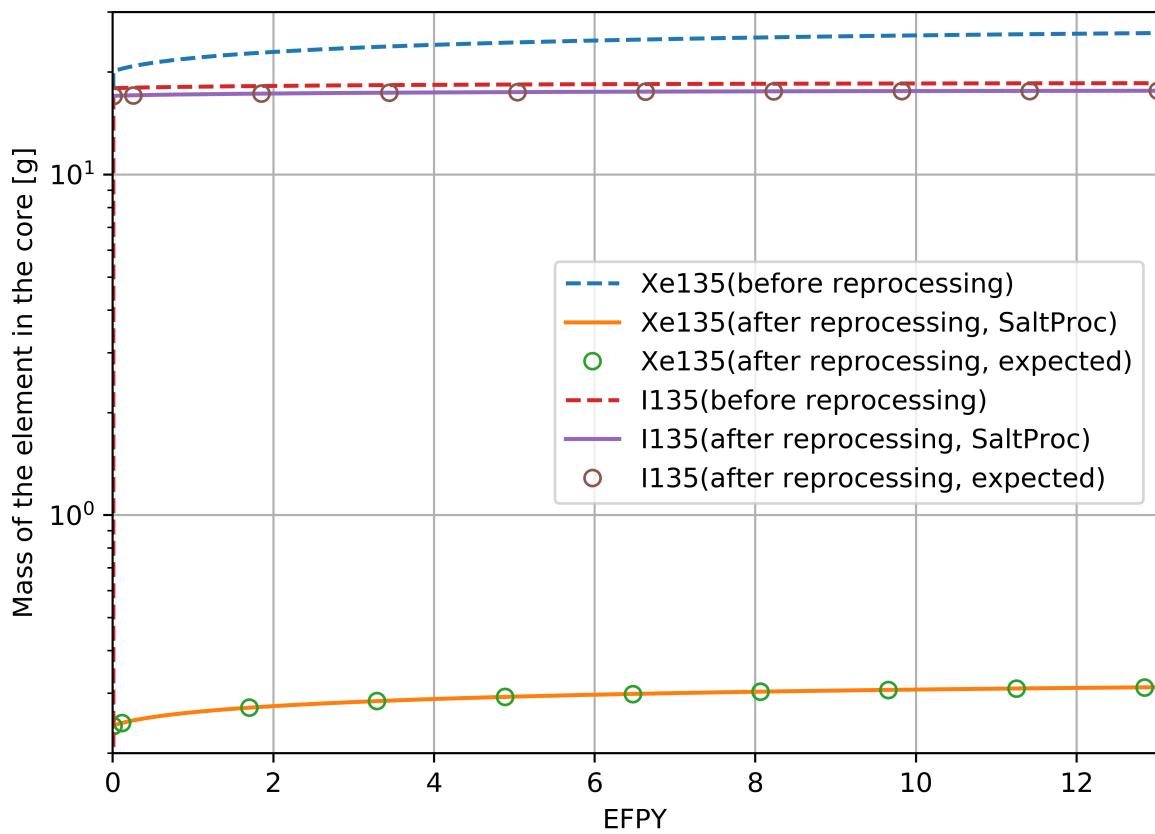


Figure 3.13: Mass of major neutron poison, ^{135}Xe , and its main precursor, ^{135}I , during 13 years of reactor operation before and after performing reprocessing in SaltProc v1.0.

CHAPTER 4

SAFETY ANALYSIS

4.1 Safety and operational parameters

The proposed work will investigate the following safety and operational parameters: the temperature reactivity feedback coefficient, reactivity worth of the control rods, and power axial offset. Dynamics of these parameters will be investigated for the TAP concept during lifetime-long operation and short-term load-following transient. Validation against previous work mentioned in section 1.2.1 will be performed for confidence building.

4.1.1 Temperature coefficient of reactivity

The main physical principle underlying the reactor temperature feedback is an expansion of heated material. When the fuel salt temperature increases, the density of the salt decreases, but at the same time, the total volume of fuel salt in the core remains constant because it is bounded by the vessel. When the moderator rod temperatures increase, the density of zirconium hydride decreases, creating additional space for fuel salt.

Chapter 2, equation 1.3 defined Total or Isothermal Temperature Coefficient (ITC) which expresses the dependence of the core reactivity on the combined effects of fuel and moderator temperature. However, fuel and moderator temperature are rarely equal because fuel heats up much faster than the moderator; thus, the fuel temperature coefficient (FTC) and the moderator temperature coefficient (MTC) should be calculated separately. In the base case simulation in this work, the fuel salt and the moderator temperatures are fixed at 900K, which is an operational temperature in the core. To determine the fuel salt temperature coefficient (FTC), I will perturb the fuel salt temperature to 800K and 1000K while fixing the

moderator temperature at 900K (base case). Likewise, the moderator temperature coefficient (MTC) will be found by perturbing the moderator temperature from 900K to 800K and 1000K, while the fuel temperature is fixed at 900K.

The range of temperature perturbation for the temperature coefficient calculation has been selected based on operational parameters. The TAP MSR operates between 773 and 973K (500-700°C), which is far below the salt boiling point of approximately 1473K. The salt freezes if the temperature drops below 773K. At the other end of the temperature spectrum, the temperature higher than 973K passively melts a freeze plug, which drains the fuel salt from the reactor vessel to the drain tanks. The drain tanks have a subcritical configuration with a large free surface area to readily dissipate heat by passive cooling [47]. Thus, calculating temperature coefficients in the temperature range from 800 to 1000K will capture the outcomes of most accident transients.

Fuel and Moderator Temperature Coefficients for the temperature in the range from 800 to 900K would be useful for analyzing the following transients related to sudden fuel salt cooling:

- increase in heat removal by the secondary system;
- increase in the fuel salt flow rate;
- planned reactor shutdown.

Temperature coefficients for the temperature in the range from 900 to 1000K would be used for transients related with salt overheating:

- loss-of-coolant accident (LOCA);
- loss-of-flow accident (LOFA);
- loss of ultimate heat sink;
- station blackout (SBO).

Thus, temperature coefficients will be calculated separately for above 900K and below 900K.

My dissertation will include the following cases:

1. FTC (moderator rods temperature is fixed at 900K):
 - (a) temperature of the salt between 900 and 1000K;
 - (b) temperature of the salt between 900 and 800K.
2. MTC (the fuel salt temperature is fixed at 900K):
 - (a) temperature of the moderator between 900 and 1000K;
 - (b) temperature of the fuel between 900 and 800K.
3. ITC:
 - (a) whole reactor temperature between 900 and 1000K;
 - (b) whole reactor temperature between 900 and 800K;

In the first case, changes in the fuel temperature will impact cross section temperature (Doppler broadening) and fuel density, but the geometry is unchanged because the fuel is a liquid. The density of fuel salt changes with respect to temperature as follows [56]:

$$\rho_{salt}(T[K]) = 6.105 - 12.720 \times 10^4 T \quad [g/cm^3] \quad (4.1)$$

The uncertainty in the salt density calculated using Equation 4.1 is approximately 0.036 g/cm^3 at 900K. In contrast, when the moderator temperature is changing, the density, cross section temperature, and the geometry are also changing due to thermal expansion of the solid zirconium hydride ($ZrH_{1.66}$) rods. Accordingly, the new moderator density and sizes will be calculated using a linear temperature expansion coefficient of $2.734 \times 10^{-5} K^{-1}$ [57]. A new geometry input for Serpent 2, which takes into account displacement of the moderator surfaces, will be created based on this information. Finally, the temperature coefficient for each case will be calculated separately as follows:

$$\alpha = \frac{k_{eff}(T_{i+1}) - k_{eff}(T_i)}{k_{eff}(T_{i+1})k_{eff}(T_i)(T_{i+1} - T_i)} \quad (4.2)$$

where

k_{eff} = effective multiplication factor

T_i = fuel salt temperature in (800K, 1000K).

By propagating the k_{eff} statistical error provided by Serpent 2, uncertainty for each temperature coefficient will be obtained using the formula:

$$\delta\alpha = \left| \frac{1}{T_{i+1} - T_i} \right| \sqrt{\frac{\delta k_{eff}^2(T_{i+1})}{k_{eff}^4(T_{i+1})} + \frac{\delta k_{eff}^2(T_i)}{k_{eff}^4(T_i)}} \quad (4.3)$$

where

δk_{eff} = statistical error for k_{eff} from Serpent output.

Notably, other sources of uncertainty are neglected, such as cross section measurement error and approximations inherent in the equations of the state providing both the salt and moderator density dependence on temperature.

4.1.2 Reactivity control system rod worth

In the TAP concept, control rods perform two main functions: to shut down the reactor at any point during operation by introducing sufficient negative reactivity and to control the excess of reactivity after moderator rod reconfiguration during regular maintenance. In an accident, the control rods would be dropped down into the core. The reactivity worth of the control rods will be calculated for various positions to separately estimate the worth of each control rod, and the whole reactivity control system. Finally, control rod with the maximum worth will be localized to conduct basic safety test: at BOL the reactor should not startup if a single rod (maximum worth rod) is accidentally ejected from the TAP core.

The reactivity worth of the single control rod is defined as:

$$CRW = (k_{eff}^W - k_{eff}^I) \times 10^5 \text{ [pcm]} \quad (4.4)$$

where

k_{eff}^W = effective multiplication factor when control rod is fully withdrawn

k_{eff}^I = effective multiplication factor when control rod is fully inserted

The statistical error of the reactivity worth will be obtained using formula:

$$\delta CRW = \sqrt{(\delta k_{eff}^W)^2 + (\delta k_{eff}^I)^2} \quad (4.5)$$

where

$\delta k_{eff}^W, \delta k_{eff}^I$, = k_{eff} statistical error from Serpent output.

4.1.3 Axial Offset

Axial Offset Anomaly (AOA) refers to a neutron flux depression in the top of a nuclear reactor core, which complicates the reactor operation. This problem occurs in Pressurized Water Reactor (PWR) plants and leads to a variety of problems: increased local power peaking factors, lower than expected burnup and decreased control rod worth. The Axial Offset is defined as:

$$A/O = \frac{p_{top} - p_{bottom}}{p_{top} + p_{bottom}} \quad (4.6)$$

where

p_{top}, p_{bottom} = fraction of rated power in a top and bottom half of the core.

For the case of the TAP MSR, off-gas system components (e.g., sparger, entrainment separator) introduce small bubbles of inert gas (helium) into the fuel salt during operation. These helium bubbles entering into the core would then introduce unpredictable reactivity. Moreover, the diameter of the bubbles would increase as they rise from the bottom to the top of the core because of approximately the 140°C temperature difference between the reactor inlet and outlet. To take into account the effect of the gas present in the fuel salt, I will split the reactor core model into a few axial regions with different corresponding salt density (figure 4.1).

The ideal gas law must be employed to find the density of the fuel salt with gas bubbles¹ in it:

$$PV = nRT \quad (4.7)$$

where

P = pressure of the gas

V = volume of the gas

n = number of moles of the gas

$$R = 8.31 \frac{J}{K \cdot mol}$$

T = temperature of the gas.

Assuming the pressure in the vessel remains fixed and the number of moles of helium is constant, the helium volume change related to temperature change can be defined as:

$$V_i^{He} = \frac{V_0^{He}}{T_0} T_i \quad (4.8)$$

¹ Assuming we know the bubble number and size from the component design.

where

$$V_0^{He}, V_i^{He} = \text{total He volume in the salt with temperature } T_0, T_i$$

T_0, T_i = temperature of the salt in a lower and i^{th} axial layer.

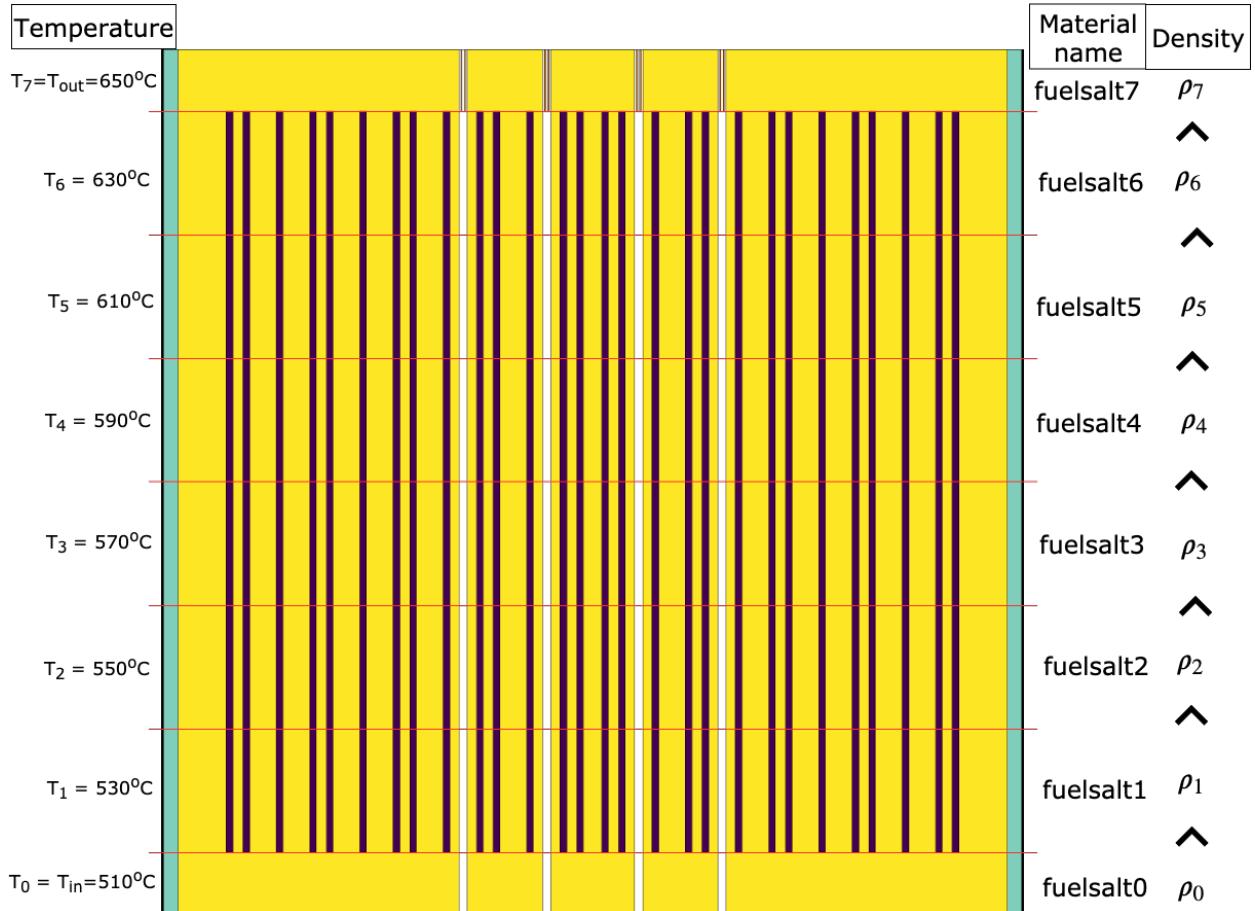


Figure 4.1: Preliminary schematic view showing the TAP model divided to multiple axial layers with different densities of the salt. This model will assume a linear temperature growth from the bottom to the top.

Using equations 4.1,4.8, density of the salt/helium mixture in each axial layer (ρ_i) will be calculated. Finally, a Serpent 2 calculation will be performed for the model with non-uniform axial density distribution in the fuel salt to determine axial neutron flux distribution and axial offset (A/O) in the TAP core.

4.2 Preliminary results

Temperature coefficients and control rod worth were calculated for the TAP reactor at startup as discussed in section 4.1.

4.2.1 Temperature coefficients and rod worth at BOL

Table 4.1 summarizes temperature effects on reactivity in the TAP core calculated for the initial fuel salt composition. The fuel temperature coefficient is -0.693 and -0.116 pcm $\Delta k/k \cdot K^{-1}$ when perturbing the salt temperature from 900K to 800K and from 900K to 1000K, respectively. The MTC and ITC are negative and relatively large for both cases. All three temperature coefficients when perturbing from 900K to 1000K did not match coefficients when perturbing from 900K to 800K because cross sections are nonlinear with temperature.

Table 4.1: Temperature coefficients for the TAP reactor at startup for different temperature perturbations.

Coefficient	From 900 to 800K [pcm/K]	From 900 to 1000K [pcm/K]
FTC	$-0.693 \pm 9.25 \times 10^{-2}$	$-0.116 \pm 9.33 \times 10^{-2}$
MTC	$-1.106 \pm 9.25 \times 10^{-2}$	$-1.195 \pm 9.27 \times 10^{-2}$
ITC	$-1.768 \pm 9.17 \times 10^{-2}$	$-1.301 \pm 9.27 \times 10^{-2}$

These reactivity coefficients will change most likely during operation due to changes in the neutron spectrum, salt volume fraction, core geometry, amount of fuel in the core and fuel composition. The FTC is expected to be more negative during operation as the spectrum thermalizes due to additional, retained fission products and actinides building up in the fuel salt. Temperature coefficient calculations will be repeated at middle-of-life and end-of-life to capture changes in these characteristics.

A configuration of 25 control rods (Figure 3.3) provides a reactivity worth of 1110 ± 9.7 pcm (1.1%) at BOL. The control rod worth is expected to become more negative during operation because the spectrum thermalizes and many fission product poisons exhibit larger absorption cross sections in the thermal energy range. At the same time, that effect would be

counteracted by actinide (particularly, plutonium) accumulation in the core. Thus, control rod worth change during the reactor operation must be estimated.

CHAPTER 5

FUTURE WORK AND PROPOSED SIMULATIONS

5.1 Summary

The need for this work has been shown by a summary of the current state of the art of MSR depletion simulator capabilities. The literature review in Chapter 1 concluded that most MSR depletion simulators typically assume ideal (rather than realistically constrained) poison removal rates for the nuclear system performance modeling. Moreover, most of the simulators assumed constant extraction efficiency vectors, which must be determined by the user in the input file and cannot be a function of other parameters. The Python toolkit, SaltProc v1.0, will directly couple with the Serpent 2 Monte Carlo depletion code for liquid-fueled MSR depletion simulation to enable realistic online reprocessing system modeling. The SaltProc v1.0 seeks to be a universal tool for fuel composition evolution analysis in MSRs with taking into account the complex fuel salt reprocessing system. Such reprocessing systems may consist of multiple components with variable removal efficiencies and rates. Moreover, these components can be connected in series, parallel, or a combination, which will be accurately treated in the SaltProc v1.0. Section 2.1 details the generic design of MSR fuel salt reprocessing systems. Section 2.3 describes the SaltProc v1.0 architecture and design that is required to successfully model comprehensive liquid-fueled MSRs with online fuel reprocessing systems.

Figure 5.1 shows an outline of this work. The current chapter details each Stage of the proposed work.

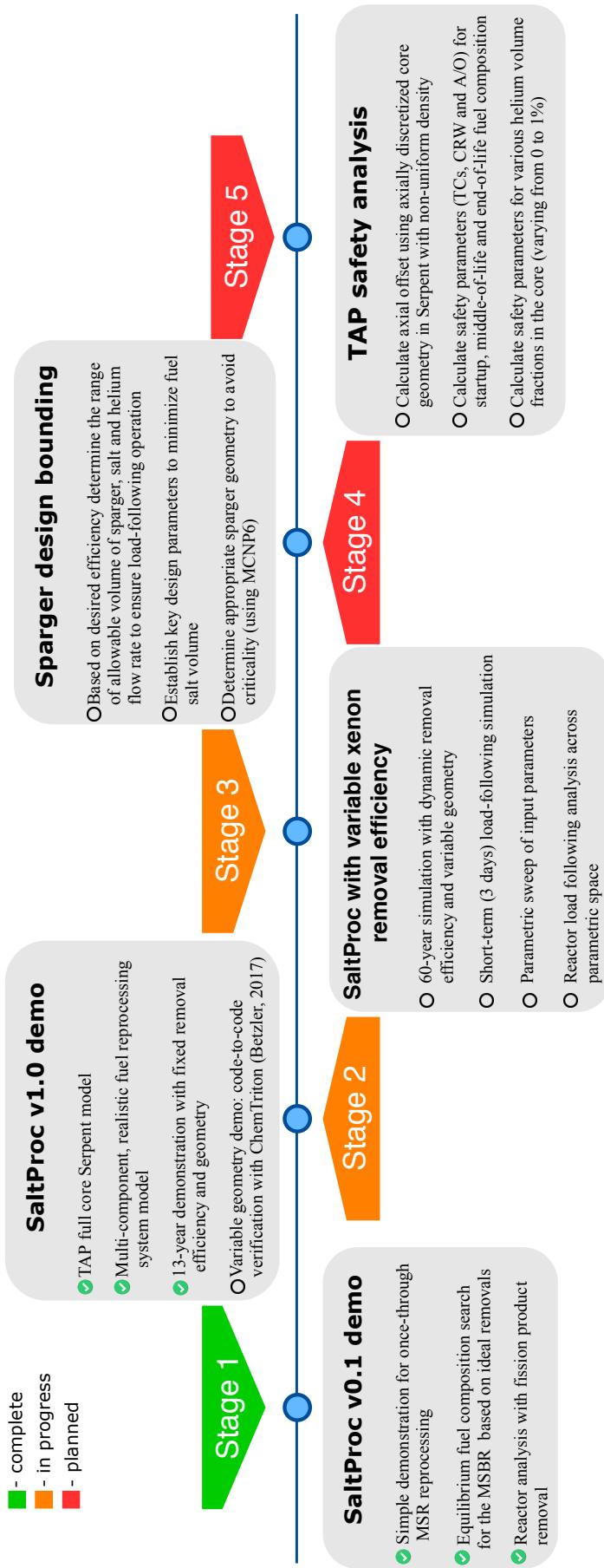


Figure 5.1: Workflow for the simulations proposed in this work.

5.2 Stage 1: Basic online reprocessing demonstration

In Stage 1, MSR online reprocessing simulation capabilities have been reviewed and summarized (Chapter 1). SaltProc v0.1 was demonstrated for simplified burnup calculation for the MSBR as a part of my M.Sc. thesis [39] and published paper [43]. These efforts illuminated depletion of the fuel salt in the MSBR for 60 years of operation and took into account the following processes:

1. FP removal from the salt with fixed, ideal extraction efficiencies (the fuel reprocessing system removed 100% of target poisons).
2. ^{233}Pa removal (100%) and feed of an equal mass of ^{233}U into the core (instantaneous ^{233}Pa decay to ^{233}U was assumed).
3. Fresh fertile material (^{232}Th) feed to maintain the constant fuel salt inventory.

Additionally, the effect of removing fission products from the fuel salt was investigated separately for a different group of FPs (noble gases, noble and seminoble metals, rare earth elements). As expected, removing fission products provides significant neutronic benefit and enables a longer core lifetime. Section 2.4 described key findings after completing Stage 1.

5.3 Stage 2: SaltProc v1.0 demonstration and validation for the TAP

Simulating a realistic multi-component fuel reprocessing system is important for calculating an accurate fuel salt composition. SaltProc v0.1 was completely refactored for modeling a complicated salt reprocessing system. To demonstrate SaltProc v1.0 capabilities, we have created a full-core TAP MSR model in Serpent 2 [58] which was described in detail in Section 3.1.3. Moreover, the multi-component fuel reprocessing system of the TAP was developed on this stage (Section 3.3). Section 3.3 also presented preliminary results of Stage 2. The Stage 2 demonstration case has following advantages over Stage 1:

- SaltProc v0.1 (Stage 1) approximated the fuel salt reprocessing system as a single “black” box, which removes the entire mass (100% removal efficiency) of processed

elements at once. In contrast, SaltProc v1.0 treats the fuel reprocessing system as a complex structure of components, each removing a specific set of elements with specific extraction efficiency.

- SaltProc v1.0 inherently checks mass conservation at each depletion step and dynamically calculates feed stream to maintain the fuel salt inventory constant.
- SaltProc v1.0 tracks the waste stream from each component.

The foremost future effort in this stage is to enable switching between multiple Serpent geometries during simulation. For the TAP concept, the number of moderator rods in the core varies from 1332 at the startup to 6700 at the EOL. The user will have an option to choose when SaltProc v1.0 should switch to next geometry: (1) after a specific depletion time (e.g., 18 months which is a common maintenance/refueling shutdown interval for LWRs); or (2) when the effective multiplication factor reaches a specific value (e.g., $1.00 < k_{eff} < 1.002$). Additionally, SaltProc v1.0 will correct the total fuel salt inventory in the primary loop to compensate for the core geometry change. Overall, the adjustable geometry capability will realistically simulate long-term (60 year) operation of the TAP reactor to obtain accurate fuel salt composition at different moments during operation.

Results obtained in Stage 2 will be used for code-to-code verification with ChemTriton/Shift results for full-core TAP core geometry from the most recent ORNL technical report TM-2017/475 [23]. Notably, the fuel salt composition evolution during the TAP reactor operation and corresponding core geometry are determinative for all next stages.

This work is developed with a test-driven development paradigm. Specifically, before any new functionality is implemented, a suite of tests is written, which as carefully define its expected behavior as possible. The code is then written to pass the test suite. In this way, the tool developed in this work is expected to be comprehensively tested in parallel with its development. Thus, after code-to-code verification with ChemTriton/Shift multiple-component integration tests will be added to the test harness to make sure that future changes in the code will not break previous functionality.

Test problems will help comprehensively define and confirm each unit of the demonstration functionality. These problems will include fundamental, information-passing tests as well as

more challenging multiple-component integration tests. Every unit of functionality within the toolkit will be tested as an integral part of development.

This milestone will result in a processing system model capable of simulating various liquid-fueled MSRs with multi-component fuel reprocessing systems but with constant separation efficiencies, defined at runtime. Additionally, this stage will demonstrate a key feature of the TAP MSR - adjusting the moderator rod configuration - which is necessary to maintain the reactor criticality during the 60-years lifetime.

5.4 Stage 3: Variable xenon extraction rate

When Stage 2 is complete, a series of extensions to the Stage 2 model will be pursued. These will incorporate extraction efficiencies as a function of many physical system design parameters (e.g., void fraction in the salt, helium bubble size). Mathematical correlations for the efficiencies will be taken from relationships in the literature [28, 59] and CFD simulations currently being conducted at the University of Illinois at Urbana-Champaign [29]. For demonstration proposes, just xenon removal efficiency will be defined as a function of many parameters (Section 2.1.1) due to limited data provided in the listed literature. For other fission products from the TAP reprocessing scheme (table 3.3), removal efficiencies will be defined based on the removal rates from the table, assuming time-independent extraction efficiency. This milestone will result in a realistic online reprocessing system model capable of modeling MSR systems with parameterized, realistically achievable process rates, and extraction efficiencies.

Another anticipated extension will test the TAP reactor ability to operate in a load-following regime. Short-term (3 days) depletion using SaltProc v1.0 will be performed with the core power changing in the $[0, 100\%]$ range with a ramp rate $10\%/\text{min}$ (to be competitive with natural gas peaking plants, which ramp at or above 10% of their capacity) [29]. Figure 5.2 shows the load curve selected to demonstrate the worst-case scenario of load-following:

1. Startup with fresh fuel and operating on 100% of hot full power (HFP) level for 40

hours to reach $^{135}\text{Xe}/^{135}\text{I}$ equilibrium;

2. Load-following power drop (0.1 HFP/min), from HFP to hot zero power (HZP);
3. Shutdown for 8 hours¹ to reach the ^{135}Xe peak;
4. Load-following power rise (0.1 HFP/min), from HZP to HFP.

This scenario can be considered as backing up solar power with nuclear on a high-solar penetration grid (e.g., in California).

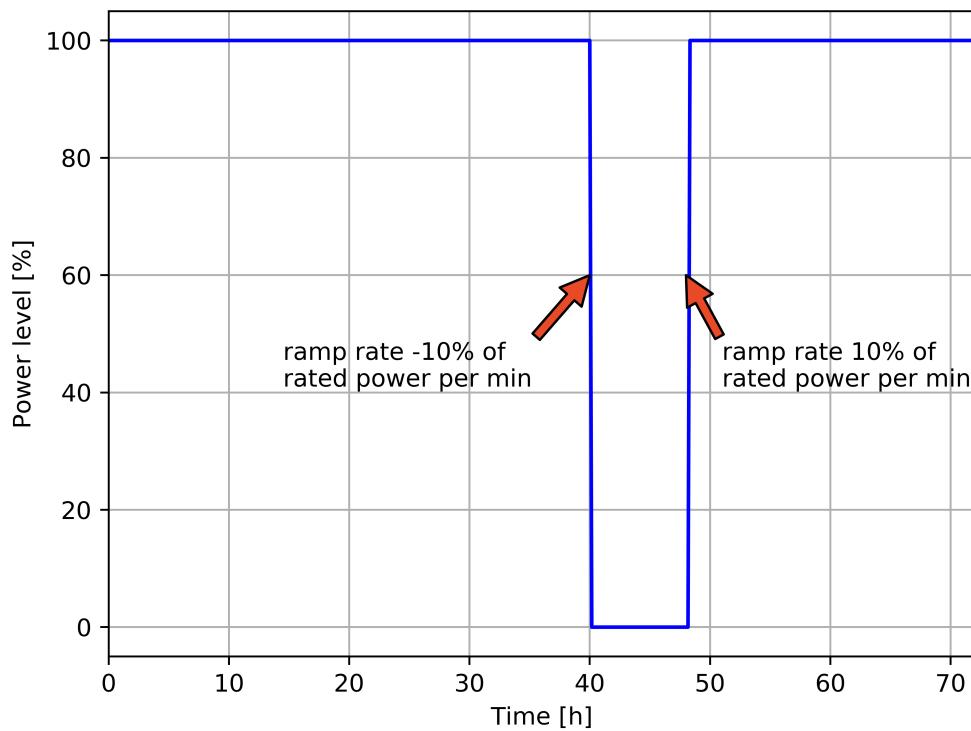


Figure 5.2: Tentative load curve for short-term load-following depletion simulation for the TAP reactor using SaltProc v1.0.

The depletion step time for short-term simulation will be varied in a range from 1 to 60 min to find a compromise between accuracy and computational cost. It is expected that load-following performance will be better at the BOL because the neutron energy spectrum thermalizes during the reactor operation. Thus, the short-term load-following simulation

¹At startup. Time after shutdown when ^{135}Xe concentration would reach maximum value greatly depends on neutron energy spectrum which for the TAP concept changes significantly during operation.

will be repeated for the BOL, the middle of life, and the EOL to assess the TAP concept performance in a load-following regime during the whole reactor lifetime.

Additionally, sensitivity analysis of input parameters in the xenon extraction correlation will be conducted to determine the range of key parameters (e.g., mass transfer coefficient, helium sparging rate, gas-liquid interfacial area, temperature) when load-following is possible for the TAP reactor in a worst-case power demand scenario. These multiple system configurations incorporating user-parametrized components in the fuel salt processing system will be collected and published in a *.json*-compatible database for use with the SaltProc v1.0 to encourage further research in this area.

5.5 Stage 4: Prototype design for the Xe removal system

As the model becomes capable of incorporating user-parametrized components with correlation-based extraction efficiency for the helium sparging component, constraints bounding a suitable sparger design will be determined and described. These design ranges (i.e., helium sparging rate) obtained from the previous Stage. The ultimate objective of the design is to ensure load-following operation during most of the operation period when minimizing the fuel salt inventory. That is, constrained optimization problem must be solved to minimize total fuel salt volume outside of the core. The target design parameters for the sparger include: the volume of sparger, helium flow rate, salt flow rate, and geometry.

Additionally, nuclear criticality safety analysis will be performed using MCNP6 [60] to confirm that the selected sparger design has a subcritical configuration. If the sparger geometry obtained during the optimization process is supercritical, the fission gas removal system would contain multiple spargers of smaller size connected in parallel. Total fuel salt volume and sparger size are expected to be smaller at the BOL and increase steadily as the neutron energy spectrum becomes softer.

5.6 Stage 5: TAP Safety Analysis

The objective of this Stage is to characterize neutronics limits related to load following. High-fidelity simulations will achieve this goal with the Serpent 2 Monte Carlo code. Specifically, changes in safety parameters (Section 4.1) will be evaluated for two time frames:

1. Long-time-scale changes in safety parameters should not compromise TAP MSR safety.
2. Load-following operation at key moments in the reactor lifetime (e.g., at startup, at the middle of life, at the end of life) must not result in significant changes in safety parameters.

Section 4.2 showed preliminary calculations of temperature coefficients and reactivity control system worth for the TAP at startup. The next step will develop a axially discretized core geometry in Serpent with non-uniform axial density distribution to estimate the axial power offset. Afterward, safety parameters will be calculated at the BOL, the middle of life, and the EOL to capture safety parameter variation over long time scales. Validation against previous work in a collaboration between Transatomic Power and ORNL [23, 16] will also be performed for confidence building. Additionally, analysis for different xenon removal efficiencies (i.e., in the range from 0 to 100%) will be performed to capture the effect of ^{135}Xe concentration on safety.

To analyze the impact of the load-following operation on TAP concept safety, safety parameter calculations will be repeated for the load-following transient. The combination of fuel and moderator temperature coefficients must remain strongly negative, and the reactivity worth of control rods must be sufficient to shut down the reactor for all times during load-following operation.

5.7 Conclusions

Details of gas removal and fuel salt processing systems in liquid-fueled MSRs have historically been conceptual rather than concrete. Usually, researchers assume ideal rather than realistically constrained poison extraction efficiency for reactor performance calculations.

This work will more realistically model an online molten salt processing system with a focus on the gas removal system of the prospective TAP MSR. SaltProc, a Python toolkit was developed as a part of this work. SaltProc couple directly with the Serpent 2 Monte Carlo burnup software to capture the evolution of fuel salt composition during reactor operation in the context of an online fuel processing system.

Modeling and simulation of the online reprocessing system in the MSR has shown promise in past research. Our work on simulating online fuel reprocessing for the thorium-fueled MSBR yielded interesting results: notable neutron energy spectrum shift and corresponding changes in safety parameters during operation. Additional preliminary work also showed promising results in modeling a simplified fuel processing system for the TAP MSR. These simulations motivate future work in modeling advanced liquid-fueled MSR plant designs.

To establish a feasible system design for molten salt fuel reprocessing, a more advanced model of the TAP MSR system with adjustable core geometry and realistically achievable extraction efficiencies will be developed. Extended SaltProc v1.0 will realistically capture the dynamics of fuels salt composition changes with higher accuracy. SaltProc v1.0 will also be employed to simulate the TAP MSR behavior in short-term transients to determine the feasibility of load following. Additionally, input parameters such as flow rates, bubble size, and the void fraction will be varied to determine the range of these parameters when the load following is possible for the TAP concept.

In addition to these simulations, several extensions are suggested to advance our preliminary work. First, the feasible design parameters of the sparger, critical component of the TAP gas removal system, will be optimized through sensitivity analysis of geometry and system conditions. To guarantee criticality safety, an MCNP6 simulation will be performed to define an appropriate sparger geometry. Further effort will focus on safety parameter evolution in the TAP reactor during lifetime (60 years), when the moderator rod configuration discretely changes. Finally, dynamics of the safety parameters will be investigated for a short-term case as was described in section 5.4: load following over three days with fixed moderator configuration and worst-case scenario of the power level change.

REFERENCES

- [1] IEA. Nuclear Power in a Clean Energy System. Technical report, IEA, May 2019.
- [2] World Energy Outlook 2017. Technical report, IEA, November 2017.
- [3] U. S. DoE. A technology roadmap for generation IV nuclear energy systems. In *Nuclear Energy Research Advisory Committee and the Generation IV International Forum*, pages 48–52, 2002.
- [4] Paul N. Haubenreich and J. R. Engel. Experience with the Molten-Salt Reactor Experiment. *Nuclear Technology*, 8(2):118–136, February 1970.
- [5] David LeBlanc. Molten salt reactors: A new beginning for an old idea. *Nuclear Engineering and Design*, 240(6):1644–1656, June 2010.
- [6] Massimiliano Fratoni, David Barnes, Ehud Greenspan, and Augusto Gandini. Design and Analysis of Molten Salt Reactor Fueled by TRU from LWR. *PHYSOR 2004*, 2004.
- [7] Dalin Zhang, Limin Liu, Minghao Liu, Rongshuan Xu, Cheng Gong, Jun Zhang, Chenglóng Wang, Suizheng Qiu, and Guanghui Su. Review of conceptual design and fundamental research of molten salt reactors in China. *International Journal of Energy Research*, 42(5):1834–1848, 2018.
- [8] Carlo Fiorina. *The molten salt fast reactor as a fast spectrum candidate for thorium implementation*. PhD, Politecnico Di Milano, March 2013.
- [9] V. Ignatiev, O. Feynberg, I. Gnidoi, A. Merzlyakov, A. Surenkov, V. Uglov, A. Zagritko, V. Subbotin, I. Sannikov, A. Toropov, V. Afonichkin, A. Bovet, V. Khokhlov, V. Shishkin, M. Kormilitsyn, A. Lizin, and A. Osipenko. Molten salt actinide recycler and transforming system without and with Th-U support: Fuel cycle flexibility and key material properties. *Annals of Nuclear Energy*, 64(Supplement C):408–420, February 2014.
- [10] Jaakko Leppanen, Maria Pusa, Tuomas Viitanen, Ville Valtavirta, and Toni Kaltiaisenaho. The Serpent Monte Carlo code: Status, development and applications in 2013. *Annals of Nuclear Energy*, 82:142–150, August 2014.
- [11] Ali Ahmad, Edward B. McClamrock, and Alexander Glasner. Neutronics calculations for denatured molten salt reactors: Assessing resource requirements and proliferation-risk attributes. *Annals of Nuclear Energy*, 75:261–267, January 2015.

- [12] H. F. Bauman, G. W. Cunningham III, J. L. Lucius, H. T. Kerr, and C. W. Jr Craven. Rod: A Nuclear and Fuel-Cycle Analysis Code for Circulating-Fuel Reactors. Technical Report ORNL-TM-3359, Oak Ridge National Lab., Tenn., January 1971.
- [13] C. W. Kee and L. E. McNeese. MRPP: multiregion processing plant code. Technical Report ORNL/TM-4210, Oak Ridge National Lab., 1976.
- [14] A. Nuttin, D. Heuer, A. Billebaud, R. Brissot, C. Le Brun, E. Liatard, J. M. Loiseaux, L. Mathieu, O. Meplan, E. Merle-Lucotte, H. Nifenecker, F. Perdu, and S. David. Potential of thorium molten salt reactorsdetailed calculations and concept evolution with a view to large scale energy production. *Progress in Nuclear Energy*, 46(1):77–99, January 2005.
- [15] M. Aufiero, A. Cammi, C. Fiorina, J. Leppnen, L. Luzzi, and M. E. Ricotti. An extended version of the SERPENT-2 code to investigate fuel burn-up and core material evolution of the Molten Salt Fast Reactor. *Journal of Nuclear Materials*, 441(13):473–486, October 2013.
- [16] Benjamin R. Betzler, Sean Robertson, Eva E. Davidson, Jeffrey J. Powers, Andrew Worrall, Leslie Dewan, and Mark Massie. Fuel cycle and neutronic performance of a spectral shift molten salt reactor design. *Annals of Nuclear Energy*, 119:396–410, September 2018.
- [17] Ian C. Gauld, Georgeta Radulescu, Germina Ilas, Brian D. Murphy, Mark L. Williams, and Dorothea Wiarda. Isotopic Depletion and Decay Methods and Analysis Capabilities in SCALE. *Nuclear Technology*, 174(2):169–195, May 2011.
- [18] Nicholas Tsoulfanidis. *The Nuclear Fuel Cycle*. American Nuclear Society, La Grange Park, Illinois, USA, 2013. 00177.
- [19] MCNP - A General Monte Carlo N-Particle Transport Code, 2004.
- [20] Carlo Fiorina, Manuele Aufiero, Antonio Cammi, Fausto Franceschini, Jiri Krepel, Lelio Luzzi, Konstantin Mikityuk, and Marco Enrico Ricotti. Investigation of the MSFR core physics and fuel cycle characteristics. *Progress in Nuclear Energy*, 68:153–168, September 2013.
- [21] J. J. Powers, T. J. Harrison, and J. C. Gehin. A new approach for modeling and analysis of molten salt reactors using SCALE. In *Transactions of the American Nuclear Society*, Sun Valley, ID, USA, July 2013. American Nuclear Society, 555 North Kensington Avenue, La Grange Park, IL 60526 (United States).
- [22] Jinsu Park, Yongjin Jeong, Hyun Chul Lee, and Deokjung Lee. Whole core analysis of molten salt breeder reactor with online fuel reprocessing. *International Journal of Energy Research*, 39(12):1673–1680, October 2015.

- [23] Benjamin R. Betzler, Sean Robertson, TAP Eva E. Davidson, Jeffrey J. Powers, and Andrew Worrall. Assessment of the Neutronic and Fuel Cycle Performance of the Transatomic Power Molten Salt Reactor Design. Technical Report ORNL/TM-2017/475, ORNL, 2017.
- [24] GitHub. GitHub: Build software better, together, 2015.
- [25] R. C. Robertson. Conceptual Design Study of a Single-Fluid Molten-Salt Breeder Reactor. Technical Report ORNL-4541, ORNL, January 1971.
- [26] R. J. Kedl and A. Houtzeel. DEVELOPMENT OF A MODEL FOR COMPUTING Xe-135 MIGRATION IN THE MSRE. Technical report, Oak Ridge National Lab., Tenn., 1967.
- [27] R. B. Briggs. Molten-salt reactor program. Semiannual progress report. Technical Report ORNL-4396, Oak Ridge National Lab., Tenn., February 1969.
- [28] F. N. Peebles. Removal of Xenon-135 from Circulating Fuel Salt of the MSBR by Mass Transfer to Helium Bubbles. Technical Report ORNL-TM-2245, Oak Ridge National Laboratory, Oak Ridge, TN, United States, 1968.
- [29] Kathryn Huff. Enabling Load-Following Capability in the TAP MSR, December 2018.
- [30] R. B. Briggs. Molten-Salt Reactor Program semiannual progress report for period ending July 31, 1964. Technical Report Archive and Image Library ORNL-3708, Oak Ridge National Laboratory, Oak Ridge, TN, United States, 1964.
- [31] Sylvie. Delpech, Cline Cabet, Cyrine Slim, and Grard S. Picard. Molten fluorides for nuclear applications. *Materials Today*, 13(12):34–41, December 2010.
- [32] X. Doligez, D. Heuer, E. Merle-Lucotte, M. Allibert, and V. Ghetta. Coupled study of the Molten Salt Fast Reactor core physics and its associated reprocessing unit. *Annals of Nuclear Energy*, 64(Supplement C):430–440, February 2014.
- [33] M. E. Whatley, L. E. McNeese, W. L. Carter, L. M. Ferris, and E. L. Nicholson. Engineering development of the MSBR fuel recycle. *Nuclear Applications and Technology*, 8(2):170–178, 1970.
- [34] W. L. Carter and E. L. Nicholson. DESIGN AND COST STUDY OF A FLUORINATION-REDUCTIVE EXTRACTION-METAL TRANSFER PROCESSING PLANT FOR THE MSBR. Technical Report ORNL-TM-3579, Oak Ridge National Lab. (ORNL), Oak Ridge, TN (United States), January 1972.
- [35] Kirk Sorensen. One-Fluid MSBR Chemical Processing -, 2006.
- [36] Jaakko Leppanen, Ville Hovi, Timo Ikonen, Joona Kurki, Maria Pusa, Ville Valtavirta, and Tuomas Viitanen. The Numerical Multi-Physics project (NUMPS) at VTT Technical Research Centre of Finland. *Annals of Nuclear Energy*, 84:55–62, October 2015.

- [37] L. Dagum and R. Menon. OpenMP: an industry standard API for shared-memory programming. *IEEE Computational Science and Engineering*, 5(1):46–55, January 1998.
- [38] OECD/NEA. The JEFF-3.1.2 Nuclear Data Library. Technical Report JEFF Report 24, OECD/NEA Data Bank, OECD/NEA, 2014.
- [39] Andrei Rykhlevskii. Advanced online fuel reprocessing simulation for Thorium-fueled Molten Salt Breeder Reactor. Master’s thesis, University of Illinois at Urbana-Champaign, April 2018.
- [40] Andrei Rykhlevskii, Jin Whan Bae, and Kathryn Huff. arfc/saltproc: Code for online reprocessing simulation of molten salt reactor with external depletion solver SERPENT. *Zenodo*, March 2018.
- [41] The HDF Group. Hierarchical data format, version 5, 1997.
- [42] Anthony Scopatz, Paul K. Romano, Paul P. H. Wilson, and Kathryn D. Huff. PyNE: Python for Nuclear Engineering. In *Transactions of the American Nuclear Society*, volume 107, San Diego, CA, USA, November 2012. American Nuclear Society.
- [43] Andrei Rykhlevskii, Jin Whan Bae, and Kathryn D. Huff. Modeling and simulation of online reprocessing in the thorium-fueled molten salt breeder reactor. *Annals of Nuclear Energy*, 128:366–379, June 2019.
- [44] Holger Krekel, Bruno Oliveira, Ronny Pfannschmidt, Floris Bruynooghe, Brianna Laugher, and Florian Bruhin. pytest: Python testing tool, 2004.
- [45] Georg Brandl. Sphinx: Python Documentation Generator, 2009.
- [46] Andrei Rykhlevskii, Alexander Lindsay, and Kathryn D. Huff. Full-core analysis of thorium-fueled Molten Salt Breeder Reactor using the SERPENT 2 Monte Carlo code. In *Transactions of the American Nuclear Society*, Washington, DC, United States, November 2017. American Nuclear Society.
- [47] Transatomic Power Corporation. Technical White Paper. White Paper 2.1, Transatomic Power Corporation, Cambridge, MA, United States, November 2016.
- [48] Transatomic Power Corporation. Neutronics Overview. White Paper 1.1, Transatomic Power Corporation, Cambridge, MA, United States, November 2016.
- [49] Benjamin R. Betzler, Jeffrey J. Powers, Andrew Worrall, L. Dewan, S. Robertson, and Mark Massie. Two-Dimensional Neutronic and Fuel Cycle Analysis of the Transatomic Power Molten Salt Reactor. Technical Report ORNL/TM-2016/742, Oak Ridge National Lab.(ORNL), 2017.
- [50] Jaakko Leppanen. Serpent – a Continuous-energy Monte Carlo Reactor Physics Burnup Calculation Code. *VTT Technical Research Centre of Finland, Espoo, Finland*, 2013.

- [51] Transatomic Power. Transatomic Reactor Documentation, March 2019. original-date: 2018-11-08T00:08:31Z.
- [52] Davide Rodrigues. *Actinide/lanthanide separation in molten salt media : application to the MSFR fuel reprocessing*. PhD thesis, 2015.
- [53] L. E. McNEESE. Engineering Development Studies for Molten-Salt Breeder Reactor Processing No. 2. Technical Report ORNL-TM-3137, Oak Ridge National Lab., Oak Ridge, TN, United States, 1971.
- [54] Sylvie Delpech. Possible routes for pyrochemical separation: Focus on the reductive extraction in fluoride media. *Pure and Applied Chemistry*, 85(1):71–87, 2012.
- [55] B. R. Betzler, J. J. Powers, N. R. Brown, and B. T. Rearden. Implementation of Molten Salt Reactor Tools in SCALE. In *Proc. M&C 2017 - International Conference on Mathematics & Computational Methods Applied to Nuclear Science and Engineering*, Jeju, Korea, April 2017.
- [56] G. J. Janz, G. L. Gardner, Ursula Krebs, and R. P. T. Tomkins. Molten Salts: Volume 4, Part 1, Fluorides and Mixtures Electrical Conductance, Density, Viscosity, and Surface Tension Data. *Journal of Physical and Chemical Reference Data*, 3(1):1–115, January 1974.
- [57] S Yamanaka, K Yoshioka, M Uno, M Katsura, H Anada, T Matsuda, and S Kobayashi. Thermal and mechanical properties of zirconium hydride. *Journal of Alloys and Compounds*, 293-295:23–29, December 1999.
- [58] Anshuman Chaube, Daniel O’Grady, Andrei Rykhlevskii, and Kathryn D Huff. TAP MSR model for Serpent 2. *Zenodo*, 2019.
- [59] C. H. Gabbard. Development of an Axial-Flow Centrifugal Gas Bubble Separator for Use in Msr Xenon Removal System. Technical Report ORNL-TM-4533, Oak Ridge National Lab., Tenn. (USA), March 1974.
- [60] C. J. Werner, J. S. Bull, C. J. Solomon, F. B. Brown, G. W. McKinney, M. E. Rising, D. A. Dixon, R. L. Martz, H. G. Hughes, and L. J. Cox. MCNP6. 2 release notes. *Tech. Rep. LA-UR-18-20808, Los Alamos National Laboratory, Los Alamos, NM*, 2018.