Chapter 5

AHTR Optimization Preliminary Work

This chapter demonstrates the preliminary work completed for AHTR optimization. I used REALM to apply genetic algorithms to maximize k_{eff} in a single AHTR fuel slab. Then, I presented spatial and energy homogenizations for applications to AHTR multiphysics simulations. The `dissertation-results` Github repository contains all the scripts, results, and plots shown in this chapter [**?**].

5.1 REALM Optimization: AHTR Fuel Slab

5.1.1 Problem Definition

This demonstration explores how inhomogeneous fuel distributions impact k_{eff} compared with homogenous fuel distributions customary in most reactor designs. I use OpenMC v0.12.0 for these neutronics calculations with the ENDF/B-VII.1 data library [12]. The reactor core explored is a straightened slab from the FHR benchmark's AHTR design. Figure 5.1 illustrates the straightened fuel slab. The slab has $27.1 \times 3.25 \times 1.85$ cm^3 dimensions with periodic boundary conditions in the x-y axis and reflective boundary conditions in the z axis. I use the same materials as in the FHR benchmark, except for the homogenization of each TRISO particle's four outer layers: porous carbon buffer, inner pyrolytic carbon, silicon carbide layer, and the outer pyrolytic carbon. The TRISO particles' dimensions remain the same. Table 5.1 reports the k_{eff} for this original straightened AHTR configuration with and without the outer layer TRISO homogenization. The TRISO particle outer four-layer

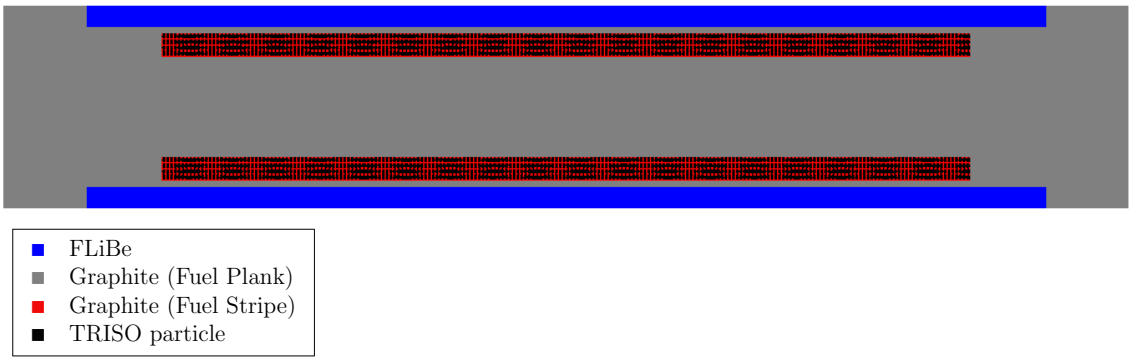


Figure 5.1: Straightened Advanced High Temperature Reactor (AHTR) fuel slab.

Table 5.1: Straightened Advanced High Temperature Reactor (AHTR) fuel slab's k_{eff} for case with no TRISO homogenization and case with homogenization of the four outer layers. Both simulations were run on one BlueWaters XE Node.

TRISO Homogenization	k_{eff}	**Simulation time [s]**
None	1.38548 ± 0.00124	233
Four outer layers	1.38625 ± 0.00109	168

homogenization resulted in a 30% speed-up without compromising accuracy with k_{eff} values within each other's uncertainty.

The REALM optimization problem's objective is to maximize the slab's k_{eff} by varying the TRISO particle packing fraction across the slab while keeping the total packing fraction constant at 0.0979. This total packing fraction is consistent with the original straightened slab with TRISO particles in fuel stripes (Figure 5.1). I divided the slab into ten slices along the x-axis between the FLiBe and graphite buffers, resulting in ten $2.31 \times 2.55 \times 1.85$ cm^3 slices. A sine distribution governs the TRISO particle packing fraction's distribution across slices:

$$PF(x) = (a \cdot sin(b \cdot x + c) + 2) \cdot NF \tag{5.1}$$

where

PF = packing fraction [−]

a = amplitude, peak deviation of the function from zero [−]

b = angular frequency, rate of change of the function argument $[\frac{radians}{cm}]$

c = phase, the position in its cycle the oscillation is at t = 0 $[radians]$

x = midpoint value for each slice $[cm]$

NF = Normalization factor [−]

I collected and normalized the sine distribution's value at each of the ten x-slices' midpoints by the total packing fraction to ensure a consistent number of TRISO particles in the slab. For example, a packing fraction distribution of $PF(x) = \big(0.5 \cdot sin(\frac{\pi}{3} \cdot x + \pi) + 2\big) \cdot NF$, results in the following packing fractions for the ten slices: 0.103, 0.120, 0.049, 0.138, 0.076, 0.081, 0.136, 0.048, 0.125, and 0.098. Figure 5.2 shows this sine distribution, highlights the packing fraction at the respective midpoints, and displays the slab's x-y axis view with packing fraction varying based on this sine distribution.

In REALM, a genetic algorithm varies the a, b, and c variables to find a combination that produces a packing fraction distribution that maximizes the slab's k_{eff}. I defined a, b, and c's upper and lower bounds as:

- $0 < a < 2$
- $0 < b < \frac{\pi}{2}$
- $0 < c < 2\pi$

I selected a variable's bounds to keep the sine distribution from falling below zero. The b and c variable bounds spread wide enough to allow the genetic algorithm to explore various sine distributions. The OpenMC evaluator calculates k_{eff}. OpenMC runs each simulation

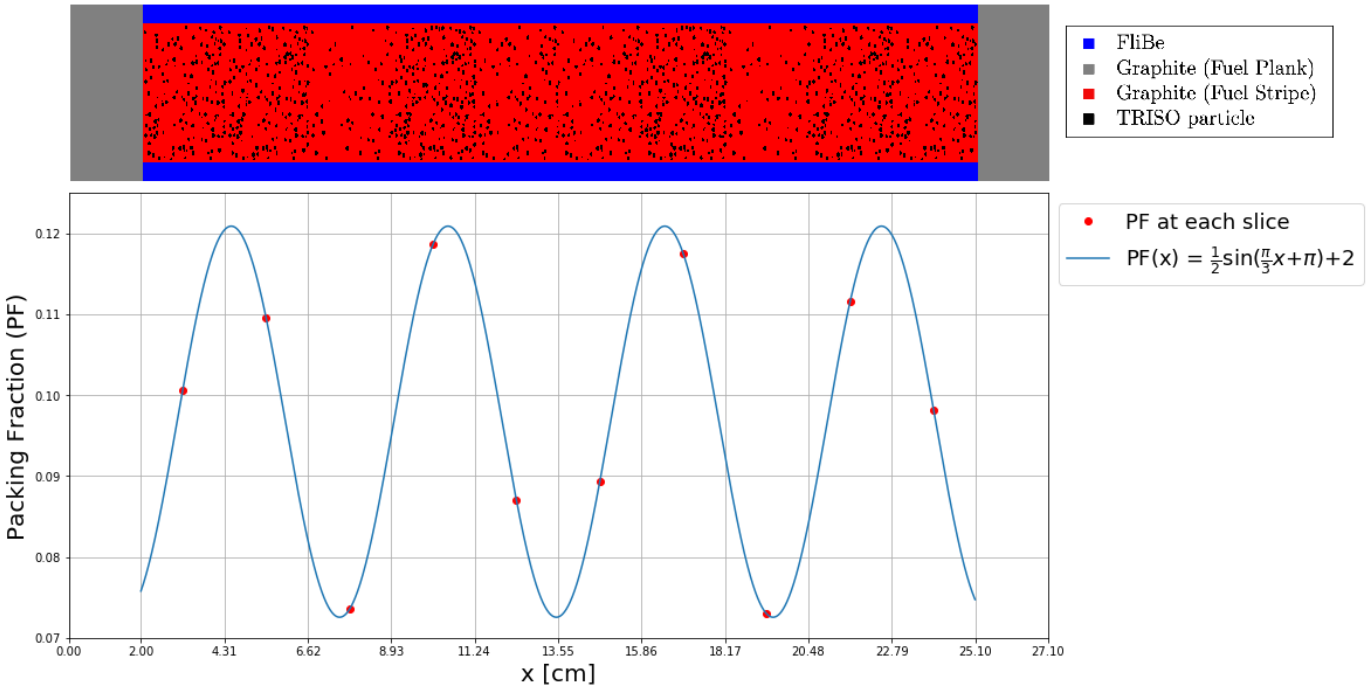


Figure 5.2: Below: $PF(x) = (0.5\ sin(\frac{\pi}{3}x + \pi) + 2) \times NF$ sine distribution with red points indicating the packing fraction at each slice. Above: Straightened Advanced High Temperature Reactor (AHTR) fuel slab with varying TRISO particle distribution across ten slices based on the sine distribution.

with 80 active cycles, 20 inactive cycles, and 8000 particles to reach ~130pcm uncertainty. Figure 5.3 shows the REALM input file for this genetic algorithm optimization problem. `ahtr_slab_openmc.py` is the template OpenMC straightened AHTR slab script that accepts a, b and c from REALM, calculates packing fraction distribution, and assigns packing fraction values to each fuel slice. Subsequently, REALM runs the templated OpenMC script to generate k_{eff}.

5.1.2 Hyperparameter Search

In REALM's input file, the user defines the genetic algorithm's hyperparameters. A good hyperparameter set guides the optimization process by balancing exploitation and exploration to find an optimal solution quickly and accurately. Finding a good hyperparameter set requires a trial-and-error process.

I performed the hyperparameter search with a coarse-to-fine random sampling scheme, whose advantages I previously discussed in Section 2.4.2. The hyperparameters varied included population size, number of generations, mutation probability, mating probability, selection operator, selection operator's number of individuals, selection operator's tournament size, mutation operator, and mating operator. I started with 25 coarse experiments and fine-tuned the hyperparameters with 15 more experiments. For each genetic algorithm experiment, I held the number of OpenMC evaluations constant at 600. The number of evaluations correlated the population size and number of generations. I randomly sampled population size and used the following equation to calculate the number of generations:

$$\text{no. of generations} = \frac{\text{no. of evaluations}}{\text{population size}} \tag{5.2}$$

Table 5.2 shows the lower and upper bounds used for each hyperparameter's random sampling.

The initial 25 coarse experiments' sought to narrow down the hyperparameters to find

```
{
    "control_variables": {
        "a": {"min": 0.0, "max": 2.0},
        "b": {"min": 0.0, "max": 1.57},
        "c": {"min": 0.0, "max": 6.28},
    },
    "evaluators": {
        "openmc": {
            "input_script": "ahtr_slab_openmc.py",
            "inputs": ["a", "b", "c"],
            "outputs": ["keff"],
            "keep_files": false,
        }
    },
    "constraints": {"keff": {"operator": [">="], "constrained_val": [1.0]}},
    "algorithm": {
        "objective": "max",
        "optimized_variable": "keff",
        "pop_size": 60,
        "generations": 10,
        "mutation_probability": 0.23,
        "mating_probability": 0.46,
        "selection_operator": {"operator": "selTournament", "k": 15, "tournsize": 5},
        "mutation_operator": {
            "operator": "mutPolynomialBounded",
            "eta": 0.23,
            "indpb": 0.23,
        },
        "mating_operator": {"operator": "cxBlend", "alpha": 0.46},
    },
}
```

Figure 5.3: Reactor Evolutionary Algorithm Optimizer (REALM) JSON input file to maximize k_{eff} in the straightened Advanced High Temperature Reactor (AHTR) fuel slab by varying packing fraction distribution with control variables a, b, and c.

Table 5.2: Hyperparameter search is conducted in three phases: *Coarse Search*, *Fine Search 1*, *Fine Search 2*. Each hyperparameter's lower and upper bounds for each search phase are listed.

Hyperparameter	**Type**	**Coarse Search Bounds**	**Fine Search 1 Bounds**	**Fine Search 2 Bounds**
Experiments	-	0 to 24	24 to 34	35 to 39
Population size (pop)	Continuous	$10 < x < 100$	$20 < x < 60$	60
Mutation probability	Continuous	$0.1 < x < 0.4$	$0.2 < x < 0.4$	$0.2 < x < 0.3$
Mating probability	Continuous	$0.1 < x < 0.6$	$0.1 < x < 0.3$	$0.45 < x < 0.6$
Selection operator	Discrete	`SelTournament`, `SelBest`, `SelNSGA2`	`SelTournament`, `SelBest`, `SelNSGA2`	`SelTournament`
Selection individuals	Continuous	$\frac{1}{3}pop < x < \frac{2}{3}pop$	$\frac{1}{3}pop < x < \frac{2}{3}pop$	15
Selection tournament size (only for SelTournament)	Continuous	$2 < x < 8$	$2 < x < 8$	5
Mutation operator	Discrete	`mutPolynomialBounded`	`mutPolynomialBounded`	`mutPolynomialBounded`
Mating operator	Discrete	`cxOnePoint`, `cxUniform`, `cxBlend`	`cxOnePoint`, `cxUniform`, `cxBlend`	`cxOnePoint`, `cxBlend`

a smaller set of hyperparameter bounds that produce higher k_{eff} values. Figure 5.4 shows the hyperparameters' plotted against each other with a third color dimension representing the k_{effave} value in each experiment's final generation. Lighter scatter points indicate higher final population k_{effave} values, which suggests better hyperparameter sets. I plotted the hyperparameters against each other to visualize the interdependence between hyperparameters. From the coarse hyperparameter search, I noticed the following trends:

- Mutation probability has a higher k_{effave}, between 0.2 and 0.4.
- Mating probability has a higher k_{effave}, between 0.1 and 0.3.
- Population size has a higher k_{effave}, between 20 and 60.
- No obvious interdependence between hyperparameters.

Next, I proceeded to the fine searches. From Figure 5.4, I narrowed down population size, mutation probability, and mating probability bounds, as shown in Table 5.2's *Fine Search 1 Bounds* column. I found no significant trends in the other hyperparameters, so

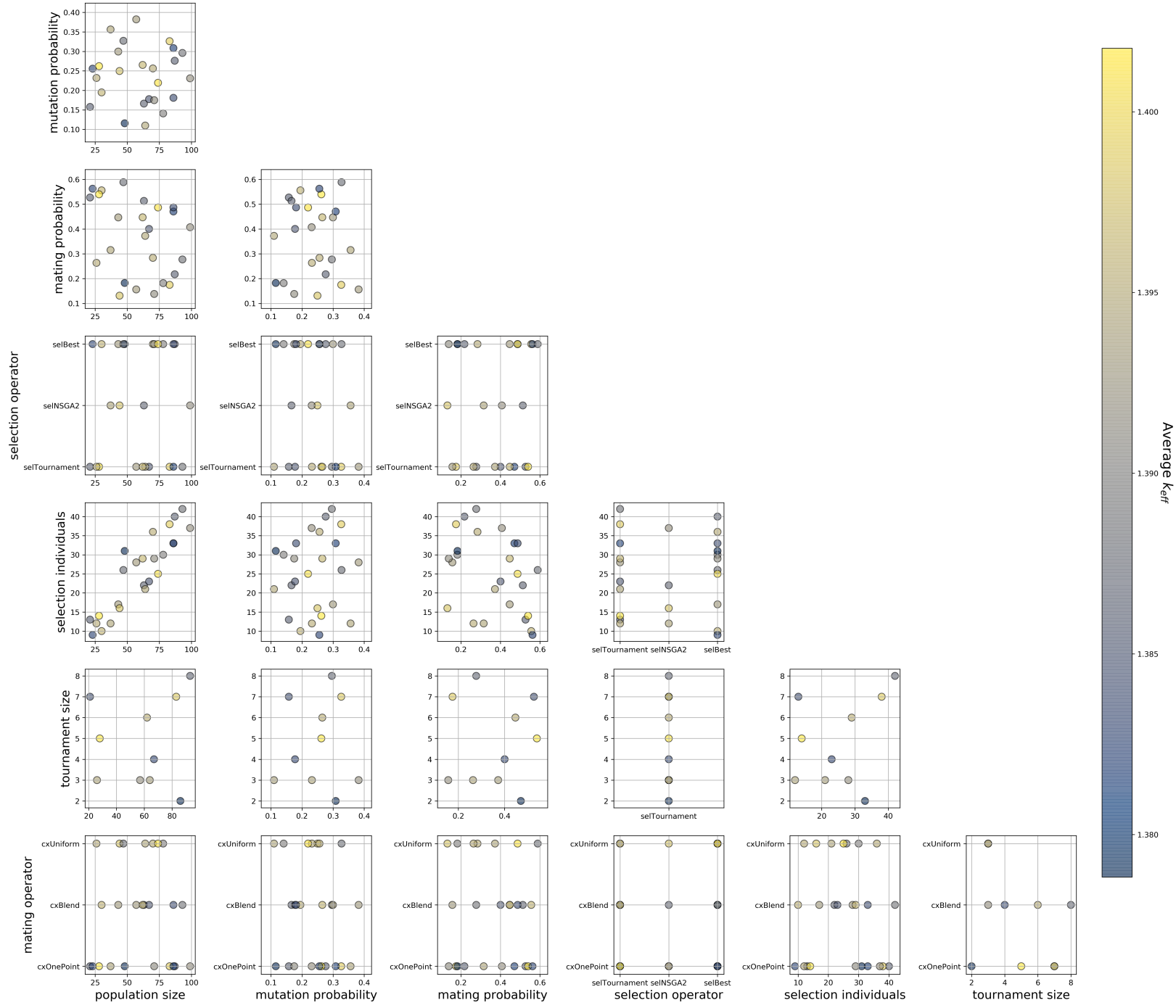


Figure 5.4: Coarse hyperparameters search's results. Hyperparameter values are plotted against each other with a third color dimension representing each experiment's final population's k_{effave}.

I left them as is. I ran ten more experiments (25 to 34), sampling hyperparameters from the *Fine Search 1 Bounds*. From these results, I conducted a second fine search with five experiments (35 to 39) with further tuned hyperparameter bounds, as shown in Table 5.2's *Fine Search 2 Bounds* column. I determined these new hyperparameter bounds based on these reasons:

- Mutation probability has a higher k_{effave}, between 0.2 and 0.3.
- I overlooked k_{effave} peaking at mating probability between 0.45 and 0.6 in the previous *Fine Search 1*, thus shifted the bounds.
- The highest k_{effave} occurred for `selTournament`.
- I narrowed down mating operator options to `cxBlend` and `cxOnePoint` since they had higher k_{effave}.
- I selected arbitrary numbers for population size, selection individuals, and tournament size since they did not correlate with k_{effave} values.

Figure 5.5 shows the relationship between hyperparameter values and a, b, c control parameters, final generation k_{effmax}, and final generation k_{effave}. The coarse experiments' scatter points are 50% transparent, while the fine experiments' scatter points are opaque. In Figure 5.5, on average, the fine experiments (opaque scatter points) have higher k_{effave}, which indicates that the hyperparameter search process met its objective of finding hyperparameter bounds that enable quicker and more accurate optimization.

Table 5.3 shows the hyperparameters for the five experiments with the highest final generation k_{effave}. Figure 5.6 shows the packing fraction distributions that produced the k_{effmax} from the top five experiments. Four experiments had similar packing fraction distributions peaking at approximately 0.23 in the slab's center. In contrast, one experiment had an exponential-like distribution with a peak packing fraction of 0.31 at the slab's side.

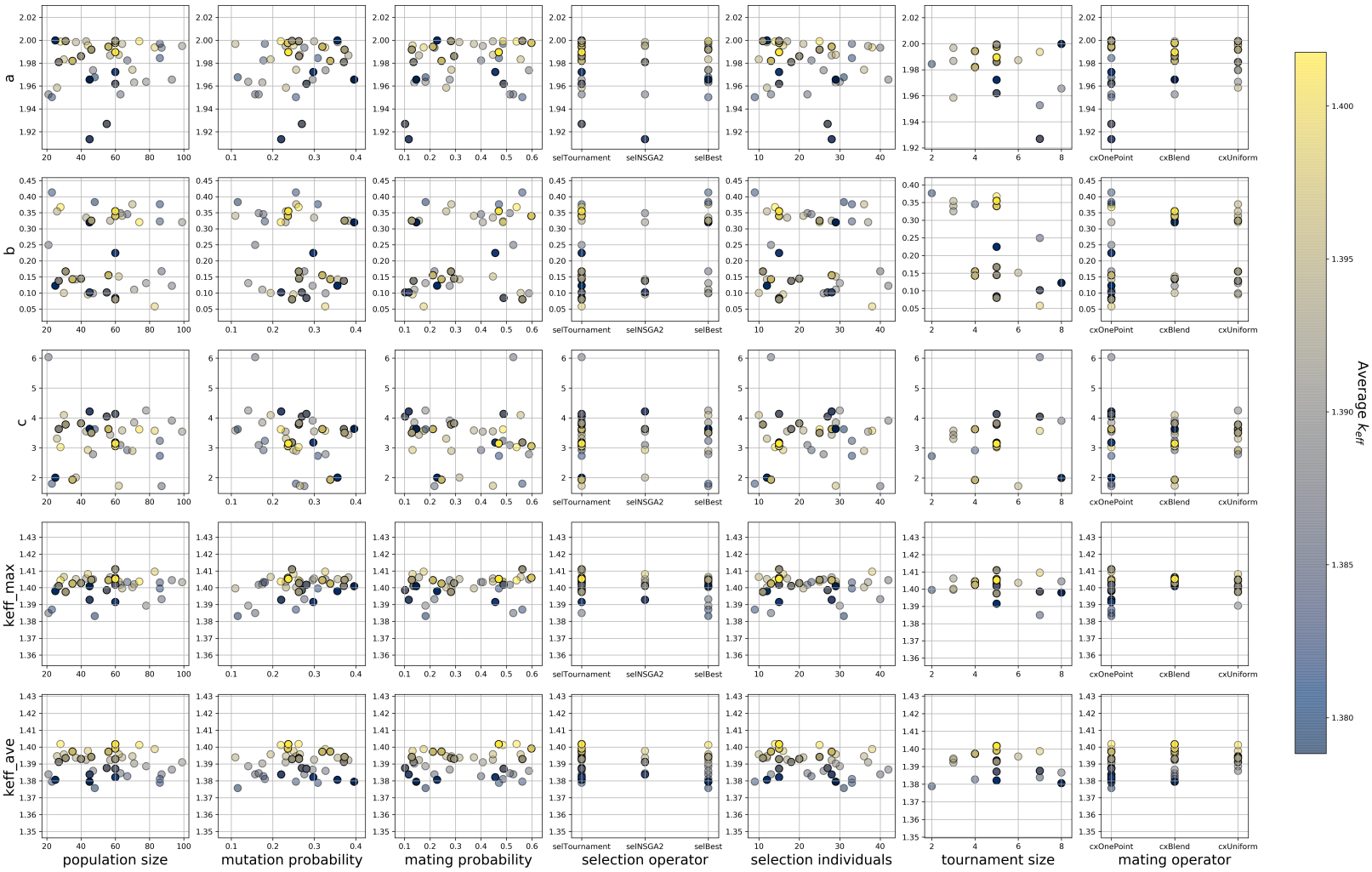


Figure 5.5: Hyperparameters search's results for all 40 experiments (coarse and fine). I plotted the hyperparameters against: a,b,c control parameters, each experiment's final generation k_{effmax}, and final generation k_{effave} with a third dimension representing each experiment's final population's k_{effave}. Coarse experiments' (0 to 24) scatter points are 50% transparent, while the fine experiments' (24 to 39) scatter points are opaque.

Table 5.3: Control Parameters, k_{eff} results, and hyperparameter values for the five hyperparameter search experiments with the highest final generation k_{effave}.

Control/Output Parameters	**Experiment 6**	**Experiment 15**	**Experiment 24**	**Experiment 36**	**Experiment 39**
k_{effave} [-]	1.39876	1.40155	1.40118	1.39906	1.40165
k_{effmax} [-]	1.40954	1.40440	1.40365	1.40590	1.40519
a [-]	1.993	1.998	1.999	1.997	1.989
b [$\frac{radians}{cm}$]	0.057	0.367	0.320	0.339	0.354
c [radians]	3.571	3.022	3.615	3.053	3.143
Hyperparameter					
Population size	83	28	74	60	60
Generations	8	22	9	10	10
Mutation probability	0.32	0.26	0.21	0.23	0.23
Mating probability	0.17	0.53	0.48	0.59	0.46
Selection operator	`selTournament`	`selTournament`	`selBest`	`selTournament`	`selTournament`
Selection individuals	38	14	25	15	15
Selection tournament size	7	5	-	5	5
Mutation operator	`mutPolynomial Bounded`	`mutPolynomial Bounded`	`mutPolynomial Bounded`	`mutPolynomial Bounded`	`mutPolynomial Bounded`
Mating operator	`cxOnePoint`	`cxOnePoint`	`cxUniform`	`cxBlend`	`cxBlend`

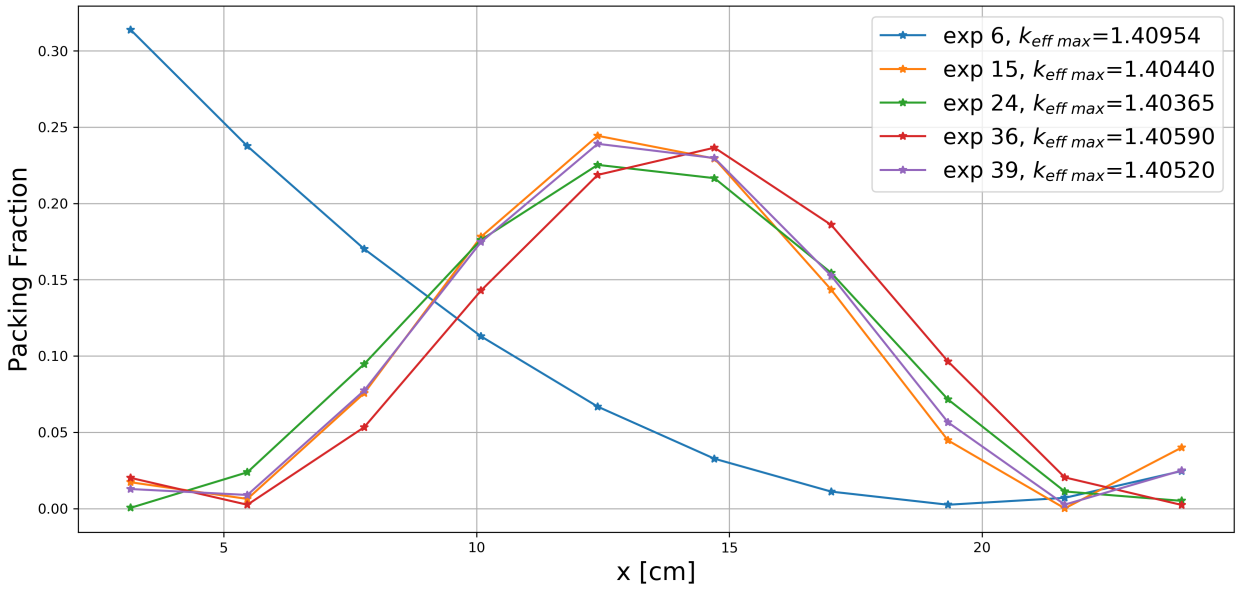


Figure 5.6: Packing fraction distribution across the x-axis of the Advanced High Temperature Reactor (AHTR) slab for the five hyperparameter search experiments with the highest final generation k_{effave}.

The similar final packing fraction distributions demonstrate genetic algorithms' robustness to find the optimal global solutions with different hyperparameters.

I ran these simulations on the BlueWaters supercomputer [54]. In each REALM simulation, each generation runs a population size number of individual OpenMC simulations. Each OpenMC simulation takes approximately 13 minutes to run on a single BlueWaters XE node. With approximately 600 OpenMC evaluations per REALM simulation, the REALM simulation takes about 130 BlueWaters node-hours. The hyperparameter search ran 40 REALM simulations, thus using approximately 5200 node-hours.

5.1.3 Results for Best Hyperparameter Set

I define the best-performing hyperparameter set as the experiment that produces the highest k_{effave} in its final generation. *Fine Search 2*'s experiment 39 produces the best performing hyperparameter set, shown in Table 5.3, with center-peaking packing fraction distribution of k_{effmax} = 1.40519. Experiment 39's k_{effmax} exceeds the original straightened AHTR configuration's k_{eff} by ~ 2000pcm, proving that optimizing inhomogenous fuel distributions enables better neutronics. Figure 5.7 shows the packing fraction distribution that produced k_{effmax} = 1.40519.

Figures 5.8a and 5.8b show the k_{eff} evolution and packing fraction distribution through the best performing 39^{th} experiment's generations. The k_{effmax} converged quickly by generation 1; however, this usually does not occur. The genetic algorithm optimizes stochastically, resulting in the possibility that the algorithm randomly samples a control parameter set that maximizes the objective function early in the optimization process. The k_{effave} demonstrates how each generation's average k_{eff} converges towards a higher value with each generation's improvements. To demonstrate how the genetic algorithm optimization process usually goes, Figures 5.9a and 5.9b show the k_{eff} evolution and packing fraction distribution through the second-best performing 15^{th} experiment's generations. Experiment 15 demonstrates how both maximum and average k_{eff} converge towards a higher k_{eff} with improvements from

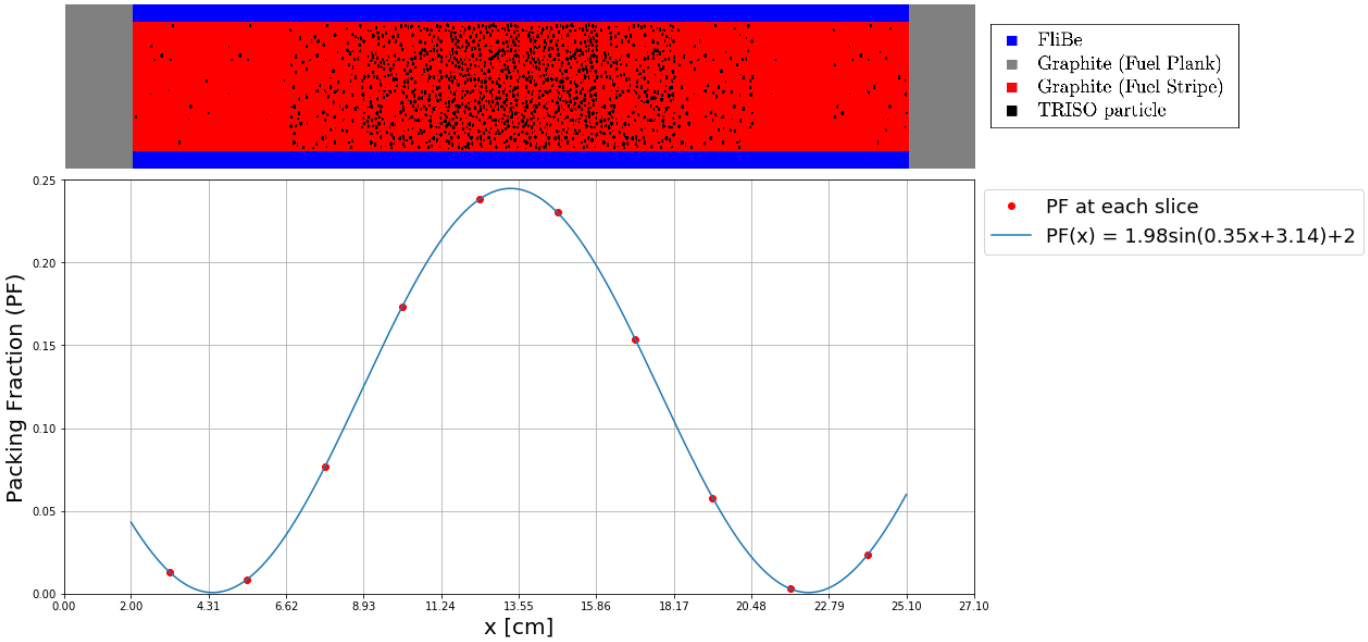
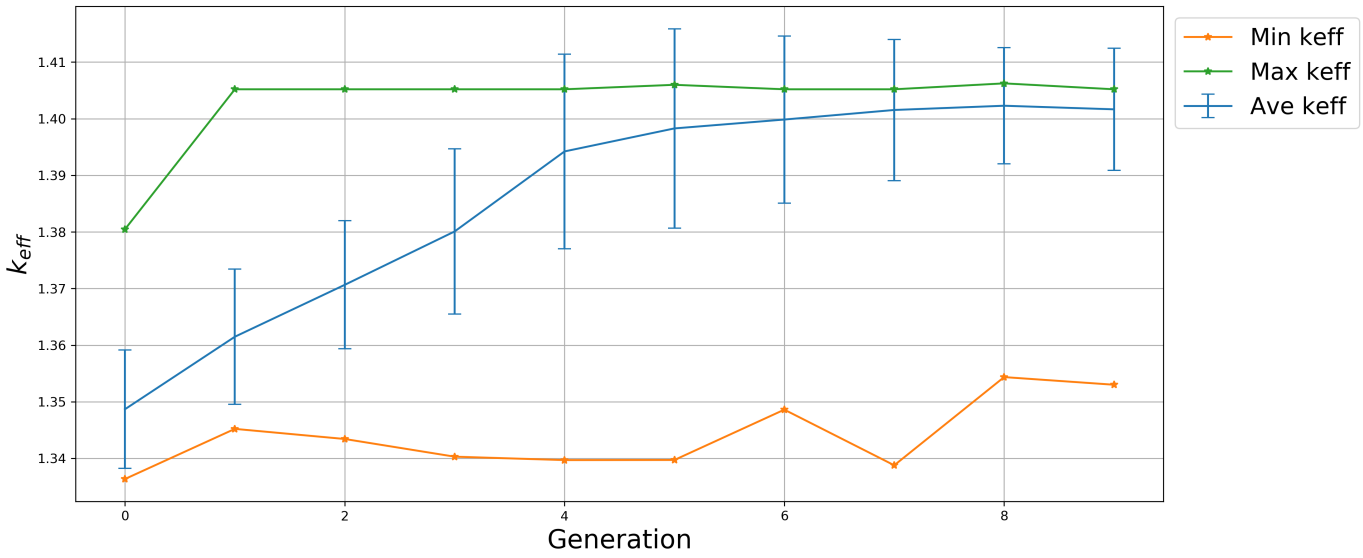
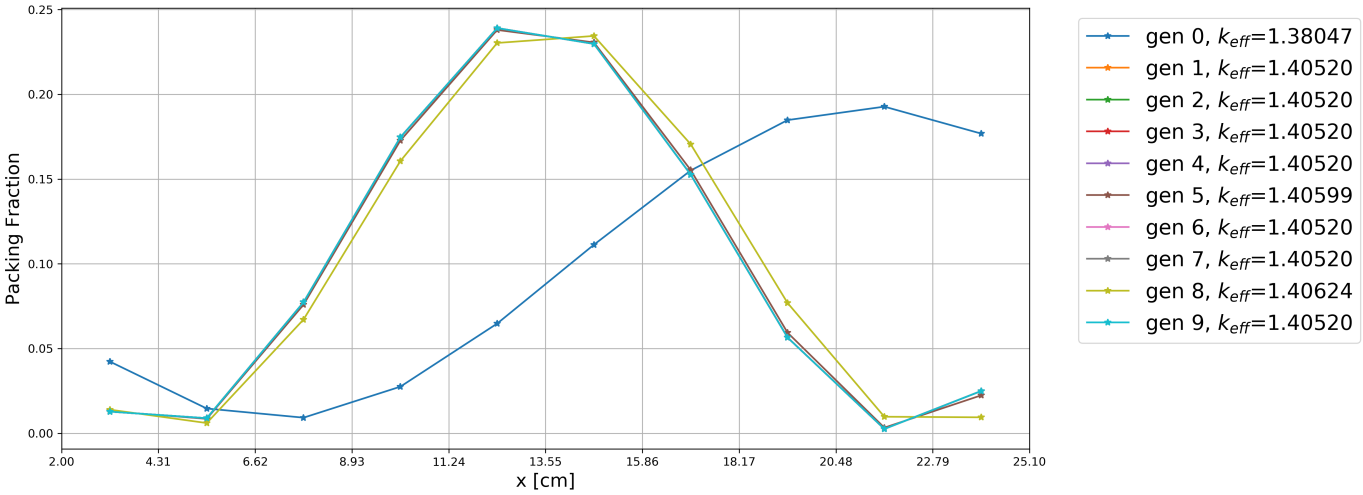


Figure 5.7: Experiment 39 packing distribution that produced $k_{effmax} = 1.40519$. Below: $PF(x) = (1.98\ sin(0.35x + 3.14) + 2) \times NF$ sine distribution with red points indicating the packing fraction at each slice. Above: Straightened Advanced High Temperature Reactor (AHTR) fuel slab with varying TRISO particle distribution across ten slices based on the sine distribution.



(a) Minimum, average, and maximum k_{eff} values evolution.



(b) k_{effmax}'s packing fraction distribution evolution.

Figure 5.8: Results for each generation for REALM's genetic algorithm optimization of the Straightened Advanced High Temperature Reactor (AHTR) Fuel Slab. The REALM simulation used the 39^{th} experiment's hyperparameter set.

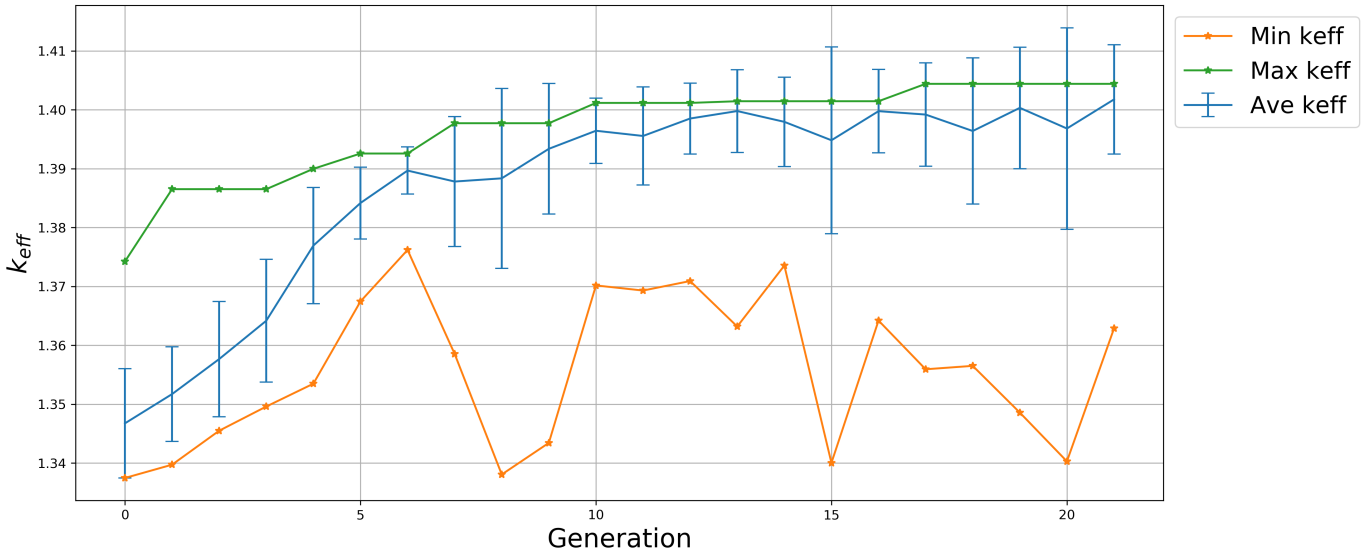
each generation.

Both Experiments 39 and 15 have packing fractions peaking at approximately 0.23 in the slab's center and decreasing to zero at the slab's sides. The amplitude, a, for the packing fraction distribution that produced k_{effmax} for Experiment 39 and the other top-five experiments (Table 5.3) have settled at the upper bound of approximately 2. A higher amplitude, a, shows that a slab geometry with larger packing fraction variations results in a higher k_{eff}. These observations about packing fraction distribution for k_{effmax} are consistent with conclusions from the FHR benchmark (Chapter 3): a high k_{eff} occurs with a good balance between fuel loading and moderation space. Fission occurs at high TRISO particle concentration areas at thermal flux; however, the neutrons are born at fast flux and require moderation to slow down to thermal ranges. Therefore, larger moderation areas ensure higher resonance escape probability for the fast neutrons resulting in higher thermal flux, leading to more fission occurring and a higher k_{eff}.

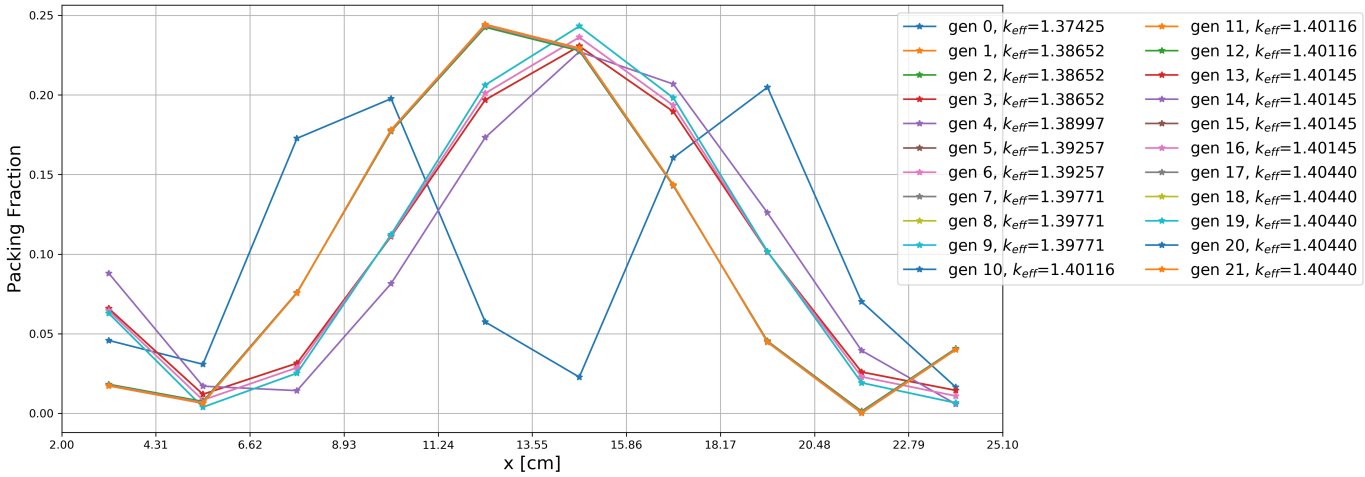
I also observed that TRISO particle packing fraction peaks in the center of the slab, proving that if the optimization problem focuses purely on the slab's neutronics by maximizing k_{eff}, the fuel tends to culminate in the middle. Center-peaking fuel density is nonideal for other key reactor core qualities, such as good heat transfer and ensuring flat power across the core.

5.2 AHTR Multiphysics Model Preliminary Work

I will use the open-source simulation tool, Moltres, to conduct AHTR multiphysics simulations. Moltres, an application built atop the MOOSE parallel finite element framework [27], contains physics kernels and boundary conditions to solve arbitrary-group deterministic neutron diffusion and thermal-hydraulics Partial Differential Equations (PDEs) simultaneously on a single mesh [49, 58]. AHTR Moltres simulations will capture thermal feedback effects, absent from the purely neutronics OpenMC simulations. The objective of setting



(a) Minimum, average, and maximum k_{eff} values evolution.



(b) k_{effmax}'s packing fraction distribution evolution.

Figure 5.9: Results for each generation for REALM's genetic algorithm optimization of the Straightened Advanced High Temperature Reactor (AHTR) Fuel Slab. The REALM simulation used the 15^{th} experiment's hyperparameter set.

up the Moltres AHTR simulation is to eventually couple Moltres with REALM for AHTR multiphysics optimization.

The benefits of Moltres over other multiphysics software, RELAP5 and NESTLE (used previously for AHTR modeling and described in Section 2.1.2), for coupled neutronics and thermal-hydraulics simulation:

- Moltres supports up to 3-D meshes, solving neutron diffusion and thermal-hydraulics PDEs simultaneously on the same mesh [58]. This is much more flexible than NESTLE and RELAP5, which only support rectangular and hexagonal assembly lattices. Therefore, Moltres can explore arbitrary reactor geometries easily.

- Moltres tightly couples neutronics and thermal-hydraulics, thus provides higher accuracy.

- Moltres, a MOOSE-based application, uses MPI for parallel computing, and compiles and runs on HPCs.

To run Moltres simulations, the user provides group constant data from a neutron transport solver, such as OpenMC, for Moltres' multigroup neutron diffusion calculations and a mesh file representing the reactor geometry. A TRISO-level fidelity mesh file is impractical and will result in an extremely long Moltres runtime. For successful AHTR Moltres simulation, I must establish suitable spatial and energy homogenization that preserves accuracy while maintaining an acceptable runtime.

5.2.1 Straightened AHTR Fuel Slab Multigroup Simulation

I use the continuous energy OpenMC simulation to generate multigroup cross section data defined over discretized energy groups and spatial segments. I then use OpenMC's multigroup calculation mode with the previously generated multigroup cross section data to calculate k_{eff}. Comparison of k_{eff} for the continuous and multigroup simulations determines if the energy and spatial homogenization used are acceptable.

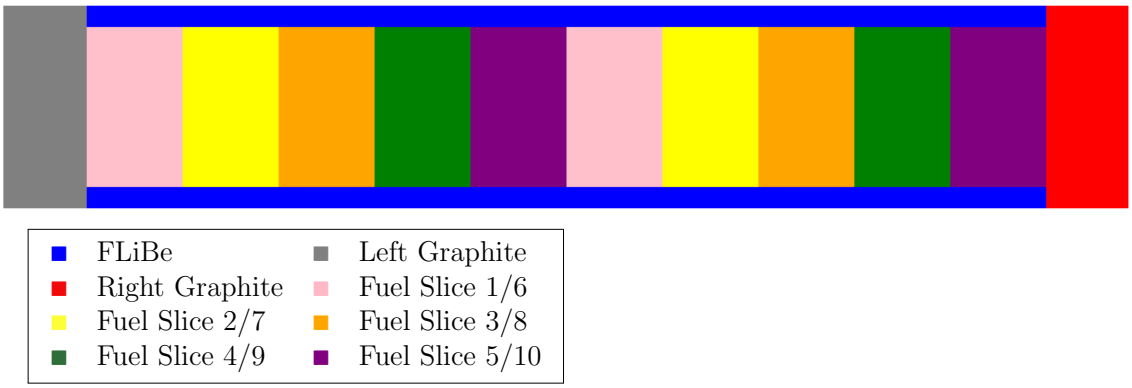


Figure 5.10: Straightened Advanced High Temperature Reactor (AHTR) fuel slab spatially discretized into 13 *cells* for OpenMC multigroup calculation.

Table 5.4: 4-group energy structures for Advanced High Temperature Reactor (AHTR) geometry derived by [28].

Group Boundaries [MeV]		
Group #	**Upper Bound**	**Lower Bound**
1	2.0000×10^{1}	9.1188×10^{-3}
2	9.1188×10^{-3}	2.9023×10^{-5}
3	2.9023×10^{-5}	1.8554×10^{-6}
4	1.8554×10^{-6}	1.0000×10^{-12}

In this section, the straightened AHTR fuel slab simulations use the TRISO particle distribution that generated k_{effmax} from the best hyperparameter set (Section 5.1.3). For spatial homogenization of the straightened AHTR fuel slab, I used OpenMC's *cell* domain type to compute multigroup cross sections for different *cells*. I discretized the slab into 13 *cells*: FLiBe, left graphite, right graphite, and ten fuel slices (each slice has a different packing fraction). Figure 5.10 illustrates the AHTR spatial homogenization for the OpenMC multigroup calculation. I used the four group energy structure derived by Gentry et al. [28] for AHTR geometries. Table 5.4 defines the group boundaries.

Table 5.5 shows the k_{eff} values from the continuous energy simulation and the spatial and energy homogenized simulation. The 26pcm difference between k_{eff} values is within both uncertainty values, assuring that the spatial and energy homogenization used is suitable for generating group constants for Moltres.

Table 5.5: Straightened Advanced High Temperature Reactor (AHTR) fuel slab's k_{eff} for case with continuous energy and space and case with spatial and energy homogenization. Both simulations were run on one BlueWaters XE Node, with 80 active cycles, 20 inactive cycles, and 8000 particles.

Homogenization	k_{eff}	**Simulation time [s]**
None	1.40473 ± 0.00115	808
Spatial and Energy	1.40499 ± 0.00109	50

5.3 Summary

This chapter demonstrated the preliminary work completed for AHTR optimization. I conducted a multigroup AHTR slab simulation with four-group energy and spatial homogenization, which resulted in k_{eff} within the uncertainty of the continuous energy simulation. The minimal k_{eff} difference assures that I can use these homogenizations when generating group constants for Moltres. I also successfully applied REALM to maximize k_{eff} in a straightened Advanced High Temperature Reactor (AHTR) fuel slab by varying the TRISO particle packing fraction distribution. The optimization process began with a coarse-to-fine random sampling hyperparameter search to find the genetic algorithm hyperparameters that worked best. Experiment 39 performed the best with a hyperparameter set that produced the highest final generation k_{effave} of 1.40165. The TRISO particle packing fraction distribution that produced the final generation's maximum k_{eff} of 1.40519 peaks at the slab's center with packing fraction distribution: $PF(x) = 1.989\ sin(0.54x + 3.143)$. This demonstration problem had a single objective function of maximizing k_{eff}. However, many other objectives should be considered, such as maximizing heat transfer and minimizing power peaking in the core. Thus, in the next chapter, I propose future simulations for optimizing these objective functions simultaneously.

Chapter 6

Future Work and Proposed Simulations

I demonstrated the need for this work with a summary in Chapter 2 of how additive manufacturing of nuclear reactor core components frees complex reactor geometries from traditional manufacturing constraints and enables reactor designers to reexamine reactor core design optimization. The literature review (Chapter 2) also concluded that stochastic evolutionary algorithm optimization methods could find global optimums for reactor design problems in the vast exploration design space enabled by additive manufacturing. Chapter 3 introduced the Fluoride-Salt-Cooled High-Temperature Reactor (FHR) benchmark with the AHTR design and highlighted its benefits, such as passive safety behavior with negative temperature coefficients. Chapter 4 introduced the Reactor Evolutionary Algorithm Optimizer (REALM) software package, which applies evolutionary algorithm optimization techniques to nuclear reactor design. In chapter 5, I successfully applied REALM to optimize the TRISO packing fraction distribution in an AHTR slab and demonstrated the neutron transport energy and spatial homogenizations for generating group constants for Moltres.

Based on the preliminary work conducted, this chapter proposes future simulations categorized into two groups: AHTR development and REALM optimization. The proposed work aims to address AHTR modeling challenges further and demonstrate using REALM for multi-objective AHTR optimization of arbitrary geometries and fuel distribution. For AHTR development, I propose the following simulations:

- AHTR 3D full core neutronics OpenMC simulation
- AHTR fuel slab and one-third fuel assembly multiphysics Moltres simulation

For REALM optimization, I propose the following REALM simulations:

- AHTR slab geometry optimization to maximize k_{eff}, minimize power peaking, and maximize heat transfer by varying TRISO x-axis distribution and FLiBe channel shape using OpenMC
- AHTR one-third fuel assembly optimization to maximize k_{eff}, minimize power peaking, and maximize heat transfer by varying TRISO x-y axis distribution and FLiBe channel shape using OpenMC

6.1 AHTR Model Development

The FHR benchmark introduced in Chapter 3 is an ongoing NEA project to assess the modeling and simulation capabilities for the AHTR. Benchmark participants, including the UIUC team, contributed Phases I-A and I-B (2D assembly steady-state and depletion) so far. The upcoming phases consist of 3D neutronics models and multiphysics models. Thus, to support the FHR benchmark, the proposed work will complete the benchmark's Phase I-C. In preparation for the later multiphysics benchmark phases, the proposed work will utilize Moltres to model AHTR multiphysics.

6.1.1 FHR Benchmark Phase I-C

The FHR benchmark's Phase I-C extends the 2D assembly model from Phases I-A and I-B into a 3D assembly model. The benchmark organizers will release Phase I-C's detailed specifications and required results in June 2021.

6.1.2 AHTR Multiphysics Model

Setting up a Moltres multiphysics simulation requires the user to provide group constant data from a neutron transport solver, such as OpenMC. Moltres neutronics calculations use

the following group constants: [49, 58]:

Σ_g^f: macroscopic fission cross section in group g

Σ_g^r: macroscopic removal cross section in group g

$\Sigma_{g' \to g}^s$: macroscopic scattering cross section from group g' to g

D_g: diffusion coefficient of neutrons in group g

ϵ_g: average fission energy per fission by a neutron from group g

ν: average neutron yield per fission by a neutron from group g

$\frac{1}{v}$: inverse neutron speed in group g

λ_i: decay constant of delayed neutron precursor (DNP) group i

β_{eff}: effective delayed neutron fraction

A Python script from the Moltres Github repository [48] extracts group constants from the neutron transport solver's output files. The Python script currently enables group constant extraction from Serpent [45] and SCALE [10] output files. I used OpenMC to model the AHTR neutronics for the FHR benchmark; thus, I will add the capability to extract group constants from OpenMC output files to the Moltres Python group constants extraction script.

Section 5.2 demonstrated that the multigroup neutronics simulation with four-group energy and spatial homogenization of the AHTR fuel slab generated a k_{eff} within uncertainty of the continuous energy and space neutronics simulation. I will utilize these homogenizations to create group constants for the Moltres AHTR fuel slab simulation. I will then set up a mesh for the AHTR fuel slab, run a Moltres simulation, and verify Moltres' ability to reproduce the following key neutronics parameters:

- k_{eff} (effective multiplication factor)
- reactivity coefficients: β_{eff}, α_D (doppler coefficient), $\alpha_{T,FliBe}$ (FLiBe temperature co-

efficient), and α_M (moderator temperature coefficient)

- Neutron energy spectrum
- $\phi_1(\vec{x},\vec{y}), \phi_2(\vec{x},\vec{y}), and \phi_3(\vec{x},\vec{y})$ (neutron flux distribution in four coarse energy groups)

Once verified, I will run a steady-state Moltres multiphysics simulation to determine the maximum temperature in the fuel slab at steady-state.

With information gleaned from the Moltres AHTR fuel slab simulation, I will test out energy and spatial homogenization for generating group constants for a one-third AHTR fuel assembly model. Then, I will proceed to set up the one-third AHTR fuel assembly model simulation, verify its key neutronics parameters, and finally, run a steady-state Moltres simulation.

6.2 REALM Optimization

Section 5.1 concluded that the AHTR slab optimization problem should be further developed by considering other objectives such as maximizing heat transfer and minimizing power peaking in the core. In the proposed work, I will explore each objective separately and then together. Table 6.1 describes each objective and how I will quantify each objective. I will

Table 6.1: Reactor Evolutionary Algorithm Optimizer (REALM) optimization problem objectives with their quantification descriptions.

Objective	**Quantification**
Best neutronics	Maximize k_{eff}
Maximize heat transfer	Maximize ϕ_{total} in areas along FLiBe coolant
Minimize power peaking	Minimize $P_{high} - P_{low}$

vary the following slab parameters to meet the described problem objectives:

- TRISO particle packing fraction distribution
- FLiBe coolant channel shape

Table 6.2: Proposed Reactor Evolutionary Algorithm Optimizer (REALM) simulations for optimizing Advanced High Temperature Reactor (AHTR) fuel assembly. Simulations explore two geometries: straightened AHTR fuel slab and AHTR's diamond-shaped section, containing six fuel slabs.

Simulation	AHTR Geometry	Objectives	Varying Parameters
1	Single fuel slab	• Maximize k_{eff}	• TRISO distribution
2	Single fuel slab	• Maximize heat transfer	• TRISO distribution
3	Single fuel slab	• Minimize power peaking	• TRISO distribution
4	Single fuel slab	• Maximize k_{eff}	• FLiBe channel shape
5	Single fuel slab	• Maximize heat transfer	• FLiBe channel shape
6	Single fuel slab	• Minimize power peaking	• FLiBe channel shape
7	Single fuel slab	• Maximize k_{eff} • Maximize heat transfer • Minimize power peaking	• TRISO distribution
8	Single fuel slab	• Maximize k_{eff} • Maximize heat transfer • Minimize power peaking	• FLiBe channel shape
9	Single fuel slab	• Maximize k_{eff} • Maximize heat transfer • Minimize power peaking	• TRISO distribution • FLiBe channel shape
10	Diamond section with six fuel slabs	• Maximize k_{eff} • Maximize heat transfer • Minimize power peaking	• TRISO distribution
11	Diamond section with six fuel slabs	• Maximize k_{eff} • Maximize heat transfer • Minimize power peaking	• FLiBe channel shape
12	Diamond section with six fuel slabs	• Maximize k_{eff} • Maximize heat transfer • Minimize power peaking	• TRISO distribution • FLiBe channel shape

I will conduct these optimizations for the straightened AHTR fuel slab geometry (as seen in Figure 5.1) and for one diamond-shaped sector (as seen in Figure 3.2) with x-y axis periodic and z axis reflective boundary conditions. Table 6.2 outlines the proposed simulations' details. I will use the optimal hyperparameters derived in Section 5.1.2 for the proposed simulations. Ideally, a new hyperparameter search should be conducted for each simulation to find the best hyperparameter set for each unique problem; however, the computational expense for conducting 11 hyperparameter searches is impractical. I find it acceptable to use the same hyperparameter set because of the problems' similarity. Table 6.3 summarizes

Table 6.3: Hyperparameter values for the best hyperparameter set calculated in Section 5.1.2.

Hyperparameters	**Values**
Population size	60
Generations	10
Mutation probability	0.23
Mating probability	0.46
Selection operator	`selTournament`
Selection individuals	15
Selection tournament size	5
Mutation operator	`mutPolynomialBounded`
Mating operator	cxBlend

the optimal hyperparameters.

I will extend the REALM simulations proposed in Table 6.2 to include Moltres evaluations if the proposed AHTR multiphysics Moltres simulations (Section 6.1.2) find approximations and assumptions that maintain accuracy while keeping acceptable Moltres runtimes.

6.3 Conclusion

Breakthroughs in metal component additive manufacturing fabrication have expedited the development of methods for nuclear reactor component additive manufacturing. The promise of cheaper and faster manufacturing of reactor components with additive manufacturing frees complex reactor geometries from previous manufacturing constraints and allows reactor designers to reexamine reactor design optimization. Therefore, I propose to explore the vast design space enabled by additive manufacturing with the evolutionary algorithm optimization technique that works well to find global optimums in multi-objective design problems, such as nuclear reactor optimization.

In the preliminary work, I designed the REALM Python package that applies evolutionary algorithm optimization techniques to nuclear reactor design using the DEAP Python module, OpenMC, and Moltres. REALM seeks to enable reactor designers to utilize robust

evolutionary algorithm optimization methods without going through the cumbersome process of setting up a genetic algorithm framework. With the many AHTRs benefits, I chose to apply the evolutionary algorithm optimization methods to this reactor type. I participated in Phase I-A and I-B of the Organisation for Economic Co-operation and Development (OECD) NEA's FHR benchmarking exercise. I also applied REALM to a single objective function problem: maximize k_{eff} in the AHTR fuel slab by varying the TRISO particle packing fraction distribution. This problem demonstrated the effectiveness and robustness of genetic algorithms at optimizing reactor parameters for an objective function. However, many other objectives should also be considered, such as maximizing heat transfer and minimizing power peaking in the core.

Therefore, I propose to further explore using REALM for multi-objective AHTR optimization of arbitrary geometries and fuel distribution. Optimization objectives include maximizing k_{eff}, maximizing heat transfer, and maximizing power peaking. I also propose to further address AHTR modeling challenges by completing the FHR benchmark's Phase I-C and to set up Moltres simulations to model AHTR multiphysics.

References

[1] Boeing: 3D printing done right. URL: https://www.boeing.com/features/innovation-quarterly/nov2017/feature-thought-leadership-3d-printing.page.

[2] Fluoride Salt-Cooled High-Temperature Reactor (FHR) Benchmark. URL: https://www.oecd-nea.org/jcms/pl_20249/fluoride-salt-cooled-high-temperature-reactor-fhr-benchmark.

[3] Roadmap for Regulatory Acceptance of Advanced Manufacturing Methods in the Nuclear Energy Industry. NEI Report, Nuclear Energy Institute. URL: https://www.nrc.gov/docs/ML1913/ML19134A087.pdf.

[4] Printed titanium parts expected to save millions in Boeing Dreamliner costs. *Reuters*, April 2017. URL: https://www.reuters.com/article/us-norsk-boeing-idUSKBN17C264.

[5] *Climate Change and Nuclear Power 2018*. Non-serial Publications. INTERNATIONAL ATOMIC ENERGY AGENCY, Vienna, 2018. URL: https://www.iaea.org/publications/13395/climate-change-and-nuclear-power-2018.

[6] Transformation In 3D: How A Walnut-Sized Part Changed The Way GE Aviation Builds Jet Engines | GE News, 2018. URL: http://www.ge.com/news/reports/transformation-3d-walnut-sized-part-changed-way-ge-aviation-builds-jet-engines.

[7] C. Behar. Technology roadmap update for generation IV nuclear energy systems. In *OECD Nuclear Energy Agency for the Generation IV International Forum, accessed Jan*, volume 17, pages 2014–03, 2014. Issue: 2018.

[8] Andrew Bergeron and J. B. Crigger. Early progress on additive manufacturing of nuclear fuel materials. *Journal of Nuclear Materials*, 508:344–347, 2018. ISBN: 0022-3115 Publisher: Elsevier.

[9] Benjamin R. Betzler, David Chandler, David H. Cook, Eva E. Davidson, and Germina Ilas. Design optimization methods for high-performance research reactor core design. *Nuclear Engineering and Design*, 352:110167, 2019. ISBN: 0029-5493 Publisher: Elsevier.

[10] J. A. Bucholz. SCALE: A modular code system for performing standardized computer analyses for licensing evaluation. NUREG NUREG/CR-0200-Vol.2-Bk.2; ORNL/NUREG/CSD-2-Vol.2-Bk.2 ON: DE82013370; TRN: 82-013156, Oak Ridge National Lab., TN (USA), 1982. URL: http://www.osti.gov/scitech/biblio/5360496.

[11] Jonathan Byrne, Philip Cardiff, Anthony Brabazon, and Michael ONeill. Evolving parametric aircraft models for design exploration and optimisation. *Neurocomputing*, 142:39–47, October 2014. URL: https://linkinghub.elsevier.com/retrieve/pii/S092523121400530X, doi:10.1016/j.neucom.2014.04.004.

[12] M. B. Chadwick, M. Herman, P. Obloinsk, M. E. Dunn, Y. Danon, A. C. Kahler, D. L. Smith, B. Pritychenko, G. Arbanas, R. Arcilla, R. Brewer, D. A. Brown, R. Capote, A. D. Carlson, Y. S. Cho, H. Derrien, K. Guber, G. M. Hale, S. Hoblit, S. Holloway, T. D. Johnson, T. Kawano, B. C. Kiedrowski, H. Kim, S. Kunieda, N. M. Larson, L. Leal, J. P. Lestone, R. C. Little, E. A. McCutchan, R. E. MacFarlane, M. MacInnes, C. M. Mattoon, R. D. McKnight, S. F. Mughabghab, G. P. A. Nobre, G. Palmiotti, A. Palumbo, M. T. Pigni, V. G. Pronyaev, R. O. Sayer, A. A. Sonzogni, N. C. Summers, P. Talou, I. J. Thompson, A. Trkov, R. L. Vogt, S. C. van der Marck, A. Wallner, M. C. White, D. Wiarda, and P. G. Young. ENDF/B-VII.1 Nuclear Data for Science and Technology: Cross Sections, Covariances, Fission Product Yields and Decay Data. *Nuclear Data Sheets*, 112(12):2887–2996, December 2011. doi:10.1016/j.nds.2011.11.002.

[13] Gwendolyn Chee. arfc/fhr-benchmark, 2021.

[14] Gwendolyn Chee. arfc/realm, 2021.

[15] Anselmo T. Cisneros and Dan Ilas. Neutronics and Depletion Methods for Parametric Studies of Fluoride Salt Cooled High Temperature Reactors with Slab Fuel Geometry and Multi-Batch Fuel Management Schemes. 2012.

[16] Lisandro Dalcn, Rodrigo Paz, Mario Storti, and Jorge DEla. MPI for Python: Performance improvements and MPI-2 extensions. *Journal of Parallel and Distributed Computing*, 68(5):655–662, 2008. ISBN: 0743-7315 Publisher: Elsevier.

[17] David L. Poole. *Computational intelligence*. Oxford University Press, 1998. URL: http://archive.org/details/computationalint00pool.

[18] Kalyanmoy Deb. *Multi-objective optimization using evolutionary algorithms*, volume 16. John Wiley & Sons, 2001.

[19] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002. ISBN: 1089-778X Publisher: IEEE.

[20] M. I. T. Facilitators, UW Madison Facilitators, and UC Berkeley Facilitators. Fluoride-Salt-Cooled, High-Temperature Reactor (FHR) Development Roadmap and Test Reactor Performance Requirements White Paper. 2013. URL: http://fhr.nuc.berkeley.edu/wp-content/uploads/2013/08/12-004-FHR-Workshop-4-Report-Final.pdf.

[21] C. D. Fletcher and R. R. Schultz. RELAP5/MOD3 code manual. Technical report, Nuclear Regulatory Commission, 1992.

[22] C. W. Forsberg, K. A. Terrani, L. L. Snead, and Y. Katoh. Fluoride-Salt-Cooled High-Temperature Reactor (FHR) with Silicon-Carbide-Matrix Coated-Particle Fuel. In *Transactions of the American Nuclear Society*, volume 107, page 907, 2012. URL: http://info.ornl.gov/sites/publications/files/Pub37875.pdf.

[23] Charles W. Forsberg, Per F. Peterson, and Paul S. Pickard. Molten-salt-cooled advanced high-temperature reactor for production of hydrogen and electricity. *Nuclear Technology*, 144(3):289–302, 2003. ISBN: 0029-5450 Publisher: Taylor & Francis.

[24] Flix-Antoine Fortin, Franois-Michel De Rainville, Marc-Andr Gardner, Marc Parizeau, and Christian Gagn. DEAP: Evolutionary algorithms made easy. *Journal of Machine Learning Research*, 13(Jul):2171–2175, 2012.

[25] Christian Gagn and Marc Parizeau. Open BEAGLE: A New Versatile C++ Framework for Evolutionary Computation. In *GECCO Late Breaking Papers*, pages 161–168. Citeseer, 2002.

[26] Aaron Garrett. inspyred: Bio-inspired Algorithms in Python. *URL: https://pypi. python. org/pypi/inspyred (visited on 11/28/2016)*, 2014.

[27] Derek Gaston, Chris Newman, Glen Hansen, and Damien Lebrun-Grandie. MOOSE: A parallel computational framework for coupled systems of nonlinear equations. *Nuclear Engineering and Design*, 239(10):1768–1778, October 2009. URL: http://www.sciencedirect.com/science/article/pii/S0029549309002635, doi:10.1016/j.nucengdes.2009.05.021.

[28] Cole Andrew Gentry. *Development of a Reactor Physics Analysis Procedure for the Plank-Based and Liquid Salt-Cooled Advanced High Temperature Reactor*. PhD thesis, 2016.

[29] Ian Gibson, David W. Rosen, and Brent Stucker. *Additive manufacturing technologies*, volume 17. Springer, 2014.

[30] GIF. A technology roadmap for generation IV nuclear energy systems. Technical Report GIF-002-00, US DOE Nuclear Energy Research Advisory Committee and the Generation IV International Forum, 2002.

[31] David E. Goldberg. Genetic algorithms in search. *Optimization, and MachineLearning*, 1989. Publisher: Addison Wesley Publishing Co. Inc.

[32] David E. Goldberg, Kalyanmoy Deb, and Dirk Thierens. Toward a better understanding of mixing in genetic algorithms. *Journal of the Society of Instrument and Control Engineers*, 32(1):10–16, 1993. ISBN: 0453-4662 Publisher: The Society of Instrument and Control Engineers.

[33] Sherrell R. Greene, Jess C. Gehin, David Eugene Holcomb, Juan J. Carbajo, Dan Ilas, Anselmo T. Cisneros, Venugopal Koikal Varma, William R. Corwin, Dane F. Wilson, and Graydon L. Yoder Jr. Pre-conceptual design of a fluoride-salt-cooled small modular advanced high-temperature reactor (SmAHTR). *Oak Ridge National Laboratory, ORNL/TM-2010/199*, 2010.

[34] M. K. M. Ho, G. H. Yeoh, and G. Braoudakis. Molten salt reactors. In A. Mndez-Vilas, editor, *Materials and processes for energy: communicating current research and technological developments*, number 1 in Energy Book Series, pages 761–768. Formatex Research Center, Badajoz, Spain, 2013 edition, 2013. http://www.formatex.info/energymaterialsbook/ http://www.energymaterialsbook.org/chapters.html.

[35] David E. Holcomb, George F. Flanagan, Gary T. Mays, W. David Pointer, Kevin R. Robb, and Graydon L. Yoder Jr. Fluoride salt-cooled high-temperature reactor technology development and demonstration roadmap. *ORNL/TM-2013/401, ORNL, Oak Ridge, TN*, 2013.

[36] David Eugene Holcomb, Dan Ilas, Venugopal Koikal Varma, Anselmo T. Cisneros, Ryan P. Kelly, and Jess C. Gehin. Core and refueling design studies for the advanced high temperature reactor. *ORNL/TM-2011/365, Oak Ridge National Laboratory, Oak Ridge, TN*, 2011.

[37] Sarah Kamalpour and Hossein Khalafi. SMART reactor core design optimization based on FCM fuel. *Nuclear Engineering and Design*, page 110970, 2020. ISBN: 0029-5493 Publisher: Elsevier.

[38] K. Koebke. A new approach to homogenization and group condensation. Technical report, 1980.

[39] Takaaki Koyanagi, Kurt Terrani, Shay Harrison, Jian Liu, and Yutai Katoh. Additive manufacturing of silicon carbide for nuclear applications. *Journal of Nuclear Materials*, 543:152577, 2020. ISBN: 0022-3115 Publisher: Elsevier.

[40] Sivam Krish. A practical generative design method. *Computer-Aided Design*, 43(1):88–100, January 2011. URL: `https://linkinghub.elsevier.com/retrieve/pii/S0010448510001764`, `doi:10.1016/j.cad.2010.09.009`.

[41] David J. Kropaczek and Ryan Walden. Constraint annealing method for solution of multiconstrained nuclear fuel cycle optimization problems. *Nuclear Science and Engineering*, 193(5):506–522, 2019. ISBN: 0029-5639 Publisher: Taylor & Francis.

[42] David J. Kropaczek and Ryan Walden. Large-Scale Application of the Constraint Annealing Method for Pressurized Water Reactor Core Design Optimization. *Nuclear Science and Engineering*, 193(5):523–536, 2019. ISBN: 0029-5639 Publisher: Taylor & Francis.

[43] David L. Krumwiede, Raluca O. Scarlat, Jae Keun Choi, Tung M. Phan, and Per F. Peterson. Three-dimensional modeling of the pebble-bed fluoride-salt-cooled, high-temperature reactor (PB-FHR) commercial plant design. In *Proceedings of the American Nuclear Society 2013 Winter Meeting*, 2013.

[44] Akansha Kumar and Pavel V. Tsvetkov. A new approach to nuclear reactor design optimization using genetic algorithms and regression analysis. *Annals of Nuclear Energy*, 85:27–35, November 2015. URL: https://linkinghub.elsevier.com/retrieve/pii/S0306454915002285, doi:10.1016/j.anucene.2015.04.028.

[45] 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. doi:10.1016/j.anucene.2014.08.024.

[46] Hsun-Chia Lin. *Thermal Hydraulics System-Level Code Validation and Transient Analyses for Fluoride Salt-Cooled High-Temperature Reactors*. PhD thesis, 2020.

[47] B. A. Lindley, J. G. Hosking, P. J. Smith, D. J. Powney, B. S. Tollit, T. D. Newton, R. Perry, T. C. Ware, and P. N. Smith. Current status of the reactor physics code WIMS and recent developments. *Annals of Nuclear Energy*, 102:148–157, 2017. ISBN: 0306-4549 Publisher: Elsevier.

[48] Alexander Lindsay. Moltres, software for simulating Molten Salt Reactors, 2017. https://github.com/arfc/moltres. URL: https://github.com/arfc/moltres.

[49] Alexander Lindsay, Gavin Ridley, Andrei Rykhlevskii, and Kathryn Huff. Introduction to Moltres: An application for simulation of Molten Salt Reactors. *Annals of Nuclear Energy*, 114:530–540, April 2018. URL: https://linkinghub.elsevier.com/retrieve/pii/S0306454917304760, doi:10.1016/j.anucene.2017.12.025.

[50] Markus List, Peter Ebert, and Felipe Albrecht. Ten Simple Rules for Developing Usable Software in Computational Biology. *PLOS Computational Biology*, 13(1):e1005265, January 2017. URL: https://dx.plos.org/10.1371/journal.pcbi.1005265, doi:10.1371/journal.pcbi.1005265.

[51] Limin Liu, Dalin Zhang, Qing Lu, Kunpeng Wang, and Suizheng Qiu. Preliminary neutronic and thermal-hydraulic analysis of a 2 MW Thorium-based Molten Salt Reactor with Solid Fuel. *Progress in Nuclear Energy*, 86:1–10, January 2016. URL: https://linkinghub.elsevier.com/retrieve/pii/S0149197015300779, doi:10.1016/j.pnucene.2015.09.011.

[52] Longwei Mei, Xiangzhou Cai, Dazhen Jiang, Jingen Chen, Yuhui Zhu, Yafen Liu, and Xiaohe Wang. The investigation of thermal neutron scattering data for molten salt Flibe. *Journal of Nuclear Science and Technology*, 50(7):682–688, 2013. ISBN: 0022-3131 Publisher: Taylor & Francis.

[53] Lawrence Murr. Frontiers of 3D Printing/Additive Manufacturing: from Human Organs to Aircraft Fabrication. *Journal of Materials Science & Technology*, 32(10):987–995, October 2016. URL: https://www.sciencedirect.com/science/article/pii/S1005030216301335, doi:10.1016/j.jmst.2016.08.011.

[54] NCSA. About Blue Waters: The National Center for Supercomputing Applications at the University of Illinois at Urbana-Champaign, 2017. http://www.ncsa.illinois.edu/. URL: http://www.ncsa.illinois.edu/.

[55] Andrew Ng, Kian Katanforoosh, and Younes Bensouda Mourri. Improving Deep Neural Networks: Hyperparameter Tuning, Regularization and Optimization, 2021. URL: https://www.coursera.org/learn/deep-neural-network.

[56] Faisal Y. Odeh and Won Sik Yang. Core design optimization and analysis of the Purdue Novel Modular Reactor (NMR-50). *Annals of Nuclear Energy*, 94:288–299, 2016. ISBN: 0306-4549 Publisher: Elsevier.

[57] James M. Osborne, Miguel O. Bernabeu, Maria Bruna, Ben Calderhead, Jonathan Cooper, Neil Dalchau, Sara-Jane Dunn, Alexander G. Fletcher, Robin Freeman, Derek Groen, Bernhard Knapp, Greg J. McInerny, Gary R. Mirams, Joe Pitt-Francis, Biswa Sengupta, David W. Wright, Christian A. Yates, David J. Gavaghan, Stephen Emmott, and Charlotte Deane. Ten Simple Rules for Effective Computational Research. *PLoS Computational Biology*, 10(3):e1003506, March 2014. URL: https://dx.plos.org/10.1371/journal.pcbi.1003506, doi:10.1371/journal.pcbi.1003506.

[58] Sun Myung Park. Advancement and Verification of Moltres for Molten Salt Reactor Safety Analysis. Master's thesis, University of Illinois at Urbana-Champaign, Urbana, IL, August 2020. URL: https://www.ideals.illinois.edu/handle/2142/108542.

[59] Cludio MNA Pereira and Celso MF Lapa. Coarse-grained parallel genetic algorithm applied to a nuclear reactor core design optimization problem. *Annals of Nuclear Energy*, 30(5):555–565, 2003. ISBN: 0306-4549 Publisher: Elsevier.

[60] Cludio M.N.A. Pereira and Wagner F. Sacco. A parallel genetic algorithm with niching technique applied to a nuclear reactor core design optimization problem. *Progress in Nuclear Energy*, 50(7):740–746, September 2008. URL: https://linkinghub.elsevier.com/retrieve/pii/S014919700800019X, doi:10.1016/j.pnucene.2007.12.007.

[61] Christian S. Perone. Pyevolve: a Python open-source framework for genetic algorithms. *Acm Sigevolution*, 4(1):12–20, 2009.

[62] Bojan Petrovic and Kyle M. Ramey. FHR/AHTR NEA Benchmark, October 2019. URL: http://montecarlo.vtt.fi/mtg/2019_Atlanta/Petrovic1.pdf.

[63] Bojan Petrovic, Kyle M. Ramey, Ian Hill, E. Losa, M. Elsawi, Z. Wu, C. Lu, J. Gonzalez, D. Novog, G. Chee, K. D. Huff, M. Margulis, N. Read, and Eugene Shwageraus. Preliminary results for the NEA FHR benchmark phase I-A and I-B (submitted). In

ANS M&C 2021 - The International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering, Raleigh, North Carolina, 2021. ANS.

[64] David Petti. The future of nuclear energy in a carbon-constrained world. *Massachusetts Institute of Technology Energy Initiative (MITEI)*, page 272, 2018.

[65] Farzad Rahnema, David Diamond, Dumitru Serghiuta, and Paul Burke. Phenomena, gaps, and issues for neutronics modeling and simulation of FHRs. *Annals of Nuclear Energy*, 123:172–179, January 2019. URL: http://www.sciencedirect.com/science/article/pii/S0306454918304572, doi:10.1016/j.anucene.2018.08.035.

[66] Farzad Rahnema, Bojan Petrovic, Christopher Edgar, Dingkang Zhang, Pietro Avigni, Michael Huang, and Stefano Terlizzi. The Current Status of the Tools for Modeling and Simulation of Advanced High Temperature Reactor Neutronics Analysis. Technical report, Georgia Institute of Technology, 2015.

[67] Kyle M. Ramey and Bojan Petrovic. Monte Carlo modeling and simulations of AHTR fuel assembly to support V&V of FHR core physics methods. *Annals of Nuclear Energy*, 118:272–282, August 2018. URL: https://linkinghub.elsevier.com/retrieve/pii/S0306454918301816, doi:10.1016/j.anucene.2018.04.003.

[68] Gbor Renner and Anik Ekrt. Genetic algorithms in computer aided design. *Computer-aided design*, 35(8):709–726, 2003.

[69] Paul K. Romano and Benoit Forget. The OpenMC Monte Carlo particle transport code. *Annals of Nuclear Energy*, 51:274–281, January 2013. URL: http://www.sciencedirect.com/science/article/pii/S0306454912003283, doi:10.1016/j.anucene.2012.06.040.

[70] Jhonathan Rosales, Isabella J. van Rooyen, and Clemente J. Parga. Characterizing surrogates to develop an additive manufacturing process for U3Si2 nuclear fuel. *Journal of Nuclear Materials*, 518:117–128, 2019. ISBN: 0022-3115 Publisher: Elsevier.

[71] Wagner F. Sacco, Hermes Alves Filho, Nlio Henderson, and Cassiano RE de Oliveira. A Metropolis algorithm combined with NelderMead Simplex applied to nuclear reactor core design. *Annals of Nuclear Energy*, 35(5):861–867, 2008. ISBN: 0306-4549 Publisher: Elsevier.

[72] Wagner F. Sacco and Cludio MNA Pereira. Two stochastic optimization algorithms applied to nuclear reactor core design. *Progress in Nuclear Energy*, 48(6):525–539, 2006. ISBN: 0149-1970 Publisher: Elsevier.

[73] Geir Kjetil Sandve, Anton Nekrutenko, James Taylor, and Eivind Hovig. Ten Simple Rules for Reproducible Computational Research. *PLoS Computational Biology*, 9(10):e1003285, October 2013. 00018. URL: http://dx.plos.org/10.1371/journal.pcbi.1003285, doi:10.1371/journal.pcbi.1003285.

[74] Cdric Sauder. Ceramic matrix composites: nuclear applications. *Ceramic matrix composites: materials, modeling and technology*, pages 609–646, 2014. Publisher: Wiley Online Library.

[75] Raluca O. Scarlat, Michael R. Laufer, Edward D. Blandford, Nicolas Zweibaum, David L. Krumwiede, Anselmo T. Cisneros, Charalampos Andreades, Charles W. Forsberg, Ehud Greenspan, Lin-Wen Hu, and Per F. Peterson. Design and licensing strategies for the fluoride-salt-cooled, high-temperature reactor (FHR) technology. *Progress in Nuclear Energy*, 77:406–420, November 2014. URL: `http://www.sciencedirect.com/science/article/pii/S0149197014001838`, `doi:10.1016/j.pnucene.2014.07.002`.

[76] Raluca O. Scarlat and Per F. Peterson. The current status of fluoride salt cooled high temperature reactor (FHR) technology and its overlap with HIF target chamber concepts. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 733:57–64, 2014. ISBN: 0168-9002 Publisher: Elsevier. URL: `http://www.sciencedirect.com/science/article/pii/S0168900213007055`, `doi:10.1016/j.nima.2013.05.094`.

[77] Herbert A. Simon. *The sciences of the artificial*. MIT press, 2019.

[78] Joseph Simpson, James Haley, Corson Cramer, Olivia Shafer, Amy Elliott, Bill Peter, Lonnie Love, and Ryan Dehoff. CONSIDERATIONS FOR APPLICATION OF ADDITIVE MANUFACTURING TO NUCLEAR REACTOR CORE COMPONENTS. page 43, May 2019.

[79] Lance L. Snead, Takashi Nozawa, Yutai Katoh, Thak-Sang Byun, Sosuke Kondo, and David A. Petti. Handbook of SiC properties for fuel performance modeling. *Journal of nuclear materials*, 371(1-3):329–377, 2007. ISBN: 0022-3115 Publisher: Elsevier.

[80] Vladimir Sobes, Briana Hiscox, Emilian Popov, Marco Delchini, Richard Archibald, Paul Laiu, Ben Betzler, and Kurt Terrani. ARTIFICIAL INTELLIGENCE DESIGN OF NUCLEAR SYSTEMS EMPOWERED BY ADVANCED MANUFACTURING. page 8, 2020.

[81] Niyanth Sridharan, Maxim N. Gussev, and Kevin G. Field. Performance of a ferritic/martensitic steel for nuclear reactor applications fabricated using additive manufacturing. *Journal of Nuclear Materials*, 521:45–55, 2019. ISBN: 0022-3115 Publisher: Elsevier.

[82] ASTM Standard. Standard terminology for additive manufacturing technologies. *ASTM International F2792-12a*, 2012.

[83] K. A. Terrani. Transformational Challenge Reactor demonstration program. Technical report, 2019. URL: `Retrievedfromhttp://tcr.ornl.gov.`

[84] Kurt A. Terrani, J. O. Kiggans, Chinthaka M. Silva, C. Shih, Yutai Katoh, and Lance Lewis Snead. Progress on matrix SiC processing and properties for fully ceramic

microencapsulated fuel form. *Journal of Nuclear Materials*, 457:9–17, 2015. ISBN: 0022-3115 Publisher: Elsevier.

[85] Mina Torabi, A. Lashkari, Seyed Farhad Masoudi, and Somayeh Bagheri. Neutronic analysis of control rod effect on safety parameters in Tehran Research Reactor. *Nuclear Engineering and Technology*, 50(7):1017–1023, 2018. ISBN: 1738-5733 Publisher: Elsevier.

[86] Michael P. Trammell, Brian C. Jolly, M. Dylan Richardson, Austin T. Schumacher, and Kurt A. Terrani. Advanced Nuclear Fuel Fabrication: Particle Fuel Concept for TCR. Technical report, 2019.

[87] P.J. Turinsky, R.M.K. Al-Chalabi, P. Engrand, H.N. Sarsour, F.X. Faure, and W. Guo. NESTLE: Few-group neutron diffusion equation solver utilizing the nodal expansion method for eigenvalue, adjoint, fixed-source steady-state and transient problems. Technical Report EGG-NRE–11406, 10191160, June 1994. URL: `http://www.osti.gov/servlets/purl/10191160-pnc6Jn/webviewable/`, `doi:10.2172/10191160`.

[88] Adrin Uriondo, Manuel Esperon-Miguez, and Suresh Perinpanayagam. The present and future of additive manufacturing in the aerospace sector: A review of important aspects. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 229(11):2132–2147, 2015. ISBN: 0954-4100 Publisher: SAGE Publications Sage UK: London, England.

[89] Venugopal Koikal Varma, David Eugene Holcomb, Fred J. Peretz, Eric Craig Bradley, Dan Ilas, A. L. Qualls, and Nathaniel M. Zaharia. AHTR mechanical, structural, and neutronic preconceptual design. Technical report, Oak Ridge National Lab.(ORNL), Oak Ridge, TN (United States), 2012.

[90] Dingkang Zhang and Farzad Rahnema. Integrated Approach to Fluoride High Temperature Reactor Technology and Licensing Challenges (FHR-IRP). Technical report, Georgia Inst. of Technology, Atlanta, GA (United States), 2019.

[91] Y. Zhu and A. I. Hawari. Thermal neutron scattering cross section of liquid FLiBe. *Progress in Nuclear Energy*, 101:468–475, 2017. ISBN: 0149-1970 Publisher: Elsevier.