PARAMETRIC OPTIMIZATION OF TRU DESTRUCTION RATES IN HTR CORES USING HYBRID METAHEURISTIC ALGORITHM AND REGRESSION METHODS

A Dissertation

by

TAKANORI KAJIHARA

Submitted to the Graduate and Professional School of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Pavel V. Tsvetkov
Committee Members,	Sean M. McDeavitt
	Karen Vierow Kirkland
	Erick Moreno-Centeno
Head of Department,	Michael Nastasi

May 2022

Major Subject: Nuclear Engineering

ABSTRACT

Nuclear waste is one of the significant problems for the sustainable use of nuclear energy. The Deep-Burn using High-Temperature Reactor (HTR) is one strategy to minimize nuclear waste. The nuclear waste or TRU incineration problem in HTR is one of the complicated optimization problems because of the broad design parameter space and constraints. Traditionally, nuclear reactor design optimization relied on the experts' experience and in-depth knowledge of each design step. It has inherent risks of the local optima. This paper proposes the multi-objective optimization framework consisting of surrogate modeling using regression algorithms and the metaheuristic search using a hybrid of genetic algorithm and simulated annealing. In our approach, multiple sets of design parameter inputs and objective value outputs from the Monte Carlo simulation code are evaluated using objective functions and constraints. In this step, correlation analysis and feasibility search based on the trust-region method are performed for better learning data sets to the surrogate model. Then, the Gaussian Process regression algorithm with suitable kernel and hyperparameters makes a surrogate model using the learning data. In the surrogate model, the optimal design parameters are searched through the iteration process of fitness evaluation, mating, average-bound crossover, wavelet mutation, and annealing. The fitness is evaluated by objective values processed with the weighted sum method based on the contribution to the objectives. If the error of optimal values and the surrogate model is below the threshold, the framework ends. If not, the additional sets of learning data are added based on the tentative optimal point, and a new surrogate model is built. The framework is applied to the TRU minimization problem using GT-HTR300 for single-batch core loading. Each TRU nuclides, initial loading of TRU and U-235 are set objectives, and 25.9% of TRU destruction rate and 58.7% of Pu-239 destruction rate with 3% of errors were achieved.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supported by a dissertation committee consisting of Professor Dr. Pavel V. Tsvetkov (Department of Nuclear Engineering), Dr. Karen Vierow Kirkland (Department of Nuclear Engineering), Dr. Sean M. McDeavitt (Department of Nuclear Engineering), and Dr. Erick Moreno-Centeno (Department of Industrial & Systems Engineering).

All other work conducted for the dissertation was completed by the student independently.

Funding Sources

Graduate study was supported by a fellowship from the Japan Student Services Organization and a research assistantship from the Department of Nuclear Engineering.

NOMENCLATURE

HTGR	High Temperature Gas Reactor
IAEA	International Atomic Energy Agency
EW	Exempt Waste
VSLW	Very Short-Lived Waste
VLLW	Very Low-Level Waste
LLW	Low-Level Waste
ILW	Intermediate-Level Waste
HLW	High-Level Waste
USNRC	United States Nuclear Regulatory Commission
DOE	The Department of Energy
TRU	Transuranics
LLFP	Long-lived Fission Products
LWR	Light Water Reactor
ANS	American Nuclear Society
MOX	Mixed OXide
VVER	Voda Voda Energo Reactor
RBMK	Reaktor Bolshoy Moshchnosti Kanalnyy
FR	Fast Reactor
AGR	Advanced Gas-cooled Reactor
THORP	Thermal Oxide Reprocessing Plant
SFR	Sodium-cooled Fast Spectrum Reactor
MA	Minor Actinoid

RBWR	Resource-renewable Boiling Water Reactor
VHTR	Very-High-Temperature Reactor
ADS	Accelerator Driven System
TRISO	Tristructural Isotropic
THTR	Thorium High-Temperature Reactor
HTTR	High-Temperature Test Reactor
HTR	High-Temperature Reactor
HEST	High-energy External Source Transmuter
GTHTR300	Gas Turbine High-Temperature Reactor 300
NP hard	Non-deterministic Polynomial-time hard
GPR	Gaussian Process Regression
RBF	Radial Basis Function
GA	Genetic Algorithm
SA	Simulated Annealing
RMSE	Relative Mean Squared Error
IRPhEP	International Reactor Physics Experiment Evaluation Project
CSV	Comma-Separated Values
BOC	Beginning of the Cycle
EOC	End of the Cycle
HALEU	High-Assay Low-Enriched Uranium

TABLE OF CONTENTS

Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

1. INTRODUCTION AND LITERATURE REVIEW

The nuclear reactor design requires adjusting many design parameters within the constraints of safety requirements and physical limitations. Traditionally, nuclear reactor design optimization relied on the experts' experience and in-depth knowledge of each design step. It has inherent risks of the local optima; especially there are nonlinearities between inputs and outputs. This chapter introduces the background of the High-Temperature Gas Reactor (HTGR) waste minimization problem as a complicated representative system of the multi-objective optimization problem. Also, the overview of the combinatorial optimization problems and multi-objective optimizations of reactor design is shown from the viewpoint of metaheuristics.

1.1 Nuclear Waste

Many countries are struggling with the disposal of nuclear waste[1]. Nuclear waste is produced from various nuclear facilities such as nuclear power plants, research reactors, and accelerators as a byproduct. The nuclear power plant is a predominant source of waste inventory among these facilities. Nuclear power is one of the reliable energy sources meeting the carbon-free policy, so that disposal strategy is essential to keep nuclear power.

1.1.1 Classification of Nuclear Waste

There are several classifications of nuclear waste. Its activity content and half-life usually classify nuclear waste. Fig. 1.1[2] shows the conceptual illustration of the waste classification decided by the International Atomic Energy Agency (IAEA). If the radioactivity is less enough to meet the criteria for clearance or exemption, it is defined as exempt waste (EW). EW is not treated as nuclear waste in many countries. It can be disposed of as general waste or recycled. If the half-life of nuclear waste is very short, it is defined as very short-lived waste (VSLW). After a few years of storage, this waste can be treated as regular waste. Depending on the activity level of nuclear waste, other wastes are classified as very low-level waste (VLLW), low-level waste (LLW), intermediate-level waste (ILW), and high-level waste (HLW). VLLW, LLW, and ILW are

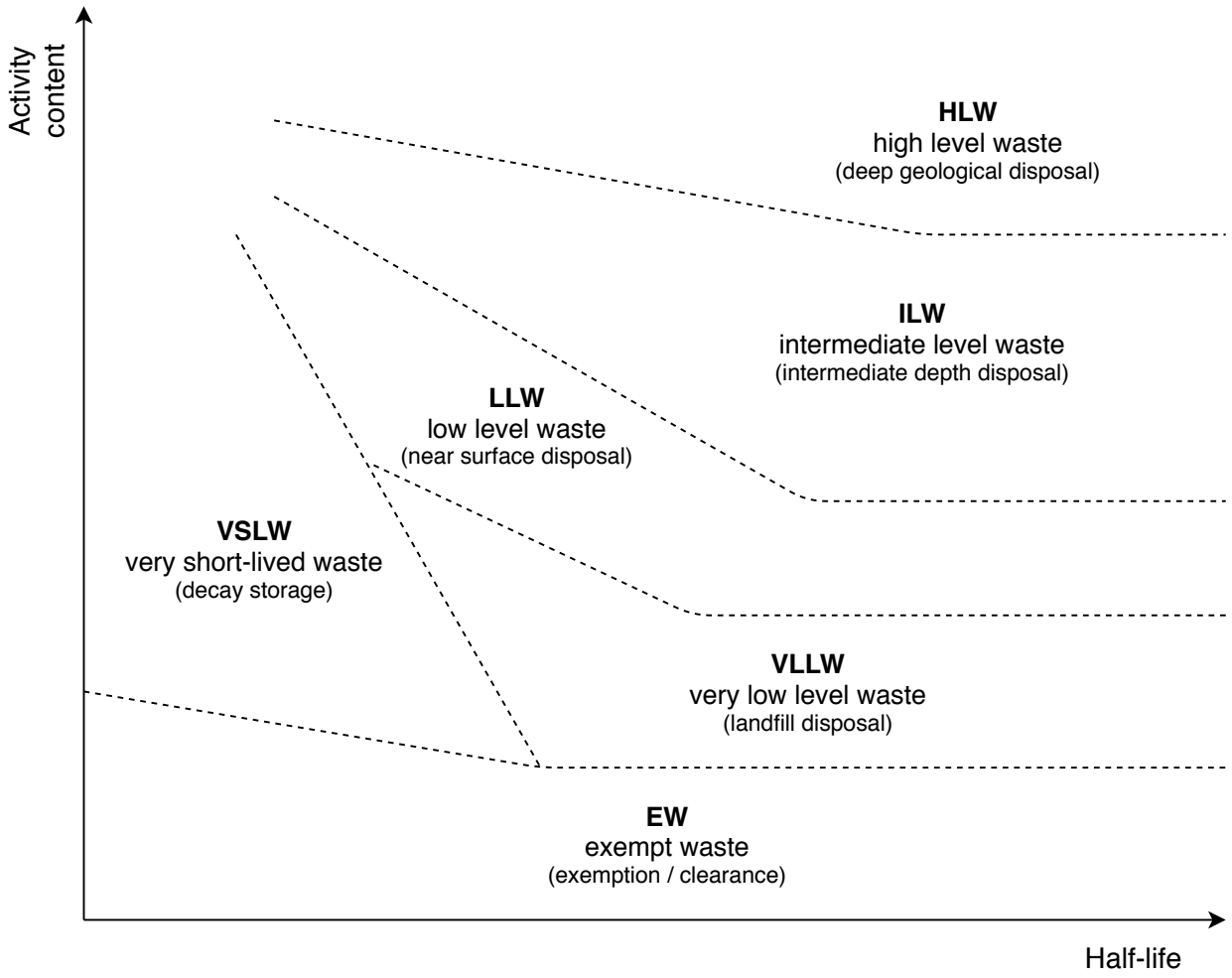


Figure 1.1: Conceptual illustration of the waste classification scheme

disposed to near-surface landfill type facility, engineered near-surface facilities, and intermediate-depth disposal facilities, respectively. If the activity level of the waste is high enough to generate significant quantities of heat and radiation, it is HLW, which requires disposal facilities in deep, stable geological formations more than several hundred meters below the surface.

United States Nuclear Regulatory Commission (USNRC) defines HLW as 'spent (used) reactor fuel when it is accepted for disposal or waste materials remaining after spent fuel is reprocessed[3]'. This definition is a subset of the IAEA definition, and this paper uses the USNRC definition if there is no additional explanation.

1.1.2 Current US policy to HLW

Since 1987, Yucca Mountain has been expected to be a repository for HLW. The Department of Energy (DOE) submitted a construction license application to the USNRC in 2008. However, the Obama Administration changed the project's direction and started to seek an alternative approach in a high-level Blue Ribbon Commission. In 2021, the energy secretary in the Biden Administration, Jennifer Granholm, confirmed opposition to the Yucca Mountain disposal and succeeded Obama Administration, which established consensus-based strategies to determine alternative waste disposal sites[4]. The continuous uncertainty of the HLW disposal is potentially harmful to the current backend of the nuclear fuel cycle. The optimization of nuclear waste and disposal systems is prominent, as mentioned in the ARPA-E Reauthorization Act of 2019[5].

1.1.3 Transuranics (TRU) and Long-lived Fission Products (LLFP)

When considering deep geological disposal, TRU and LLFP are the major nuclides in HLW. TRU is an artificially made, radioactive nuclide with a higher atomic number than uranium, such as neptunium, plutonium, americium, and curium. They are not in the fresh nuclear fuel composing uranium-235, uranium-238, and oxygen-16. Parts of these uranium nuclides transmute to TRU nuclides by absorption reaction and radioactive decay through the operation in the nuclear power plants. After the conventional Light Water Reactor (LWR) operation, typically, 1.1 wt% of fuel transmutes to the TRU[6]. On the other hand, LLFP is a fission product with a half-life of more than 200,000 years. There are only seven nuclides that can meet this criterion. These seven nuclides are technetium-99, tin-126, selenium-79, zirconium-93, cesium-135, palladium-107, and iodine-129. Technetium-99 has the shortest half-life: 211,000 years, and iodine-129 has the longest half-life: 15,700,000 years.

1.1.4 Spent Fuel Policy and Inventory Status in the World

Spent fuel is directly disposed to the disposal site (once-through fuel cycle), or it is reprocessed to recover fissile material such as U-235 and Pu-239 (closed fuel cycle). Currently, the USA and Germany adopt a direct once-through cycle, while France, China, Japan, and Russia adopt

reprocessing and recycling policies to some extent. The UK just ended the reprocessing in 2020.

Their detailed policy is as follows. In the USA, the last reprocessing plant stopped operation in 1972. Then, the reprocessing of spent fuel from commercial reactors has been prohibited since 1977 due to the proliferation concerns and cost estimation. However, recently, the DOE and the American Nuclear Society (ANS) have been interested in gradually restarting reprocessing efforts for advanced reactor technologies in the long term[7][8][9]. Germany reprocessed their spent fuel in the past on their experimental facility and the UK and French facilities. The total amount of reprocessed fuel is approximately 6,500 tons. However, their legislature prohibited the shipment of spent fuel for the reprocessing purpose in mid-2005. Therefore, the resulting spent fuel is planned for Mixed Oxide (MOX) fuel fabrication, while the remaining spent fuel will be disposed of directly as HLW [10].

France has been actively reprocessing spent fuel since the start of the La Hague plant in 1967. Their reprocessing ability is nearly 1,050 tons per year, approximately 91 % of annual spent fuel production in France. The reprocessed plutonium is mixed with uranium and refabricated to MOX fuel[11][12]. Japan has reprocessed their spent fuel by their experimental facility in Tokai and the UK and French commercial facilities. They are constructing a commercial reprocessing facility in Rokkasho based on the technology of La Hague in France but significantly delayed due to the vitrification unit problem and new safety standards followed by accident at the Fukushima Daiichi nuclear power plants[12][13]. China operated a pilot reprocessing plant in 2013. They are expecting to reprocess spent fuel from LWR. Their current commercial plant project with Orano started in 2020 and plan to complete by 2030[14]. Russia aims to recycle fissile material as a fundamental practice. Currently, they reprocess spent fuel from Voda Voda Energo Reactor (VVER)-440 and BN600 at the RT-1 spent fuel reprocessing facility in Ozersk. They are using reprocessed uranium for fabricating fuel for Reaktor Bolshoy Moshchnosti Kanalnyy (RBMK) and plutonium for MOX fuel for fast reactors (FRs). They are constructing a full-scale RT-2 facility by 2025[15].

The UK started the reprocessing spent fuel for their Magnox reactor in Windscale in 1956 and then continued in Sellafield from 1964. Then, they expanded their capability for Advanced

Gas-cooled Reactor (AGR) and LWR in 1994. They were reprocessing their own spent fuels, and many spent fuels from foreign countries for years but ceased their Thermal Oxide Reprocessing Plant (THORP) operation for AGR and LWR in 2018 due to less demand and aging. Their Magnox reprocessing plant will also stop in 2020, and there is no plan to build a new reprocessing plant[12][16].

According to the IAEA, approximately 250,000 tHM of spent fuel is stored, and 120,000 tHM of reprocessed spent fuel is stored worldwide[1]. The inventory of spent fuel and reprocessed fuel in each area except for India and Pakistan is shown in Table 1.1[1]. Approximately one-third of

Table 1.1: Inventory of Spent Fuel in the World(tHM).

	Wet storage	Dry storage	Reprocessed	Total
Africa	850	50	n/a	900
Eastern Europe	28,600	2,700	3,200	40,000
Western Europe	37,000	4,600	108,400	154,100
Far East	32,100	5,700	8,600	46400
North America	79,300	41,900	n/a	121,200
Latin America	3,000	2,000	n/a	5,000
Global Total	180,800	56,900	120,300	367,600

fuel has been reprocessed, and 75% of the unreprocessed spent fuel is still in the spent fuel pool in the nuclear power plant sites.

The acceptance of a disposal site for nuclear waste is essential. Although a few countries, such as the case of Finland in 2015, have already determined to construct a disposal site in the specific site, the process of disposal site selection is arduous in many countries. The site selection's difficulty is partly due to the long half-life of radioactive materials represented by the TRU and LLFP. The radioactive toxicity of some nuclides remains even after 100,000 years, making a fear for the residents in the candidate site. Even if the politics of nuclear waste disposal make the issues more complicated beyond the scientific view, decreasing the amount of waste and radioactive toxicity will contribute the easier site selection.

1.2 Waste Minimization

1.2.1 Waste Minimization Overview

Waste minimization strategies and techniques have been proposed to resolve such a complicated situation in the world. Many scenarios have been put forward with different kinds of reactor types. A Sodium-cooled Fast spectrum Reactor (SFR) is planned to use for burning TRU and Minor Actinoids (MAs) by taking advantage of the high transmutation ability of fast-spectrum neutrons[17][18]. Also, a new LWR has been proposed to burn TRU for providing an alternative option to the SFR, such as the Resource-renewable Boiling Water Reactor (RBWR)[19]. Another option is a Very High-Temperature Reactor (VHTR) or HTGR[20], utilizing thermalized neutron and high burnup. A concept beyond the conventional nuclear reactor is Accelerator Driven System (ADS), where neutrons generated by the installed accelerator are directly emitted to a target assembly containing TRU with fissile materials. This system efficiently induces neutron captures and fission of TRU, but there are many technical challenges for both accelerator and reactor. In 2011, the US DOE evaluated and screened nuclear fuel cycle options[21]. DOE emphasized the reduction of nuclear waste through the fuel cycle in these scenarios. Both fast spectrum reactors and thermal reactors are selected for the candidates of fuel cycle options.

Fast spectrum reactor has the most robust capability to destroy TRU entirely because it has favorable fission to capture ratio for target nuclides. However, relatively small cross-sections of TRU nuclides in a fast spectrum region results in a slower depletion process so that multiple cycles would be necessary to achieve a complete burnup of TRU.

Even though the HTGR option is not favorable to destroy TRU entirely, a relatively large cross-section of fissile TRU nuclides in the target spectrum and higher burnup achieved by the Tristructural isotropic (TRISO) fuel make this option feasible for the near-term solution. It can achieve high transuranic destruction efficiency. HTGR technology was demonstrated in the US, UK, and Germany in the past. Also, Japan and China are currently operating experimental or demonstration types of HTGRs. So commercial HTGRs are expected to be deployed shortly. Also,

a preliminary study showed the advantage of using HTGR in combining SFR for TRU destruction purposes [22]. In this two-tier scenario, the TRU from LWR is recycled and burned in an HTGR as the 1st-tier. Then, TRU in spent fuel of HTGR is recycled again to be burned in an SFR as the 2nd-tier. It was shown that the high destruction efficiency of HTGR can reduce the number of SFR for TRU destruction compared with other recycling scenarios such as MOX+LWR. Considering TRU destruction efficiency and possible near-term deployment, this study selected HTGR as the favorable near-term option for TRU burn.

1.2.2 HTGR

HTGR is a thermal-spectrum reactor that can produce higher temperatures than conventional reactors, and helium is used as a coolant. Inert and low-density helium, the high heat capacity of graphite core, and the high stability of TRISO fuel realize an inherent safety reactor system. Historically, Dragon in the UK, AVR and Thorium High-Temperature Reactor (THTR) in Germany, and Peach Bottom and Fort St. Vrain in the US were successfully operated. Nowadays, High-Temperature Test Reactor (HTTR) in Japan and High-Temperature Reactor(HTR)-10 and HTR-PM in China are operating, and several companies in the US are pursuing commercial deployments such as X-Energy, Kairos Power, and Framatome.

1.2.3 Deep Burn

The Deep-Burn using HTGR is one strategy to minimize nuclear waste. HTGR uses TRISO fuel, made of tiny particles with a hundred-millimeter diameter core of oxide fuel kernel surrounded by multiple carbon and silicon carbide layers. This fuel can achieve high fuel burnup levels compared with the conventional fuel for LWR. Also, graphite, which has lower energy decrement compared with hydrogen, allows non-fissile nuclides to interact with resonance or epithermal neutrons, which contribute to destruct thermally non-fissile TRU nuclides[23].

Previous studies, VHTR loading nuclear waste, showed a high TRU destruction rate. In the case of the VHTR model loading spent fuel from AP1000, TRU destruction rates reached approximately 30%. Furthermore, the destruction rate of Pu-239 was almost 54.3% [24]. This

research also realized higher incineration performance using High-energy External Source Transmuter (HEST)[25]. Another research realized a much higher destruction rate by fuel shuffling up to 63.7% of TRU destruction[26]. One research investigated the destruction capability for the fuel shuffling pattern[27][28]. They showed the TRU incineration characteristics for the multiple scenarios of shuffling. These studies above are based on prismatic core HTGRs with traditional block fuels. On the other hand, prismatic core with so-called Pin-in-block type fuels was also attempted to deploy deep-burn concept in Japan[29] and showed 95% of Pu-239 destruction. They also expand their target to TRU incineration[30]. Through the research effort shown above, the basic feasibility of a deep burn strategy was proven in the aspect of neutronics. However, optimization strategies rely on reactor physics knowledge, and the search space of reactor design parameters and fuel loading patterns is extensive. Therefore, there is room to explore the global optima of waste minimization using mathematical optimization.

1.3 Research Objectives

The objective of this research, as documented in this Ph.D. dissertation, is to develop and demonstrate a consistent optimization framework for reactor design optimization problems using a combination of regression algorithms and metaheuristics optimization approaches. The TRU minimization problems applied to HTRs are solved as a demonstration example showcasing applications of the developed and implemented optimization framework. The main objective will be considered through the following tasks to accomplish the above:

1. Developing an optimization framework consisting of an automatic input generator and output assimilation for Serpent2, a trust region-based two-phase iteration algorithm, a regression system using the Gaussian process regression method, and metaheuristics search algorithms using the hybrid metaheuristics of genetic algorithms and simulated annealing.

2. Modeling a high fidelity HTTR and Gas Turbine High-Temperature Reactor 300 (GTHTR300) cores based on a pin-in-block type fuel assembly design using Serpent 2.

3. Developing a set of optimization design parameters, objective functions, and weighed func-

tion for waste for the waste minimization problems.

4. Developing constraints for criticality and reactivity.

5. Analyzing the feasibility and performance of the developed methods for the TRU minimization problems.

1.4 Optimization

This section overviews the mathematical optimization and the representative algorithms used in the engineering design problems. Mathematical optimization is the process of finding solution sets for maximizing or minimizing the value of objective functions under the given constraints. Let *S* be feasible search space and objective function f such that:

$$f : S \mapsto \mathbb{R}. \tag{1.1}$$

Then, the general problem is to find a optimal solution x^* such that:

$$f(x^*) \ll f(x), \forall x \in S. \tag{1.2}$$

Furthermore, the optimal cost is denoted as

$$f^* = f(x^*) = \min_{x \in S} f(x), \tag{1.3}$$

and the set of optimal solutions are

$$x^* = \arg\min_{x \in S} f(x), \tag{1.4}$$

where x is a decision variable vector that determines the value of the objective function. In this equation, the optimization problem is minimization, but a modification for a maximization problem can be nearly always achievable. *S* is determined by given constraints. These problems with

constraints are called constrained optimization problems. Constraints are explicitly defined as equations or determined as the result of physical simulations. The simulation model can be the combination of multiple equations. However, many cases require more complex procedures, including stochastic methods.

The objective of the mathematical optimization problem is to find an optimal solution, infeasibility, unbounded, or non-optimal. Even if an optimal solution exists, sometimes it is difficult to find a globally optimal solution. The concept of neighborhood is introduced to explain the locally optimal solution. In this problem, the neighborhood is defined as

$$N(x) \subseteq S, \tag{1.5}$$

that are in close to x. Here, the locally optimal solution is defined as

$$f(x^*) \leq f(x), \forall x \in S \cap N(x^*). \tag{1.6}$$

The globally optimal solution is a locally optimal solution. In a complicated optimization problem, the goal is sometimes set to one of the locally optimal solutions that are good enough to approach the globally optimal solution in a reasonable time.

1.4.1 Combinatorial Optimization

The combinatorial optimization problem is a problem with a structure of combination in a feasible search space. In this problem, solutions are typically represented as a combination, permutation, graph, and integers. Traveling salesman problems, minimum spanning tree problems, and knapsack problems are examples of combinatorial optimization problems. Also, many engineering design problems are categorized as combinatorial optimization problems. Theoretically, all combinatorial optimization can be formulated as an integer optimization problem. In complicated combinatorial optimization problems, exhaustive search is not practical because of its NP-hard. Note that a problem P is called NP-hard if all problems in NP polynomially reduce to P. Also, even

a globally optimal solution is often difficult to find in a limited time.

The NP-hard combinatorial optimization problem can be solved in three different approaches of algorithms: exact algorithm, approximation algorithm, and a heuristic algorithm.

1.4.1.1 Exact Algorithm

The exact algorithm finds a globally optimal solution for a given problem. Even a Non-deterministic Polynomial-time hard (NP-hard) combinatorial optimization problem, some problems such as political districting problems require a globally optimized solution in nature. In this approach, the computation time of NP-hard is $O(C^n)$. So, significant effort to find an efficient algorithm for the specific problem is necessary.

1.4.1.2 Approximation Algorithm

The approximation algorithm finds a feasible solution with provable guarantees on the proximity of the solution to the globally optimal solution for a given problem. In most problems, the exact optimal solution is not required, so the approximation algorithm is selected to reduce computation time significantly. Many approximation algorithms need $O(n^C)$ computation time. Several algorithms are well known for the approximation algorithm, such as greedy algorithm, local search algorithm, and primal-dual methods.

1.4.1.3 Heuristic Algorithm

Heuristic algorithms find a feasible solution that does not guarantee the proximity of the globally optimal solution for a given problem. There are many algorithms and insights that can apply to many optimization problems except for a few counterexamples. Also, there are experimentally good algorithms but cannot guarantee the approximation due to the complex theoretical analysis. Heuristic algorithms are based on the accumulation of this knowledge and discoveries.

1.4.2 Multi-Objective Optimization

In a multi-objective optimization problem, more than one objectives to maximize or minimize are set. So instead of a single objective function, an objective vector $\mathbf{F}(x)$ is defined as the set of

objective functions $f(x)$ such that:

$$\mathbf{F}(x) = f_1(x), f_2(x), ..., f_k(x), \tag{1.7}$$

where each of $f_k(x)$ is a different objective function and k is the number of objective functions. So, the optimal solution is defined as

$$\mathbf{F}(x^*) = \min_{x \in S} \mathbf{F}(x), \tag{1.8}$$

This multi-objective optimization problem usually does not have a unique optimal solution because optimal solutions are different among each objective function. Several methods are proposed to evaluate each objective function and determine the optimal solution to a unique set of values.

1.4.2.1 Weighted Sum Method

The most straightforward approach is the weighted sum method:

$$U = \sum_{i=1}^{k} w_i f_i(x), \tag{1.9}$$

$$\sum_{i=1}^{k} w_i = 1, \tag{1.10}$$

where w is a weight vector. The relative value of the weights is the relative importance of each objective. Each objective function can be adjusted depending on the engineering judgment.

1.4.2.2 Weighted Global Criterion Method

For considering the Pareto optimal points, weighted global criterion method is introduced such that,

$$U = \left\{ \sum_{i-1}^{k} [w_i (f_i(x) - f_i^{\circ})]^p \right\}^{1/p}. \tag{1.11}$$

The solution of this formula depends on both w and p. Depending on how to set the p value, the behavior of this equation changes.

1.4.2.3 Lexicographic Method

Lexicographic method will prioritize the objective functions by ordering such that:

$$\begin{gathered} \min f_i(x) \\ s.t.\ f_j(x) \leq f_j(x_j^*);\ j = 1...(i-i);\ i > 1;\ i = 1...k, \end{gathered} \tag{1.12}$$

where i is a function's position after ordering by the preference, $f_j(x_j^*)$ is the minimum value for the jth objective function.

1.4.2.4 Bounded Objective Function Method

The bounded objective function method is the way to focus on a single objective function with limits on other objective functions such that

$$l_i \leq f_i(x) \leq \epsilon_i;\ i = 1...k;\ i\ \neq s, \tag{1.13}$$

where l_i and ϵ_i are the lower and upper bounds for objective function, respectively. This gives a weakly pareto optimal point and Pareto optimal solution.

1.4.3 Metaheuristics

Metaheuristics is a high-level procedure to find a heuristic combining several ideas based on basic search algorithms such as greedy and local search algorithms. These algorithms are widely adopted for combinatorial optimization problems, including a black box problem. Metaheuristic algorithms are inspired by various natural phenomena and mechanisms: a genetic algorithm is inspired by evolution theory, and simulated annealing is inspired by physical phenomena. There are two strategies to find better optimal value: intensification and diversification. Intensification is a way to search for similar combinations, which were evaluated as good in the past. In most combinatorial optimization problems, it is known that each good optimal point tends to have a similar structure. On the other hand, intensification fixes the search area in local optima and prevents searching for a broader area to find global optima. Hence, diversification is necessary.

Diversification is a strategy to search different structures of the past combinations. Metaheuristics iterate the two processes: (1) find a combination based on the record, (2) evaluate the combination and produce information for the next iteration. Each metaheuristic has different ideas for every two processes.

1.4.4 Genetic Algorithm

A genetic algorithm is a metaheuristic search algorithm inspired by famous Darwin's theory of species evolution[31]. Holland originally proposed the genetic algorithm in 1975,[32] and Goldberg popularized it in 1989[33]. This algorithm iterates the group of solutions through the process of fitting evaluation, crossover, and mutation. This process is similar to natural selection and species evolution. The genetic algorithm can also be categorized as a variant of stochastic beam search because successor states are reproduced by merging two-parent states instead of modifying a single state. Since it does not depend on the derivatives, the genetic algorithm can be applied to many engineering problems.

Genetic algorithms start with a set of randomly selected candidates. These candidates are called population. Some technical words of genetic algorithms are borrowed from the study of evolution. Also, each candidate is named a chromosome, and a single parameter in the candidate is named a gene. These chromosomes or candidates are selected or modified through the process of selection, crossover, and mutation. If the termination criteria meet the population after the mutation process, they will be the last population, and the process ends. If not, the population iterates through the process until the criteria meet. Table 1.2 shows the typical list of the evolution-inspired technical term of genetic algorithms. The genetic algorithm term is primarily used in this paper in the context of genetic algorithm. Fig. 1.2 illustrates the graphical definition of these terms.

At first, the initial population set is generated. Every single parameter x_i is a gene, and a series of genes, which is a possible candidate of the solution are chromosome X_j such that:

$$X = (x_1, x_2, ..., x_i, ..., x_N), \tag{1.14}$$

Table 1.2: Comparison between general term and genetic algorithm term.

General term	Genetic algorithm term
Decision variable	Gene (of chromosome)
Solution	Chromosome
Old solution	Parent
New solution	Children / Offspring
Fitness function	Quality of individual
Selection	Surviving parents

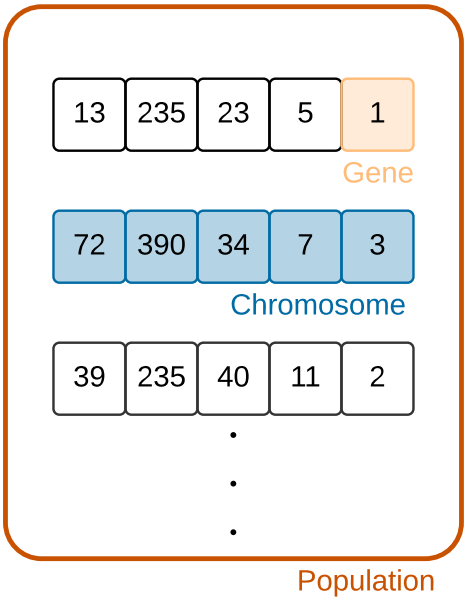


Figure 1.2: The graphical definition of gene, chromosome, and generation.

where N is the number of genes. Then, the population P is a group of chromosomes such that:

$$P = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_j \\ \vdots \\ X_M \end{bmatrix} = \begin{bmatrix} x_{1,1} & x_{1,1} & \cdots & x_{1,i} & \cdots & x_{1,N} \\ x_{2,1} & x_{1,1} & \cdots & x_{1,i} & \cdots & x_{2,N} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{j,1} & x_{1,1} & \cdots & x_{j,i} & \cdots & x_{j,N} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{M,1} & x_{1,1} & \cdots & x_{M,j} & \cdots & x_{M,N} \end{bmatrix}. \tag{1.15}$$

Here, there are M chromosomes generated as the first population, and these candidates will get through the multiple stages in each iteration.

1.4.4.1 Selection

Each chromosome is evaluated by the quality of individual or fitness function in the selection process. The chromosomes obtain a higher score in terms of the quality of individual can be the better candidate. Based on the quality of individual evaluation, some chromosomes are selected as parents for the mating process or as the next generation. This selection is processed deterministically or stochastically by the probability given by the result of the quality of individual. Some representative selection methods are introduced here. Proportionate selection[33] evaluates each chromosome as follows:

$$P_k = \frac{F(X_k)}{\sum_{j=1}^{M} F(X_j)}, \tag{1.16}$$

where P_k is the probability of the kth chromosome being selected, $F(X)$ is the fitness function of a chromosome. The higher the fitness function value is, the more often selected for the parent in the next process. This selection method still has a chance to select less favorable chromosomes in low probability, contributing to diversification.

Another famous selection method is ranking selection. This method ranks all chromosomes based on the fitness function value at first. Then, ranked chromosomes are assigned a probability based on their rank with the following equations:

$$P_k = U - (S_k - 1) \times Z, \tag{1.17}$$

$$S_k = Rank(X_k), \tag{1.18}$$

$$\sum_{j=1}^{M} P_j = 1, \tag{1.19}$$

$$U = \frac{Z(M-1)}{2} + \frac{1}{M}, \tag{1.20}$$

where S_k is the rank of the kth chromosome, Z is an arbitrary value. A higher Z value gives

more chance to the higher rank chromosome. The last selection method is tournament selection. Tournament selection consists of two iterative processes. In the first process, the candidate of the parent is randomly selected with a uniform distribution. So, the first selection process does not consider the fitness function of chromosomes. Each chromosome can be equally selected stochastically. Then, the best chromosome in terms of the fitness function is selected from the chromosomes selected in the first process. Only one chromosome is selected as a parent for each second process. This iteration continues until all parents are selected. Each selection method determines how easily getting the convergence in the genetic algorithm iteration is and strengthens the diversification.

1.4.4.2 Crossover

In the crossover process, selected chromosomes will exchange parts of their parameters in some manners, and then exchanged chromosomes will be candidates as offspring. They can generate new chromosomes, which may achieve the more optimal solution. In most cases, some parents and all children are selected for the next generation. There are two major types of crossover: point crossover and uniform crossover. Fig. 1.3 illustrates how these crossover work. Point crossover randomly selects one or multiple points. For example, in the case of one-point crossover, if each chromosome has five genes, an offspring from the first pair will get the first three genes from the first parent chromosome and get the remaining two genes from the second parent. In contrast, the other offspring from the same pair will get the first three genes from the second parent and the remaining two genes from the first parent. In the case of two-point crossover, a single dotted line on the figure becomes two dotted lines and exchanges the genes accordingly. The uniform crossover chooses some pair of genes as shown in (B) of Fig. 1.3 instead of determining the exchange area by line(s). Also, the result of the one-point crossover is shown as:

$$X_1^{child} = (x1, x2, ..., x_c, x'_{c_1}, ..., x'_N), \tag{1.21}$$

$$X_2^{child} = (x'1, x'2, ..., x'_c, x_{c_1}, ..., x_N), \tag{1.22}$$

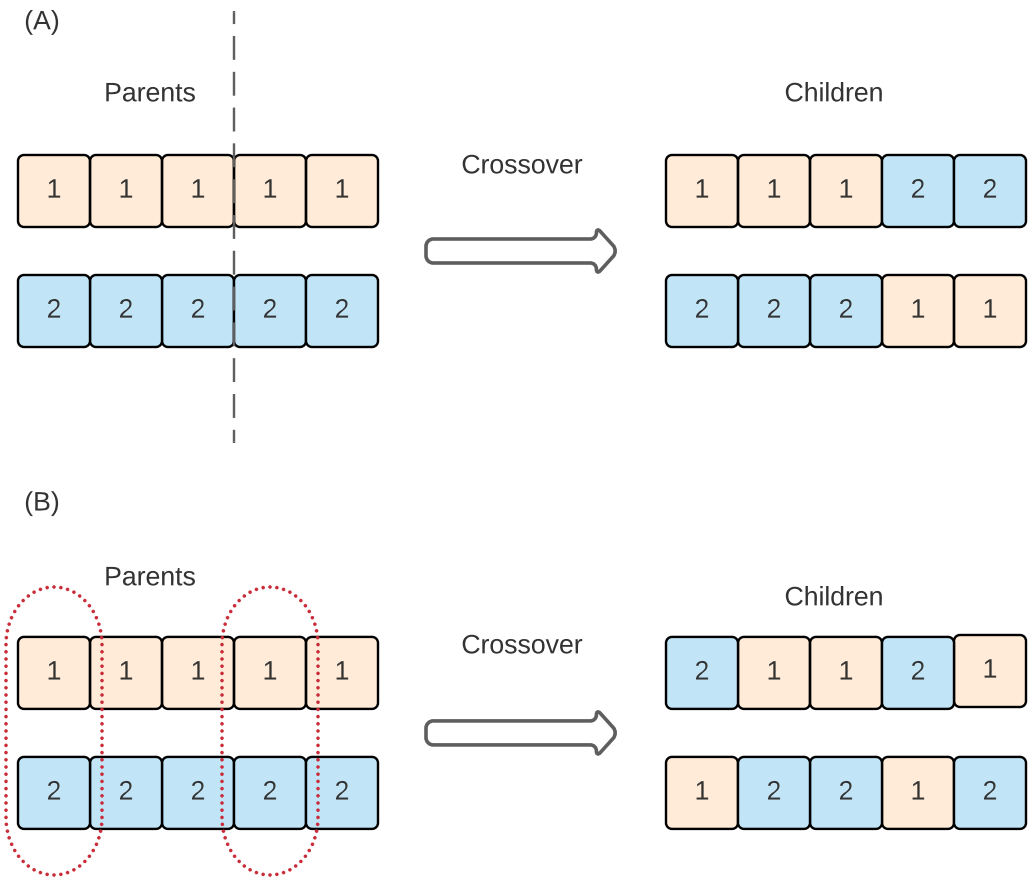


Figure 1.3: Examples of how crossover works: (A) point crossover and (B) uniform crossover.

where c is the crossover point and N is number of genes in parent chromosomes.

Typically in the early iteration, the chromosomes have diversity in the population. So, the effect of crossover is significant. However, in the later iteration, most chromosomes will become similar so that the impact of the crossover is smaller than the earlier iteration.

1.4.4.3 Mutation

In the mutation, genes are randomly changes to the different values in the range of possible values such that:

$$x_i' = Random\left(x_i^{(L)}, x_i^{(U)}\right), \tag{1.23}$$

where x_i' is a new gene value after the mutation, $x_i^{(U)}$ is the upper bound of the gene value, and $x_i^{(L)}$ is the lower bound of the gene value. The mutation is essential because it expands the capability of diversification, and the probability of mutation is set by hyperparameter. Higher mutation probability can realize more diversification, refrain from the localized search, and interrupt the convergence.

1.4.4.4 Termination Criteria

After finishing each stage, the termination criteria will decide whether the current population is selected as the last population or back to the selection process. The standard termination criteria of the genetic algorithm are the number of iterations, run time of computation, and the improvement curve of the latest duration of iterations. Fig. 1.4 illustrates the flow of the whole genetic algorithm process. Until the termination criteria are satisfy with some condition such as number of iterations, convergence, and accuracy, the process of selection, mating, crossover, and mutation.

1.4.4.5 Hyperparameters

Through the genetic algorithm process, several hyperparameters determine the behavior of the algorithm: The size of population M, the number of parents, the probability of crossover, and the probability of mutation. The best combination of hyperparameters depends on specific problems' search space. It is often required to search for good hyperparameters by multiple experiments.

1.4.5 Simulated Annealing

Simulated annealing is a metaheuristic inspired by physical phenomena: annealing in the metallurgy field. Kirkpatrick *et al.* developed the simulated annealing in 1983[34]. The physical annealing process heats the solid, and it cools gradually until it has the regular crystal lattice arrangement without faults. The simulated annealing starts from the random solution, called the current state. Then, a new state in the neighborhood of the current state is generated. If the fitness function value of the new state is better than the current state, the new state is accepted. Even if the new state is worse than the current state, the new state is accepted if the acceptance function decides the acceptance of the new state. The value of the acceptance function depends on the sys-

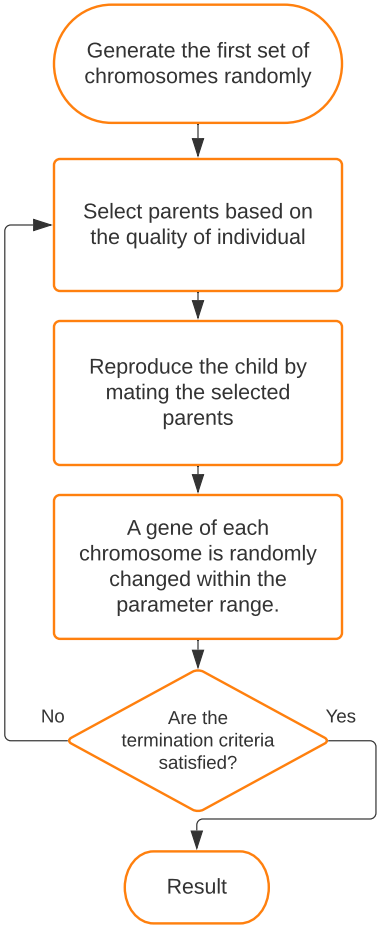


Figure 1.4: The flow of the genetic algorithm.

tem temperature. The system temperature is high at the beginning of the optimization process, but it gradually decreases, and the acceptance of unimproved state changes also decreases.

Initial state X is defined as:

$$X = (x_1, x_2, ..., x_i, ..., x_N), \tag{1.24}$$

where N is the number of decision variable and x_i is ith decision variable. Then, new neighborhood decision variables are generated with the random walk such that:

$$X_i^{new\ state} = (x'_1, x'_2, ..., x'_i, ..., x'_N), \tag{1.25}$$

$$x_i' = x_i + Random(-\epsilon, \epsilon), \tag{1.26}$$

where ϵ is a small value. Then, acceptance function determines whether this newly generated state is accepted or not. This function is defined as:

$$P(X, X^{new\ state}) = \begin{Bmatrix} 1 & if\ F(X^{new\ state}) < F(X) \\ e^{\frac{-\Delta F}{\lambda}} & Otherwise \end{Bmatrix}, \tag{1.27}$$

$$\Delta F = |F(X^{new\ state}) - F(X)|, \tag{1.28}$$

where $F(X)$ is the fitness function value, and λ is the system's temperature. If the new state is better than the previous value, the new state is always accepted. Otherwise, acceptance is determined by the function depending on the system's temperature. When the system's temperature is high, acceptance is easy, but the acceptance rate decreases when the system's temperature is low.

The thermal equilibrium at each temperature occurs when many new states are generated. This number is denoted by β.

The reduction of system's temperature occurs in the linear decreasing method as follows:

$$\lim_{t \to +\infty} \lambda_t = 0,\ t > 0, \tag{1.29}$$

$$\lambda_t = \lambda_0 - \alpha \times t, \tag{1.30}$$

$$\alpha = \frac{\lambda_0 - \lambda_T}{T}, \tag{1.31}$$

where t is the interation number of optimization process, λ_0 is the initial temperature, λ_t is the temperature in tth iteration, T is the total iterations, and α is the cooling factor.

The termination criteria determine whether the algorithm goes to the next iteration or stops. The criteria are typically the number of iterations, the improvement curve of the states between the latest iterations, and computation time. Fig. 1.5 illustrates the flow of the whole genetic algorithm process. The parameters are changed if the acceptance criterion satisfied, and the temperature is gradually decreasing through the condition change of the thermal equilibrium.

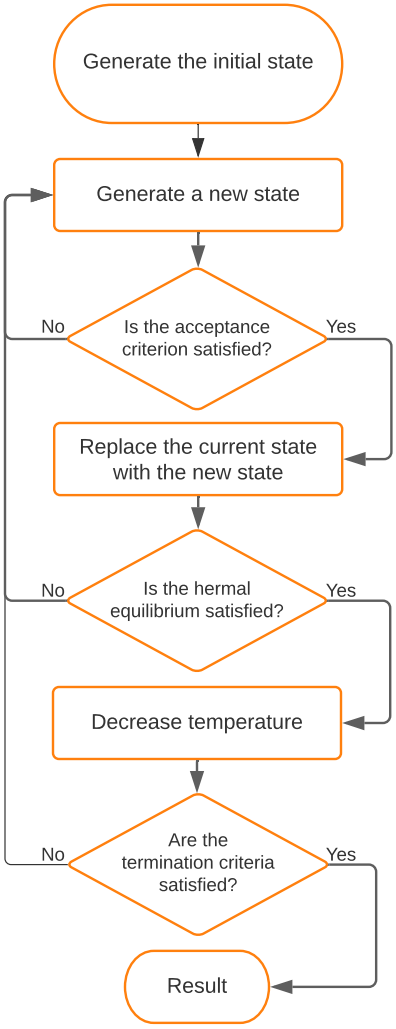


Figure 1.5: The flow of simulated annealing.

1.4.6 Trust-Region Method

The Trust-region method is a numerical optimization method[35]. It starts from a region around the initial best point, and the region around the initial point is assumed to be trusted. In other words, the model is well represented in the region around the initial point. Then, it takes a step forward within the trust-region and gradually expands the trust-region area. Here, the initial point is defined

as x_k. The set of points around x_k is a trust-region in this case. This region is defined as:

$$B(x_k, \Delta k) = ||x - x_k|| \leq \Delta_k, \tag{1.32}$$

where Δ_k is a radius of area. When the model is considered quadratic with derivative information such as:

$$m(x) = f(x_k) + \nabla^T f(x_k)x + x^T \nabla^2 f(x_k)x. \tag{1.33}$$

the optimization problem is reduced to solve a trust-region subproblems:

$$\begin{cases} \min_x = m(x) \\ s.t.\ \ x \in B(x_k, \Delta_k) \end{cases} \tag{1.34}$$

The size of the trust-region is necessary to be updated through the iteration. If the iteration makes a satisfactory reduction, Δ_k becomes larger. If the improvement is limited in the iteration, Δ_k is stationary or decrease. the ratio of actual reduction in the objective function is

$$\rho_k = \frac{f(x_k) - f(\overline{x})}{m(x) - m(\overline{x})}. \tag{1.35}$$

The trust-region size is determined by following rule:

$$\Delta_{k+1} = \begin{cases} \gamma_{inc}\Delta_k, & \text{if } \rho_{\mathrm{k}} \geq \eta_1 \\ \Delta_k, & \text{if } \eta_0 \leq \rho_{\mathrm{k}} < \eta_1 \\ \gamma_{dec}\Delta_k, & \text{if } \rho_{\mathrm{k}} < \eta_0 \end{cases} \tag{1.36}$$

$$0 \leq \eta_0 \leq \eta_1 < 1 \tag{1.37}$$

$$0 < \gamma_{dec} < 1 \tag{1.38}$$

$$\gamma_{inc} > 1 \tag{1.39}$$

Also, the trust-region center is determined as follows:

$$x_{k+1} = \begin{cases} \overline{x}, & \text{if } \rho_{\mathrm{k}} \geq \eta_0 \\ x_k, & \text{if } \rho_{\mathrm{k}} < \eta_0 \end{cases} \tag{1.40}$$

1.4.7 Regression Algorithm

The regression model is an estimated model from a set of statistical processes in the relationships between datasets and target variables. There are many kinds of regression algorithms. In this research, two regression algorithms are selected due to their applicability for the small amount of dataset or dataset with missing data. Gradient boosting regression is often the best accuracy on many problems and handles a mixture of feature types, and it is unsuitable for very high dimensional sparse feature problems in terms of accuracy and computational cost.

1.4.7.1 Gaussian Process Regression

Gaussian Process Regression (GPR) is a non-parametric, Bayesian approach to regression. In the beginning, a linear function is assumed such that:

$$y = wX + \epsilon. \tag{1.41}$$

The Bayesian approach specifies a prior distribution, $p(w)$, on the parameter, w. Then, probabilities defined by the data sets are relocated using Bayes' rule.

$$p(w|y, X) = \frac{p(y|X, w)p(w)}{p(y|x)}. \tag{1.42}$$

Here, posterior is shown as likelihood times prior divided by marginal likelihood. The posterior distribution or updated distribution incorporates information from the prior distribution($p(w)$) and datasets(X). Then, prediction at unknown points(x') is calculated as:

$$p(f'|x', y, X) = \int_w p(y'|x', w)p(w|y, X)dw. \tag{1.43}$$

At this point, it is assumed that the prior and likelihood is Gaussian through the integration, and this Gaussian distribution is the prediction with variance.

Since Gaussian process regression is non-parametric, it calculates the probability distribution through the functions that fit the datasets. The Gaussian process function is represented using a mean function $m(x)$ and covariance function, $k(x, x')$. There are several covariance kernel functions to fit. In this research, constant with radial basis function (RBF) with noise is selected:

$$k(x, x') = \sigma_f^2 \exp -\frac{1}{2l^2}||x - x'||^2. \tag{1.44}$$

The prediction of the two-dimensional case is modeled with uncertainty, as shown in Fig. 1.6 The points that have sufficient training data have very narrow confidence interval, and the point without enough training data shows more uncertainty.

1.4.7.2 Gradient Boosting Regression

Gradient boosting[36] is a way to make a prediction model by the ensemble of weak prediction models. The typical choice of the weak prediction model is decision trees. Gradient boosting regression is customized to realize higher performance. The representative boosting system is, for instance, XGBoost[37] The fundamental process of making predictions is as follows. Here, the objective function which we want to minimize is L. The initial model is described as:

$$F_0(0) = \arg\min_{\gamma} \sum_{i=1}^{n} L(y_i, \gamma), \tag{1.45}$$

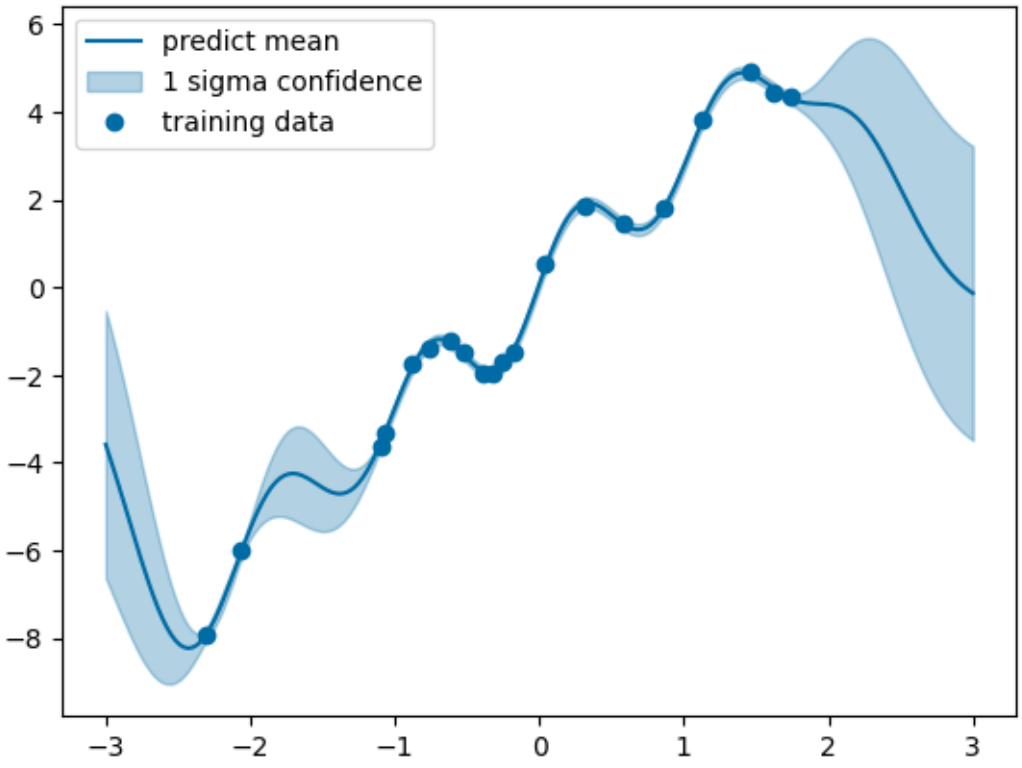


Figure 1.6: The prediction example of GPR.

where y_i is ith target value, γ is predictions. n is the number of data. For a given number of iteration steps M, calculate the pseudo residuals using gradient :

$$r = -\left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i}\right]_{F(x)=F_{m-1}(x)} \quad \text{for } i = 1, ..., n. \tag{1.46}$$

Then, fit base learner $h_m(x)$ to pseudo residuals r. The updated model is shown using base learner and previous model with step magnitude multiplier γ_m:

$$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x). \tag{1.47}$$

Step magnitude γ is multiplied by learning rate, a factor between 0 and 1 for each gradient step to realize slow convergence toward observed values.

1.4.8 Application of Metaheuristics in the Nuclear Engineering Field

Genetic algorithms have been used in the LWR design in the nuclear engineering field, including core design[38][39][40], plant design[41], fuel management[42][43], spent fuel management[44], and fuel loading pattern[45]. Recently, more applications for the Generation IV reactor have been conducted, such as FR core optimization by genetic algorithm[46] and artificial neural network[47]. Also, an original metaheuristics algorithm combining the characteristics of famous algorithms was proposed by Bevins and Slaybaugh[48].

Even though these metaheuristics algorithms are powerful tools to optimize the design in the nuclear engineering field, it is time-consuming because it requires exploring a vast design parameter space with many physics model calculations. A recent study attempted to solve this problem using regression methods in the FR optimization problem based on genetic algorithms[49][50] and artificial neural networks[51]. They built surrogate models and used them through genetic algorithms to reduce the computation time of direct Monte Carlo physical simulations.

In the field of TRU minimization, there are multiple pieces of research for optimization, as shown in the section 1.2.3. However, there is no research considering systematic mathematical optimization. This paper shows the optimization framework which can be applied to the waste minimization problem.

2. METHODOLOGY

This chapter explains the development of an optimization framework for the reactor design parameters.

2.1 Framework of Optimization

In this optimization framework, the optimal design parameters are searched through the multiple stages of algorithms. The optimization target is assumed to be the nuclear reactor design optimization problem which includes black-box constraints in the physics model simulation. These problems are challenging because of many design parameters, including continuous and discrete, nonlinear characteristics between design parameters and target output values, and black-box constraints. A surrogate model, an equation-oriented approximation of the physics model simulation, can significantly decrease computation time. However, the optimization process has approximation errors due to the incompleteness of the surrogate model, and the optimal answer can be inaccurate. This optimization framework proposes the methodology to reduce the required computation time and keep necessary accuracy. Fig. 2.1 shows the schematic flow of the framework consisting of four stages of the process.

The first stage of the optimization process is a sensitivity analysis for investigating and selecting the vital optimization design variables. The importance is evaluated based on the sensitivity coefficient and correlation among design parameters and outputs.

The second stage of the optimization process is a feasible search. In this stage, the feasible region of the design parameter space is looked over using the trust region method rather than finding the optimal point.

The third stage of the optimization process is making a surrogate model. The surrogate model is built using the inputs and outputs data obtained in the sensitivity analysis and the feasible search. Gaussian process regression is used for the regression algorithms to construct the model due to the accuracy and speed. The surrogate model can reduce computation time drastically compared with

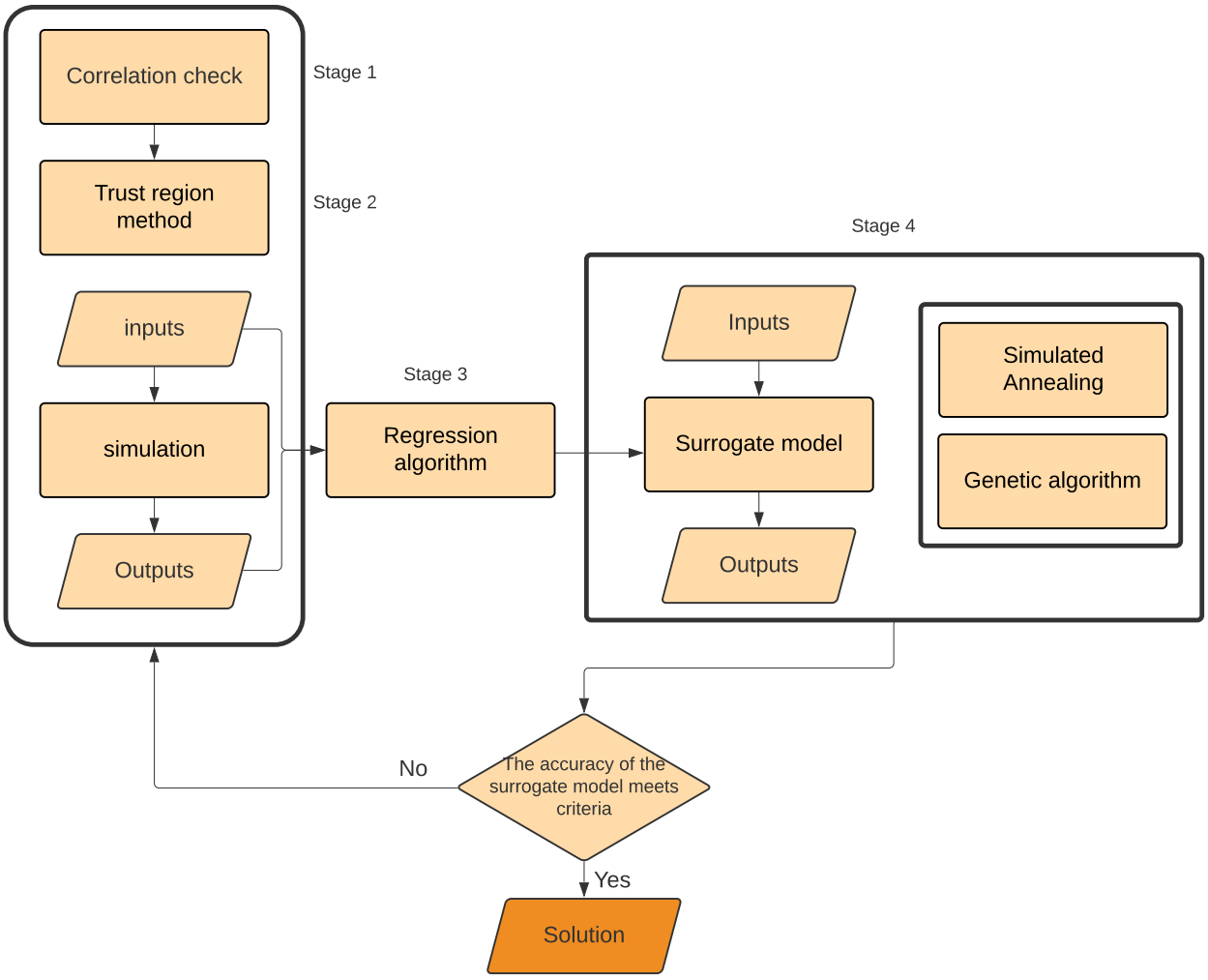


Figure 2.1: Schematic flow of the optimization framework.

the original physics simulation.

The fourth stage of the optimization process is optimal search. In this stage, the optimal point search is performed in the surrogate model. The search algorithm is the hybrid metaheuristics based on genetic algorithm and simulated annealing.

After the process of each stage, the validity of the surrogate model is evaluated to decide the optimal point candidate should be finalized or not. Both the cross-validation and the test is performed for the validation. If the accuracy of the surrogate model does not meet the criteria, additional data points are added, and then the iteration from stage 2 to stage 4 is performed again.

2.2 Stage 1: Correlation Check and Sensitivity Analysis

The sensitivity analysis is performed to search the correlation between input parameters and output parameters such as objectives and constraint parameters. Each input parameter is sampled using centered Latin Hypercube Sampling with fixed other parameters in the range of design space limit. The sampling is conducted until their means, and standard deviations reach convergence. The Pearson correlation coefficients between input parameters, objectives, and constraints values are calculated to evaluate the linear correlations. The positive coefficient shows that the output parameter increases when the input parameter increases, while the negative coefficient indicates that the input parameter's increase results in the output parameter's decrease. If a specific input parameter has negligible correlation coefficients with all objectives and constraint values, this input parameter is omitted from the optimization problem. The input value is assumed to be a fixed value through the process.

2.3 Stage 2: Feasible search

The algorithm in the feasible search stage aims to find a feasible point rather than find an optimal point. There is an idea that a trust region-based feasible search is performed before solving an optimization problem[52]. Some optimization algorithms look for feasible points and simultaneously decrease the objectives using penalty-based methods. However, the advantage of a feasible search is that at least a feasible point is obtained when the code stops for some unexpected error. It is imperative when the infeasible region is broad in the design search space and the computation time for the infeasibility and optimization have a significant difference. The feasible search starts when the initially given parameter is infeasible, and the feasible search continues until a feasible point is obtained. Fig. 2.2 shows the schematic flow of the feasible search stage. This stage can be interpreted as the minimization problem for a constraint violation function. The initial input parameter is generated randomly using centered Latin Hypercube Sampling to spread the sample points evenly across the design space range. When the given parameter is infeasible, the constraint violation is calculated. Then, using the trust-region method, the neighborhood of the initial point

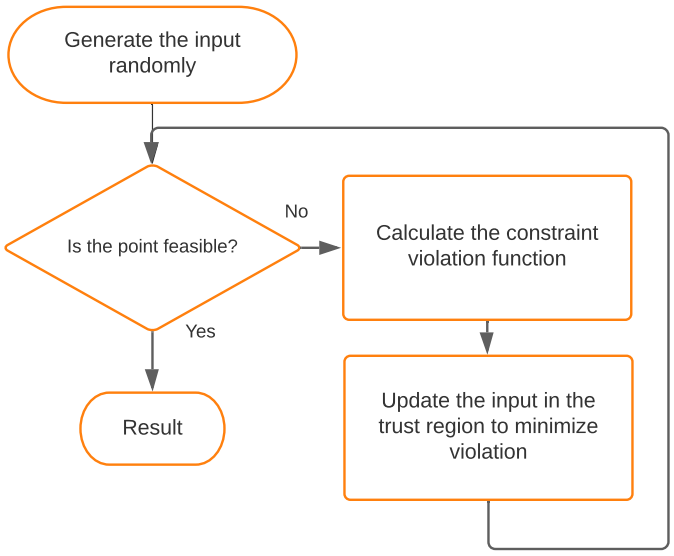


Figure 2.2: Schematic flow of the feasible search stage.

is searched to minimize the constraint violation. This problem is shown as

$$\begin{cases} \min_x \ \theta^r(x), \\ s.t. \ \ g_j(x) \geq 0 \ \ \forall j \in \{1...q\}, \\ ||x - x_k|| \in \Delta_k, \end{cases} \tag{2.1}$$

where θ^R is the constraint violation, g is the constraint, q is the total number of constraint, x_k is the neighborhood of a point, and Δ_k is the range of trust region.

The neighborhood search by the trust region method is iterated, and when the feasible region is obtained by updating the input parameters, that point is recorded, and the feasible stage ends.

2.4 Stage 3: Regression Analysis

The calculation of the physics simulation model is computationally expensive. Even though metaheuristic algorithms can efficiently search the design value optima in the design search space,

it takes a significant amount of time if the search space is vast. So, the surrogate model constructed by the regression algorithm is often used instead of the physical simulation model. The surrogate model is built using a Gaussian process regression in this framework. All the input values and output values from the sensitivity analysis stage and feasible search stage are used for making the first surrogate model. The optimal search is performed in this surrogate model in the next stage. The surrogate model is incomplete and inaccurate compared with the original physical model, so the results obtained in the surrogate model have errors. Hence, the iteration approach is adopted to build a better surrogate model and reduce errors. If the error is not below the threshold after the optimal value is obtained in the optimal search stage, the process is back to the feasible search stage. Then, more input and output parameters emphasizing the vicinity of the optimal point candidate are added to improve the accuracy of the surrogate model.

2.5 Stage 4: Metaheuristic Optimization Search in Surrogate Model

As mentioned in the chapter 1 , metaheuristic is powerful tool for the black-box optimization problem. Genetic algorithms and simulated annealing are representative metaheuristic algorithms and are often selected for engineering black-box optimization problems. However, the original genetic algorithms and simulated annealing typically do not realize a well-balanced search for intensification and diversification. Hence, the hybrid of genetic algorithm and simulated annealing was adopted in this optimization framework. There are several ways to combine genetic algorithms and simulated annealing. Genetic Algorithm-Simulated Annealing (GA-SA) Hybrid[53] is used as the base algorithm in this framework.

The schematic flow of the hybrid of genetic algorithm and simulated annealing is shown in Fig. 2.3. The fundamental flow is similar to the genetic algorithm, but the concept of temperature, which is used in the simulated annealing, is adopted. If the temperature is high, the chromosomes can change their genes freely. However, they do not accept the worse chromosomes after the temperature decreases. The conventional genetic algorithm tends to diverse the parameters through the algorithm iteration, but the hybrid of genetic algorithm and simulated annealing algorithm considers more intensification in the later iterations.

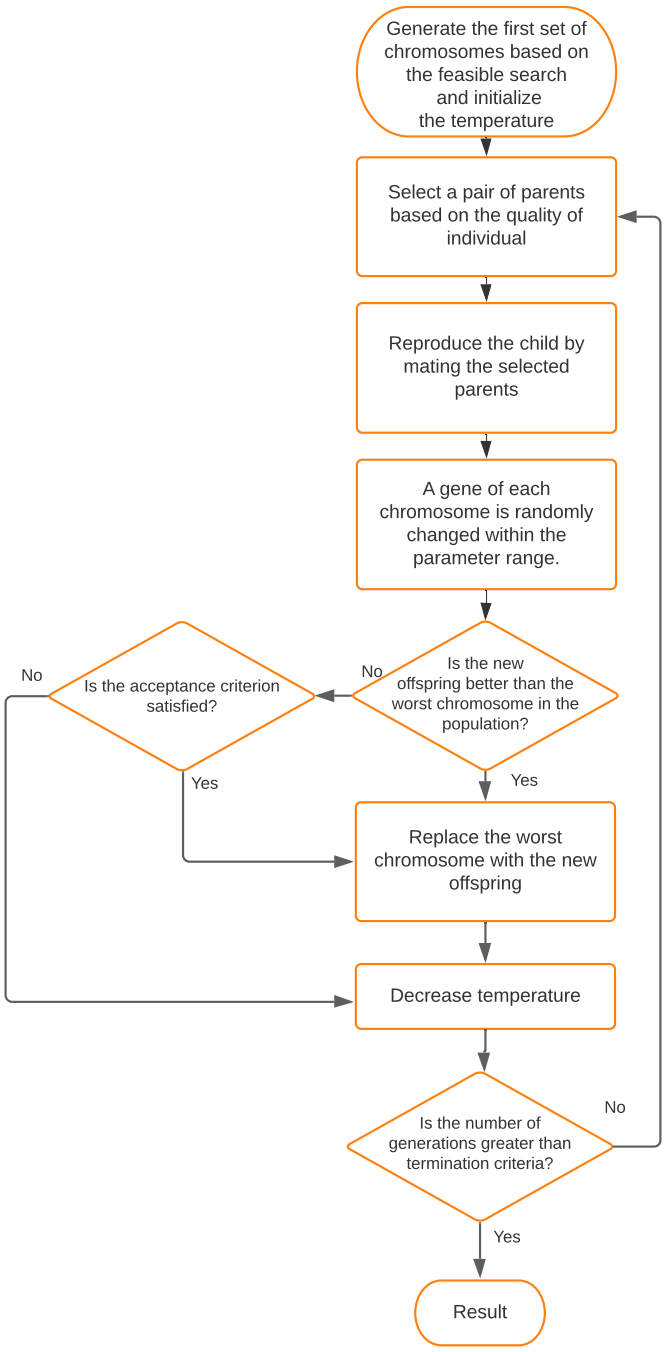


Figure 2.3: Schematic flow of the hybrid of genetic algorithm and simulated annealing algorithm.

The initial population set is generated from the result of the feasibility search stage. So. the first population matrix is as shown in Eq. 1.15. Each column represents each design parameter. So,

each column has randomly generated numbers in the range of each design space.

Then, a pair of parents are selected. In this framework, the modified proportionate selection is employed as a selection method. The probability of the kth chromosome being set is evaluated as follows:

$$P_k = \frac{F(X_k)^{\alpha t}}{\sum_{j=1}^{M} F(X_j)^{\alpha t}}, \tag{2.2}$$

where $F(X)$ is the fitness function of a chromosome, α is a constant parameter depending on the problem, t is the number of generations in the hybrid algorithm. The fitness function of the multiobjective is calculated using the weighted sum method shown in equation (1.9). In this modified selection, the probability of a better fit chromosome being chosen increases when the generation proceeds. In the earlier generation, less-fit chromosomes have some chance of choice, but their luck will gradually decrease at the later generation.

An average-bound crossover operation[54] is adopted in the crossover process instead of a traditional point crossover or uniform crossover. The selected pair of chromosomes will undergo the crossover operation in the probability of $p_c \in [0, 1]$, a crossover probability parameter. In this operation, four new offspring are reproduced from a pair of two parents instead of two offsprings, typical for the conventional crossover operation, as shown in the section 1.4.4.2. Then two offsprings are selected from these four offsprings based on the fitness function. The two of these offsprings are offsprings generated by average crossover as follows:

$$\mathbf{O}_1 = \frac{(\mathbf{P}_1 + \mathbf{P}_2)}{2}, \tag{2.3}$$

$$\mathbf{O}_2 = \frac{(\mathbf{P}_{\max} + \mathbf{P}_{\min})(1 - w_a) + (\mathbf{P}_1 + \mathbf{P}_2)w_a}{2}. \tag{2.4}$$

Also, the other two of the offsprings are offsprings generated by bound crossover as follows:

$$\mathbf{O}_3 = \mathbf{P}_{\max}(1 - w_b) + \max(\mathbf{P}_1, \mathbf{P}_2)w_b, \tag{2.5}$$

$$\mathbf{O}_4 = \mathbf{P}_{\min}(1 - w_b) + \min(\mathbf{P}_1, \mathbf{P}_2)w_b, \tag{2.6}$$

where offspring vector $\mathbf{O}_j$ and parent vector $\mathbf{P}_k$ is shown as:

$$\mathbf{O}_j = [o_{j,1}, o_{j,2}, ..., o_{j,i}, ..., o_{j,N}], \quad j = 1, 2, 3, 4, \tag{2.7}$$

$$\mathbf{P}_k = [p_{k,1}, p_{k,2}, ..., p_{k,i}, ..., p_{k,N}], \quad k = 1, 2, \tag{2.8}$$

O is a gene of offspring vector, N is the number of genes. $w_1, w_2 \in [0, 1]$ is weight parameters for average crossover and bound crossover, respectively. When $w_1 = 1$, two offsprings from average crossover are the same: middle point between two parents. When it approaches 0, an offspring will gradually increase and reach above a larger parent value. On the other hand, $w_2 = 1$ generates the offspring that is the maximum or minimum combination of parents. When it approaches 0, the offspring will gradually reach the highest and the lowest parameters. $\mathbf{P}_{\max}$ and $\mathbf{P}_{\min}$ is a vector consisting of either maximum or minimum value of each gene parameter. Note that they are not the maximum or minimum values in the selected parents. They are defined as:

$$\mathbf{P}_{\max} = \left[x_1^{(U)}, x_2^{(U)}, ..., x_i^{(U)}, ..., x_N^{(U)}\right], \tag{2.9}$$

$$\mathbf{P}_{\min} = \left[x_1^{(L)}, x_2^{(L)}, ..., x_i^{(L)}, ..., x_N^{(L)}\right], \tag{2.10}$$

where $x_i^{(U)}, x_i^{(L)}$ is the maximum or minimum number of gene x_i, respectively. $\max(\mathbf{P}_1, \mathbf{P}_2)$ and $\min(\mathbf{P}_1, \mathbf{P}_2)$ is a vector consisting of larger or smaller values of each gene between $\mathbf{P}_1$ and $\mathbf{P}_2$. For instance, $\max([1, 2, 3], [3, 2, 1]) = [3, 2, 3]$.

Two offsprings are selected based on the fitness function and pooled into the population from these four offsprings. Four offsprings will have four different characters to efficiently search for the global optima compared with the conventional crossover operation.

After the crossover operation, the mutation operation is conducted. The wavelet mutation[54] is adopted in this framework instead of a conventional uniform mutation. The selected pair of

chromosomes will undergo the mutation operation in the probability of mutation shown as

$$p_m = 1 - \tau/T, \tag{2.11}$$

where τ is the current iteration number, and T is the total iteration number. The mutated gene is calculated as follows:

$$o'_{j,i} = \begin{cases} o_{j,i} + \delta \times \left(x_i^{(U)} - o_{j,i} \right), & \text{if } \delta > 0, \\ o_{j,i} + \delta \times \left(o_{j,i} - x_i^{(L)} \right), & \text{if } \delta \leq 0, \end{cases} \tag{2.12}$$

$$\delta = \psi_{a,0}(\varphi) = \frac{1}{\sqrt{a}} \psi \left(\frac{\varphi}{a} \right), \tag{2.13}$$

where ψ is the Morlet wavelet function:

$$\psi(x) = e^{-x^2/2} \cos(5x), \tag{2.14}$$

$$\psi_{a,b}(x) = \frac{1}{\sqrt{a}} \psi \left(\frac{x-b}{a} \right). \tag{2.15}$$

So,

$$\delta = \frac{1}{\sqrt{a}} e^{-\frac{(\varphi/a)^2}{2}} \cos \left(5 \left(\frac{\varphi}{a} \right) \right), \tag{2.16}$$

where $\varphi \in [-2.5, 2.5]$ and it is generated randomly. The Morlet wavelet function is shown in Fig. 2.4. This is the symmetry function at zero, and the integral of the Morlet wavelet function is zero. Also, most of the change in the function is between the interval of $-2.5 \leq x \leq 2.5$.

If $|\delta|$ is more significant, the value change by mutation is also more significant, and if $|\delta|$ is smaller, the value change is small. From the property of wavelet function, the total positive and negative change by mutation should be approximately the same when the sample size is large enough. a is the dilation parameter which is the function of the algorithm iteration shown in

$$a = e^{-\ln(g)(1-\tau/T)^{\zeta} + \ln(g)}, \tag{2.17}$$

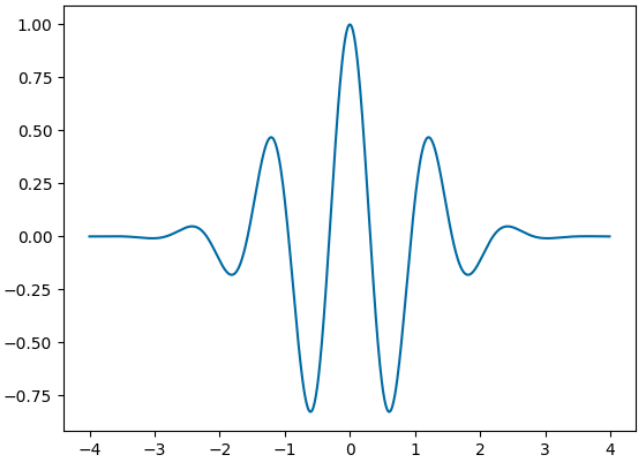


Figure 2.4: The Morlet Wavelet.

where ζ is the shape parameter, g is the upper limit of a, τ is the current number of iteration, and T is the total number of iteration. If $\zeta = 1$, $log(a)$ is linearly increases with respect to τ/T. If $\zeta > 1$, the relation between $log(a)$ and τ/T is concave curve. If $\zeta < 1$, it is convex curve. When the $zeta$ is low, a does not significantly increase until the later generation. Since larger a has more chance to obtain larger δ, low ζ keeps large shift of mutation in longer iteration period.

After the crossover and mutation operation, if the new offsprings are better than the worst in the population, the worst chromosomes are replaced with new offspring. If not, they are replaced if the acceptance criterion based on temperature is satisfied. This criterion is the same as the conventional simulated annealing criterion, as shown in inequality (1.27). Then, the temperature decreases at the end of the iteration. The hybrid algorithm stops and moves into the next stage if the generation exceeds the pre-set maximum number of generations. If not, it backs to select new chromosomes from the updated population.

Through the hybrid algorithm, the earlier generation of the population has more chance for a broad search area to find global optima. However, several parameters depending on the generation change in the later generation, and the search focuses on the intensification. For example, higher-quality individuals are selected more in the selection process, the probability of mutation decreases, the magnitude of mutation decreases, and the survival probability of less-fit chromosome

decreases. The hybrid of genetic algorithm and simulated annealing can perform well-balanced diversification and intensification to search for better optima in a limited time. In short, the primitive genetic algorithm is weak for the capability of intensification of the search. However, in this GA-SA hybrid algorithm, the diversification capability is robust at the earlier stage, and their balance shifts to the intensification at the later stage.

2.6 Validation

After obtaining a candidate of optimal values by the hybrid algorithm of genetic algorithm and simulated annealing in the surrogate model, the validation of the model is necessary. Even if the global optima is achieved in the surrogate model, it does not assure that the optimal point is also the global optima in the original physics model. So, the validity of the surrogate model is checked. The surrogate model is evaluated in two steps. The first step is cross-validation, and the second step is the test. On the cross-validation, parts of data sets are used to evaluate the surrogate model. On the other hand, the test uses new physical model test data. If the surrogate model does not pass the test, the date used for the test are abandoned and not used for the regression modeling.

In the cross-validation step, K-fold cross-validation is performed. The data sets used for the surrogate model are divided into K equal parts. Then, K-1 parts are used for the modeling, and 1 part is used for the test. This process is repeated k times in total. So, each subset of the data sets experiences a validation set. The example of the 5-fold cross-validation is shown in Fig. 2.5.

The error is calculated using the Relative Mean Squared Error (RMSE).

$$\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\frac{R_{model,i}-R_{val,i}}{R_{val,i}}\right)^2} \leq \epsilon_2, \tag{2.18}$$

where N is the number of selected data points, $R_{model,i}$ is the ith predicted fit function value in the surrogate model using subset data, and $R_{val,i}$ is the ith fit function value by the validation test subset. ϵ_2 is the RMSE threshold that is required to show the surrogate model is valid. If the surrogate model is evaluated to be accurate, the test step follows. If not, the process is back to the stage 2 with additional data points to strengthen the surrogate model.

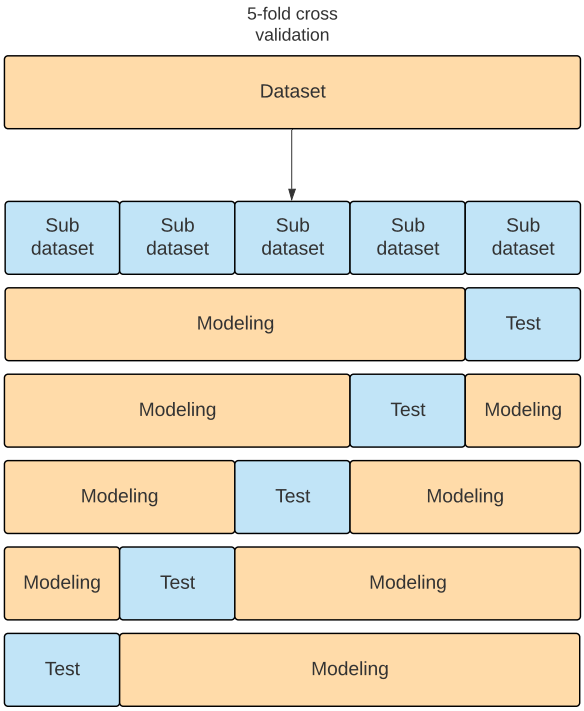


Figure 2.5: The example of the 5-fold cross-validation flow.

In the test step, both the accuracy of the optimal point candidate and the model. For the optimal point, the RMSE should be less than the tight threshold as follows:

$$\sqrt{\left(\frac{R_{Surrogate} - R_{phys}}{R_{phys}}\right)^2} \leq \epsilon_1, \tag{2.19}$$

Also, the following criterion is used for evaluating the model accuracy.

$$\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\frac{R_{Surrogate,i} - R_{phys,i}}{R_{phys,i}}\right)^2} \leq \epsilon_2, \tag{2.20}$$

where $R_{Surrogate,i}$ is the ith predicted fit function value in the surrogate model, and $R_{phys,i}$ is the ith fit function value by physical calculation. ϵ_1 and ϵ_2 is the RMSE threshold that is required to show the surrogate model is valid and $\epsilon_1 < \epsilon_2$. if the error is not below the threshold, the surrogate model

is considered inaccurate and back to the previous stage with additional data sets. If the surrogate model and the optimal point candidate pass both the validation step and the test step, the optimal point candidate is determined as the optimal design parameters.

2.7 Code Structure of Framework

The framework consists of multiple python modules, as shown in Fig. 2.6. After setting some

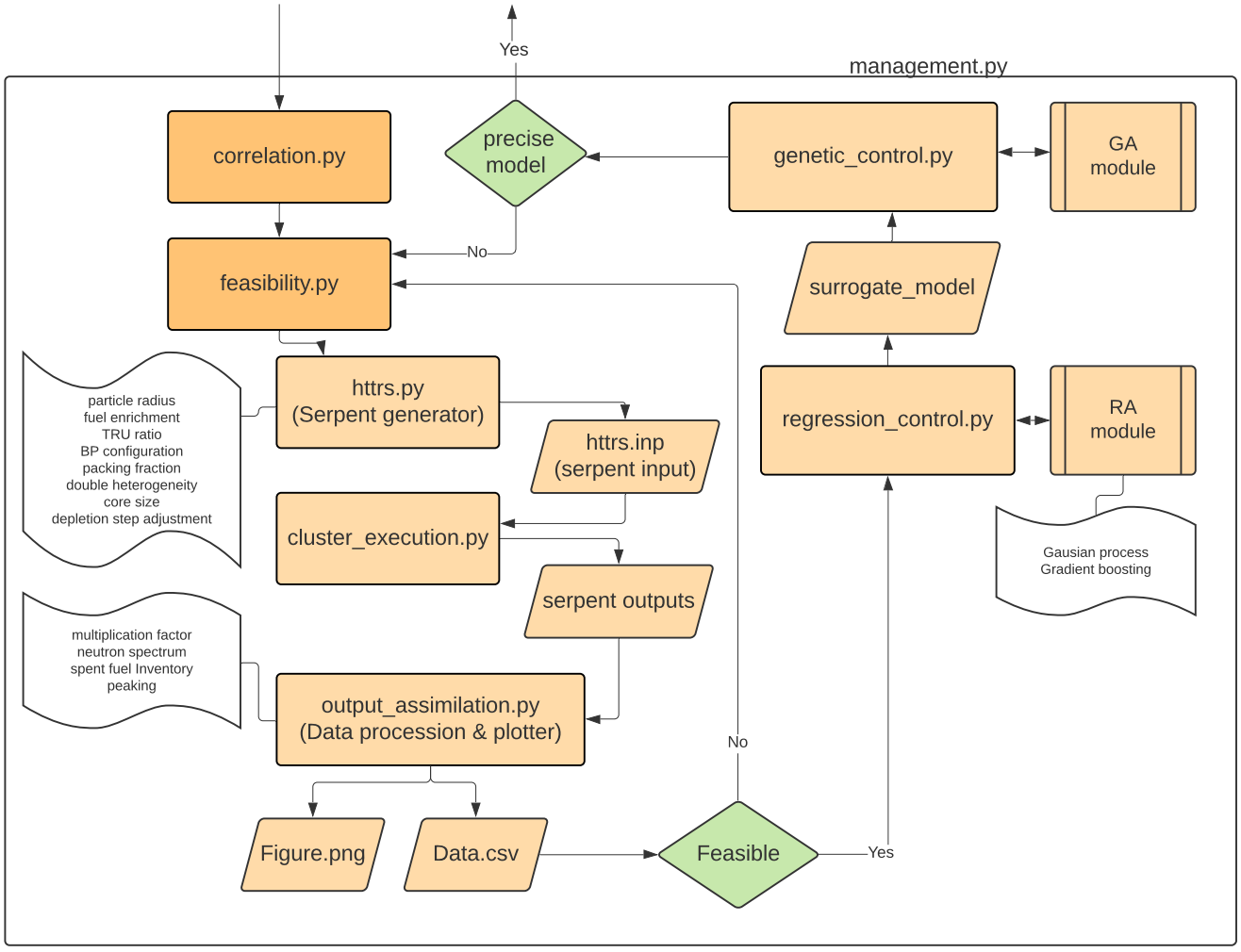


Figure 2.6: A schematic sketch of code structure.

given information such as the target design parameters, the range of design parameters, spent fuel

configuration, and hyperparameters, the framework can automatically calculate all simulation and optimization processes. If unforeseen errors happen, the results at that time will be stored, and users can resume the optimization process at that point. Since each function is modulated, this optimization framework can be adjusted for the different reactor cores and optimization targets.

management.py controls all of the flows in the framework. It calls each python module, runs functions to connect each module, writes logs, and finds errors.

correlation.py makes a sufficient set of design parameters in the given range. It gives these parameters to *httrs.py* and calculates correlation among design parameters, objective values, and meaningful outputs.

feasibility.py selects design parameters using the trust-region method and iterates the learning data generation until feasibility criteria are satisfied.

httrs.py generates serpent input files from the given design parameter information. It can set a particle radius, fuel enrichment in a different region, TRU ratio, burnable poison enrichment, packing fraction, and core size. It also creates stochastic geometry data for the heterogeneity of TRISO fuel particles in the fuel compact by tailoring packing fraction and fuel radius. This module also calculates all the necessary values for the serpent inputs. Another useful function is the time step management. Depending on the given design parameters, depletion time is quite different, so simulating in a fixed time step is time-consuming. This module generates a relatively short period of steps and checks some constraints such as criticality. If the reactor's criticality is kept at the end of the calculation, the calculation continues for additional time steps using preserved information. This time step management can eliminate unnecessary depletion calculation, and total computation time can be decreased.

cluster_execution.py is used to run the Serpent on the cluster. The necessary queue on the cluster is automatically executed until all calculations are ended.

output_assimilation.py extract necessary data from serpent output files. For instance, it converts and caluculates objective values and important output parameters from the output files and generates Comma-Separated Values (CSV) files. The values at the End of the Cycle (EOC) is

estimated by the interpolation. It can also visualize representative data such as multiplication factors through the depletion, neutron spectrum, peaking map in the reactor core. This module uses SerpentTools[55] for data format conversion.

regression_control.py makes a surrogate model using designated regression algorithms. Scikit-learn[56] is used for the regression algorithms.

genetic_control.py iterates the optimization process by the genetic algorithm. The algorithm scheme and hyperparameter can be set in this module.

2.8 Tools

For the implementation of frameowork, following packages are used.

- serpentTools[55]
- NumPy[57]
- pandas[58]
- Scikit-learn[56]
- SciPy[59]

3. REACTOR MODELS AND OPTIMIZATION RESULTS

The optimization framework is applied to the TRU incineration problem in the HTR. The reactor model is based on the HTTR [60], an existing experimental reactor in Japan, and GTHTR300[61], a conceptual commercial-scale high-temperature reactor. The reactor is assumed to be a single-batch core loading in this research scope.

3.1 Serpent Model

The HTTR and GTHTR300 are modeled using Serpent 2.32[62]. The fuel block, control rod guide block, reflector block are built in the high fidelity 3D model. Traditionally, double heterogeneity of the reactor core was the problem to model the high-temperature reactor. In this research, the double heterogeneity of TRISO fuel is realized using the serpent disperse function. Note that the movement of the control rod is not considered in this research scope. In the framework, a code generates the serpent input by setting the design parameters such as packing fraction and fuel composition.

3.2 HTTR Optimization Problem

The first reactor core model is based on the HTTR. The HTTR is a helium-cooled, graphite-moderated test reactor in Japan. Since HTTR is a relatively small reactor, the number of neutrons in the core is insufficient compared with an annular graphite-moderated core such as GT-HTR300. However, the small core and detailed publications of reactor design parameters are suitable for the test case of the framework. So, this reactor optimization problem is set for the preliminary case of the framework application.

3.2.1 HTTR Reactor Parameters

The primary sources to build the model are the IAEA working group report[63] and the International Reactor Physics Experiment Evaluation Project (IRPhEP) report[64]. The primary design specifications are shown in Table 3.1. Since the HTTR is an experimental reactor, the output

Table 3.1: The Major Design Specification of the HTTR.

	HTTR
Thermal Power	30 MW
Outlet Coolant Temperature	850-950
Core Diameter	2.3 m
Core Height	2.9 m
Fuel	UO_2 (TRISO coated)
Enrichment	3.4 - 9.9 wt%
	6 wt% (on average)
Type	Pin-in-Block
Block Material	Graphite
Coolant	Helium
Number of Fuel Blocks	150
Number of Fuel Columns	30
Number of Pairs of Control Rods	16

power and core size are not large. The outlet coolant temperature is one of the highest in the HTRs. The fuel is TRISO coated Uranium oxide fuel, and there is multiple different Uranium enrichment depending on the region in the core.

Fig. 3.1 and 3.2 show the horizontal and axial view of the serpent HTTR core model.

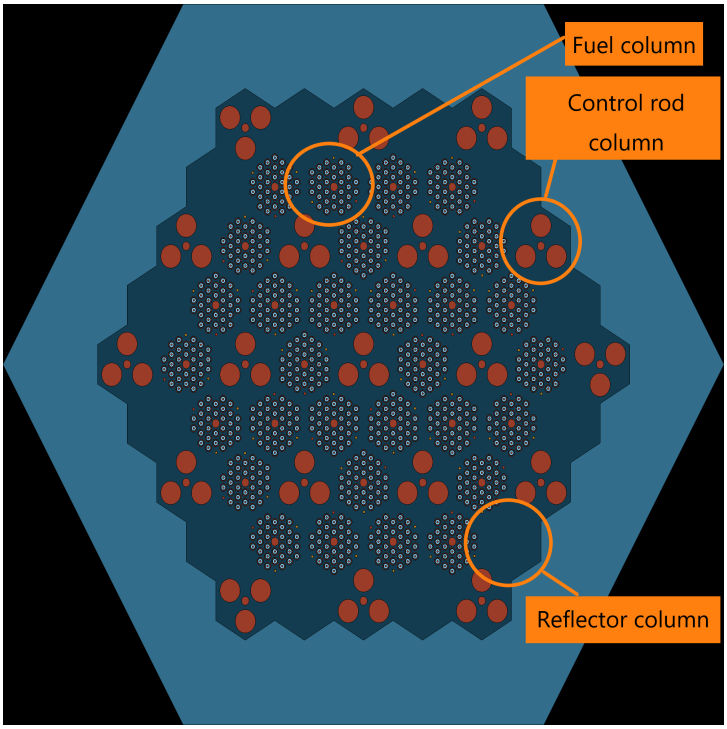


Figure 3.1: Horizontal cross-sectional view of HTTR core.

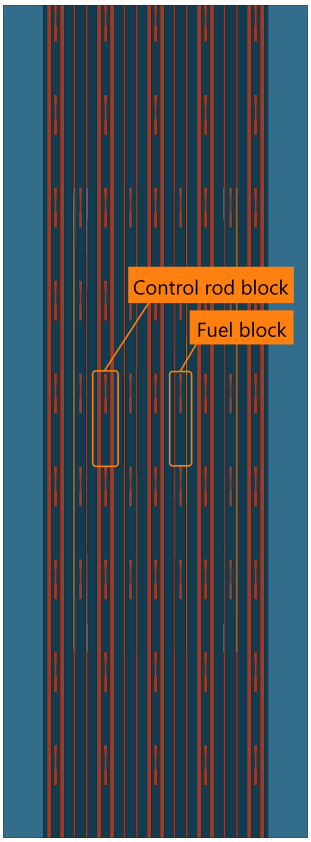


Figure 3.2: Axial cross-sectional view of HTTR core.

The composition of TRU is based on the spent fuel from the AP1000 model simulation[65][66]. The AP1000 is operated with the 3 wt% of Uranium oxide fuel for 630 days at a power of 3400MW. The TRU portion of the spent fuel from AP1000 takes 12 years before loading the GT-HTR300. The mass fraction of TRU composition is shown in Table 3.8. The mass fraction of TRU composition is shown in Table 3.2.

Table 3.2: TRU composition of the spent fuel from AP1000.

Nuclide	Mass fraction	Nuclide	Mass fraction
Np-236	7.83E-08	Am-241	5.34E-02
Np-237	3.32E-02	Am-242	1.02E-05
Np-238	7.62E-03	Am-243	5.45E-03
Pu-239	5.11E-01	Cm-243	9.47E-06
Pu-240	2.77E-01	Cm-244	8.62E-04
Pu-241	6.61E-02	Cm-245	4.74E-05
Pu-242	4.50E-02	Cm-246	3.26E-06
Pu-244	9.73E-07		

The Energy spectrum in the fuel region one of the GT-HTR300 with 27% of TRU ratio is shown in Fig. 3.3. The design parameters for this spectrum are 27% of TRU ratio, 287μm of fuel kernel radius, 15% of packing fraction, 7% of Uranium enrichment, and 5% of boron enrichment. Compared with the spectrum of HTR with the UO_2 fuel, the peak of the thermal region is lower because some TRU nuclides such as Am-241, Pu-240, and Pu-242 have high absorption capability of the thermal neutron. If the TRU ratio increases, the spectrum becomes harder. The peak of the spectrum between 10^{-8} and 10^{-6} increases when the TRU ratio decreases, while it decreases when the TRU ratio increases.

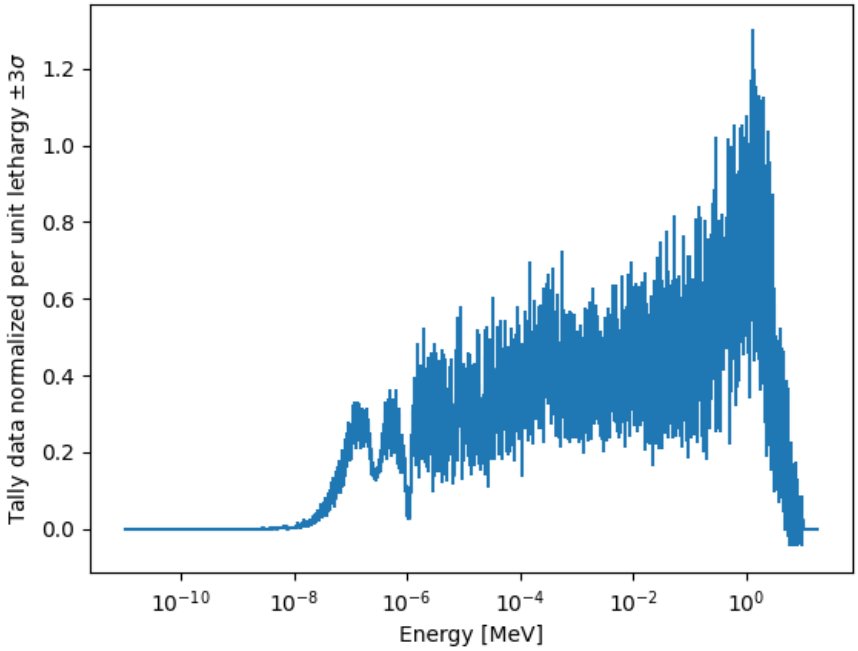


Figure 3.3: The energy spectrum in the fuel region of the HTTR with 27% TRU ratio.

3.2.2 Optimization Target

The major reactor core-related design parameters are selected for the optimization target. The list of design parameters with their value ranges is shown with these ranges in Table 3.3. The fuel

Table 3.3: The target design parameters and their range of values for the HTTR.

Design parameters	Value ranges
TRU ratio	10 - 90 %
Radius of fuel kernel	150 - 425 μ m
Packing fraction	15 - 40 %
Uranium enrichment	3 - 15 %
Boron enrichment	1 - 5 %

kernel is composed of a homogeneous mixture of Uranium and TRU. The TRU ratio is the initial fraction of TRU fuel in the fuel kernel loaded to the HTR. The remaining part of the kernel is enriched Uranium. The radius of the fuel kernel has uncertainty in actual fuel production. However, all fuel kernels are assumed to be the same radius for the design parameter in this problem. The radius of each coating layer surrounding the fuel kernel is adjusted depending on the radius of the fuel kernel. The thickness of each layer is the same in any radius cases of fuel kernel. The packing fraction is the volume percentage of TRISO fuel in the fuel compact. Uranium enrichment is the weight percentage of U-235 in the uranium portion(U-235 and U-238) of the fuel kernel. The distribution of the TRISO fuel in the fuel compact was modeled heterogeneously using stochastic methods. Some sets of fuel blocks are grouped, and stochastic heterogeneous geometry was generated for each group of fuel blocks for memory reduction of the simulation. The boron enrichment is the percentage of B-10 in the burnable poison.

3.2.3 Objective Function

The objective funciton to evaluate the minimization is set destruction rate of the TRU, D. The objective function for waste minimization is defined as:

$$\begin{aligned} \text{max. D} &= 1 - \frac{\sum_{i=1}^{n} m_{i,EOC}}{\sum_{i=1}^{n} m_{i,BOC}} && (3.1) \\ k_{\text{eff,BOC}} &\geq 1.0 && (3.2) \\ \rho(t) &< 0. && (3.3) \end{aligned}$$

where $m_{i,EOC}$ is the total mass of TRU nuclide i at the EOC, $m_{i,BOC}$ is the total mass of TRU nuclide i at the Beginning of the Cycle (BOC), and N is the total number of target TRU nuclides. $k_{\text{eff,BOC}}$ is the effective neutron multiplication factor at the BOC. $\rho(t)$ is the reactivity coefficient.

3.2.4 Selected Algorithm and Hyperparameters

In the HTTR problem, the selected algorithms are simpler than those in Chapter 2. The surrogate model is made using the pairs of input design parameters and TRU destruction rates obtained from the Serpent simulations. The Gaussian process regression is selected for the regression algorithm. The kernel of the Gaussian process regression is the RBF without parameter optimization.

Instead of a hybrid of genetic algorithm and simulated annealing, the conventional genetic algorithm was applied to search the optima. The detail of the genetic algorithm flow is shown in Fig. 3.4. Twenty sets of design parameters or chromosomes are randomly selected as the first generation, and the surrogate model estimates each set to obtain the TRU destruction rate after the depletion. If there are five design parameters, for example, a chromosome in the generation has five genes. As a crossover process, ten chromosomes are selected in a roulette method for mating, and weight is based on the destruction rate so that chromosomes with a higher destruction rate can survive more for the next generation. Then, each selected mate has two offspring by exchanging genes depending on the designated crossover probability.

Furthermore, each offspring occasionally occurs mutation in the designated mutation proba-

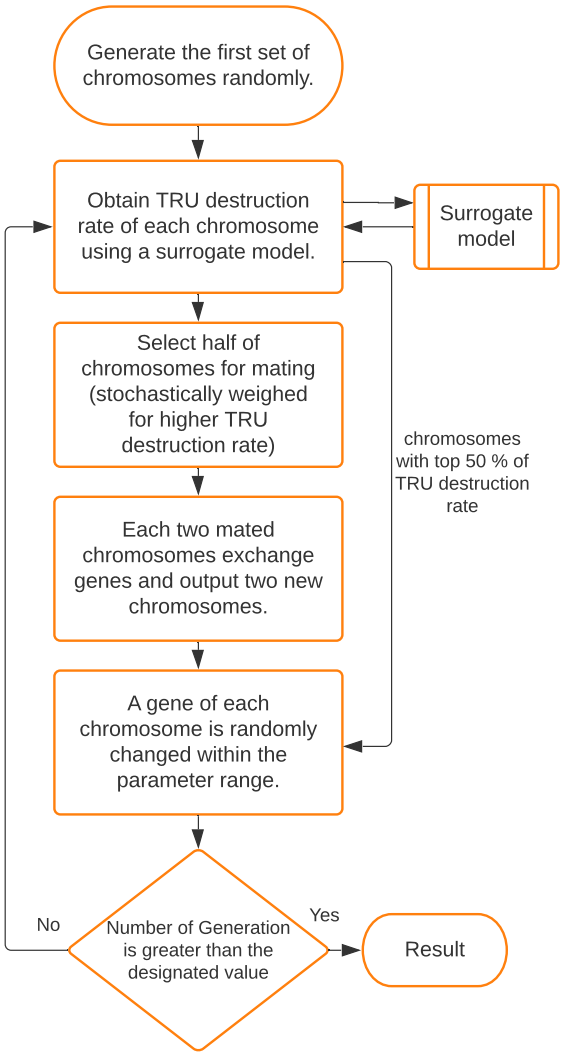


Figure 3.4: A macroscopic flow of the genetic algorithms.

bility. When the mutation occurs, a gene randomly changes in the possible range. Finally, all offspring and original chromosomes with higher destruction rates are selected as the following generation sets. The genetic algorithm parameters are shown in Table.3.4. Crossover and mutation probabilities are determined by grid search.

Table 3.4: The genetic algorithm's hyperparameters.

Parameter	Value
Number of chromosomes per generation	20
Number of generations	300
Crossover probability	0.8
Mutation probability	0.1

3.2.5 Results and Discussion

Fig. 3.5 shows the last iteration of genetic algorithms in the framework, and Table 3.5 shows the final destruction rate and design parameters.

Through the generation, every parameter has changed to the nearly minimum or maximum border of each design parameter range. The result is that each parameter has positive or negative linear correlations to the destruction rate, and there is no infeasible region at the maximum or minimum boundary combinations of design parameters in this design rage and this reactor. The trend of each design parameter is analyzed in more detail in the next section.

Also, the TRU destruction rate is shallow. For the TRU incineration problem in the thermal spectrum, some nuclides such as Pu-239 will decrease the amount by the fission through the burnup, but some TRU nuclides increase the amount because of the capture reactions of the other nuclides. Even though part of these nuclides can fission after transmutation to the fissile nuclide, the effect of buildup is significant. In this HTTR case, the buildup of actinoids occurs, but the fission of dominant TRU nuclides such as Pu-239 does not happen so much compared with the capture reaction. So, the net TRU destruction rate is low. This reactor has no favorable neutron flux for the TRU incineration.

Another problem is that the TRU destruction rate does not optimize the problem completely. For example, if the loading amount of TRU is minimal due to the optimal TRU destruction rate, it is not suitable for efficient TRU incineration. Solving a multi-objective optimization problem is necessary to obtain the practical optimal solution.

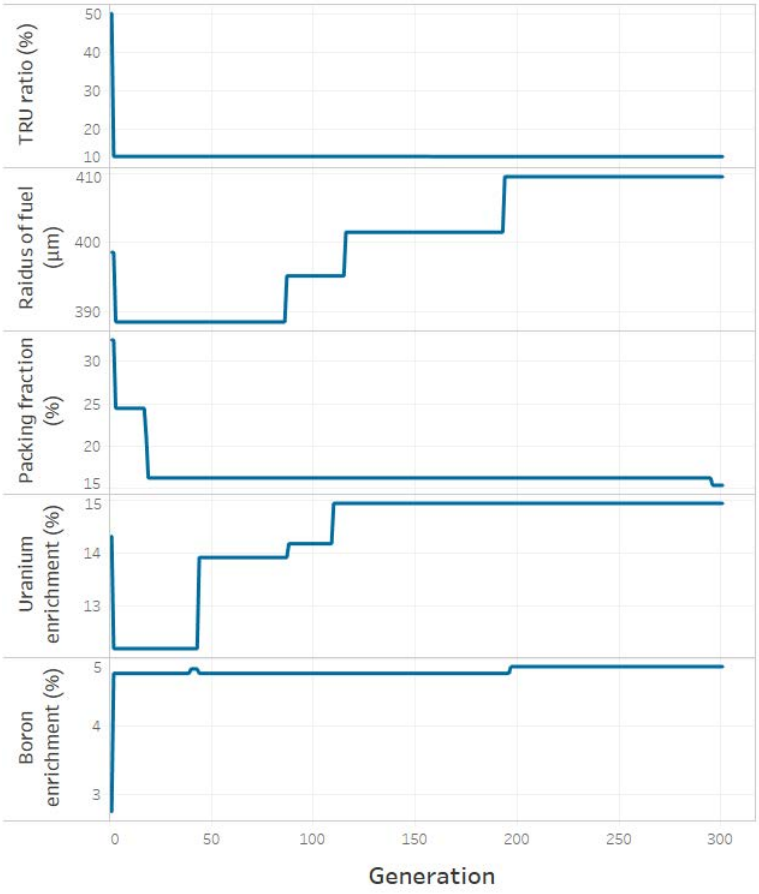


Figure 3.5: The change of design parameters through the genetic algorithm iteration.

Table 3.5: The optimized design parameters for the best TRU destruction rate and the best TRU destruction rate.

Parameter	Value	Unit
TRU ratio	13	%
Radius of fuel kernel	410	μ m
Packing Fraction	15	%
Uranium enrichment	15	%
Boron enrichment	4.9	%
TRU destruction rate	5.1	%

3.3 GT-HTR300 Optimization Problem

GT-HTR300 is developed from a successful experience with the HTTR. The previous section shows that HTTR is an existing experimental reactor and achieved criticality in 1998. GTHTR300 is an ongoing reactor concept, and detailed design configurations are published. GT-HTR300 is a prismatic block type and deploys pin-in-block type fuel. GT-HTR300 has an annular graphite-moderated core, and the remaining region is filled with reflector blocks as shown in Fig. 3.6 and 3.7.

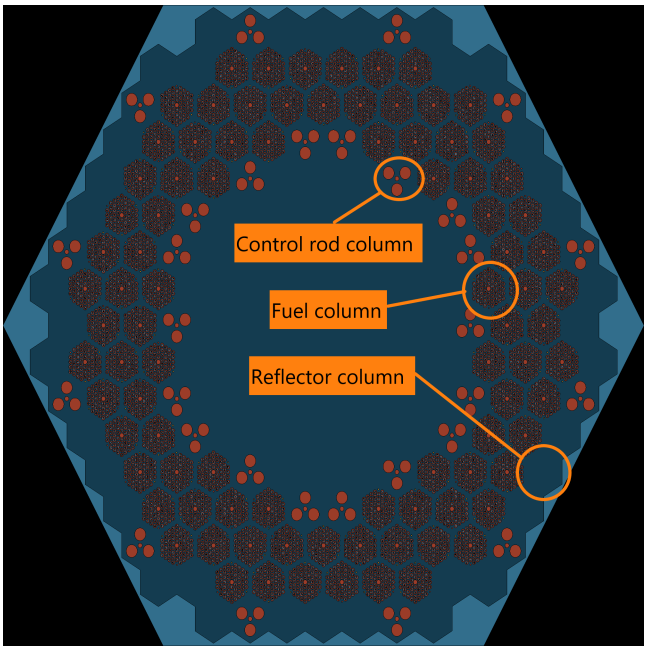


Figure 3.6: The horizontal cross-sectional view of GTHTR300 core.

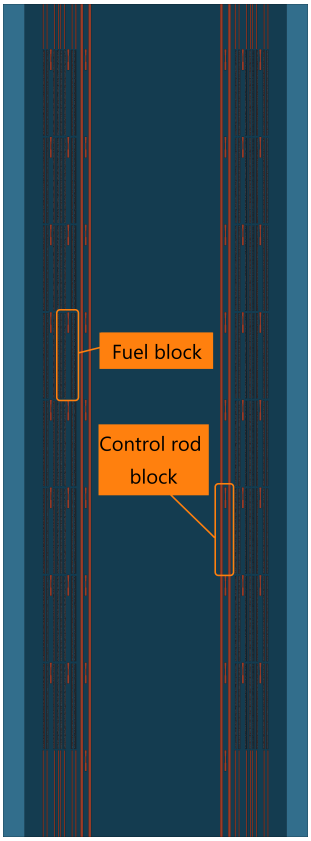


Figure 3.7: The axial cross-sectional view of GTHTR300 core.

The reactor core consists of 90 fuel columns arranged in an annular ring. 30 Control rod columns surround these fuel columns, and the remaining outer and inner parts are filled with removable reflector columns. As the characteristics of the gas-cooled graphite-moderated reactors, it has low power densities, resistance to the fission product release, and passive safety in the case of accidents.

3.3.1 GT-HTR300 Reactor Parameters

The fundamental design of GT-HTR300 is similar to the HTTR, but there are many updates for more sophisticated designs. For example, the fuel rod is integral and sleeveless for higher heat flux, and particle buffer is enlarged to retain fission products steadily. Also, fuel pins are more densely arranged. Multiple versions of GTHTR300 are published, so appropriate design parameters are picked from those publications[61][67][68] and the configuration of HTTR[63][64]. The primary design configurations of the reactor in this study are included in Table 3.6 and 3.7. The outlet

Table 3.6: The major design specification of the GTHTR300 model.

	GTHTR300
Thermal Power	600 MW
Outlet Coolant Temperature	850-950 ° C
Core Diameter	5.5 m
Core Height	8 m
Fuel	UO_2 (TRISO coated)
Enrichment	12 - 19 wt%
Type	Pin-in-Block
Block Material	Graphite
Coolant	Helium
Number of Fuel Blocks	720
Number of Fuel Columns	90
Number of Pairs of Control Rods	30

temperature is the same as the HTTR, but the output thermal power is much higher, and the size is much larger than the HTTR as a commercial reactor. The fuel is UO_2, but the TRU is added to

this problem. The number of fuel blocks, fuel columns, and control rods are similar to the number of representative GT-HTR300 designs, and the number of fuel blocks is significantly higher than the HTTR. Table 3.7 is the detailed geometry of the reactor core design. Each dimension is quite

Table 3.7: The detail specification of the GTHTR300 model.

	GTHTR300
Fuel Kernel Radius	150 - 425 μm
Buffer Layer Thickness	140 μm
Inner PyC Layer Thickness	25 μm
SiC Layer Thickness	40 μm
Outer PyC Layer	25 μm
Graphite Shaft Radius	4.5 mm
Fuel Compact Radius	1.2 cm
Fuel Compact Height	8.4 cm
Non-fueled Region Thickness of Compact	1.0 mm
Spacer Height	3.0 mm
Buffer Height	3.0 mm
Upper End Tap Height	9.0 mm
Lower End Tap Height	12.0 mm
Coolant Hole Radius	1.95 cm
Fuel Rod Height	105 cm
Fuel Block Height	105 cm
Fuel Pitch	4.7 cm
Fuel Block Pitch	41 cm

different from the HTTR. For example, the height of the fuel compact is more extended, and the fuel pitch is denser than that of HTTR.

The composition of TRU is based on the spent fuel from the AP1000 model simulation[65][66] as same as the HTTR case. The AP1000 is operated with the 3 wt% of Uranium oxide fuel for 630 days at a power of 3400MW. The TRU portion of the spent fuel from AP1000 takes 12 years before loading the GT-HTR300. However, Neptunium is extracted from the spent fuel because it makes the burnup period short, and the toxicity of Neptunium is not significant compared with other TRU nuclides. The PUREX process is assumed to be the method of Neptunium extraction. The mass

fraction of TRU composition is shown in Table 3.8. The primary component of the TRU is Pu-239

Table 3.8: TRU composition of the spent fuel from AP1000.

Nuclide	Mass fraction	Nuclide	Mass fraction
Pu-238	7.88E-03	Am-242	1.02E-05
Pu-239	5.28E-01	Am-243	5.45E-03
Pu-240	2.87E-01	Cm-243	9.47E-06
Pu-241	6.83E-02	Cm-244	8.62E-04
Pu-242	4.66E-02	Cm-245	4.74E-05
Pu-244	1.01E-06	Cm-246	3.26E-06
Am-241	5.52E-02		

and Pu-240. In the simulation model, these nuclides and Uranium are assumed to be homogeneous in the TRISO fuel.

The Energy spectrum in the fuel region one of the GT-HTR300 with 20% of TRU ratio is shown in Fig. 3.8. The design parameters for this spectrum are 20% of TRU ratio, 287μm of fuel kernel radius, 20% of packing fraction, 12% of Uranium enrichment, and 3% of boron enrichment. Compared with the spectrum of HTR with the UO_2 fuel, the peak of the thermal region is low because some TRU nuclides such as Am-241, Pu-240, and Pu-242 have high absorption capability of the thermal neutron. The overall shape of the energy spectrum is similar to the case of HTTR.

The effective neutron multiplication factor change through the depletion time is shown in Fig. 3.9. The neutron multiplication factor starts around 1.08 at the BOC. The neutron multiplication factor steadily decreases after the dip at the initial depletion period. The core becomes subcritical around 1200 days from the start date. The error bar is based on the 3σ of statistical errors. The design parameters of this depletion are 85% of TRU ratio, 405μm of fuel kernel radius, 18.7% of packing fraction, 12.7 - 18.3% of Uranium enrichment, and 4.6% of boron enrichment.

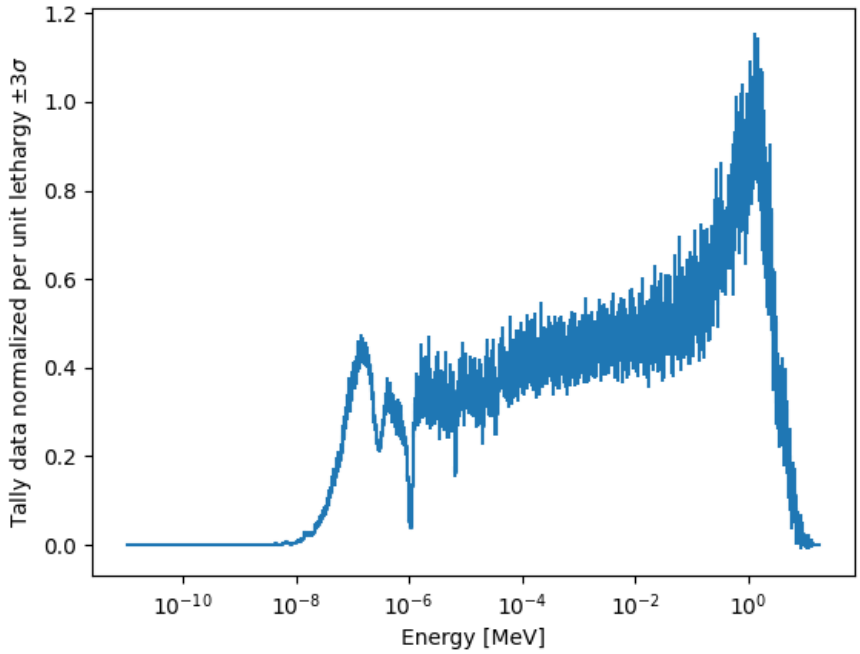


Figure 3.8: The energy spectrum in the fuel region of the GT-HTR300 with 20% of TRU ratio.

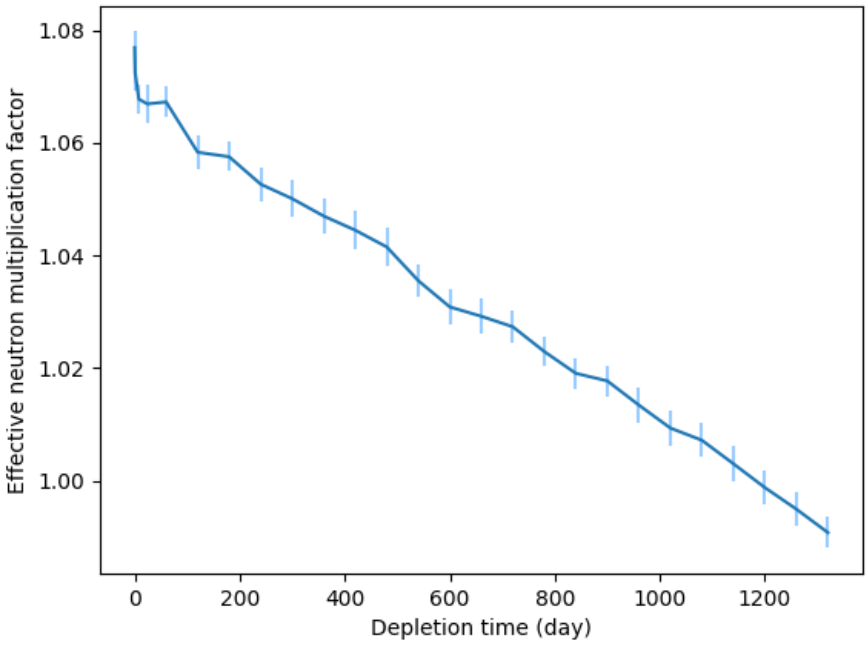


Figure 3.9: The effective neutron multiplication factor through the depletion of the GT-HTR300.

3.3.2 Optimization Target

The primary optimization target is the same as the case of HTTR. The list of design parameters is shown with the ranges of values in Table 3.9. The range of packing fraction and uranium

Table 3.9: The target design parameters and their range of values for the GTHTR300.

Design parameters	Value ranges
TRU ratio	10 - 90 %
Radius of fuel kernel	150 - 425 μ m
Packing fraction	15 - 35 %
Uranium enrichment in region 1	12 - 19 %
Uranium enrichment in region 2	12 - 19 %
Uranium enrichment in region 3	12 - 19 %
Uranium enrichment in region 4	12 - 19 %
Boron enrichment	1 - 5 %

enrichment is changed for the GT-HTR300. The upper boundary of the packing fraction decreases to 35%, and the bottom and top boundary of Uranium enrichment increase 12 % and 19 %, respectively. This is because the required enrichment of Uranium is higher in the GT-HTR300 core compared with HTTR. Also, the definition of boron enrichment is changed. Here, boron enrichment is the weight percentage of the B_4C in the B_4C-C complex for the burnable poison, and the boron composition in the B_4C is set natural composition. The Uranium enrichment of the Uranium portion of fuel kernels is set differently depending on the reactor core region. The four regions of different Uranium enrichment are set as shown in Fig. 3.10 and 3.11. Region one is the bottom half of the inner reactor core, region two is the bottom half of the outer reactor core, region three is the top half of the inner reactor core, and region four is the top half of the outer reactor core. The reactor core usually has heterogeneous Uranium enrichment for flattening the neutron spectrum.

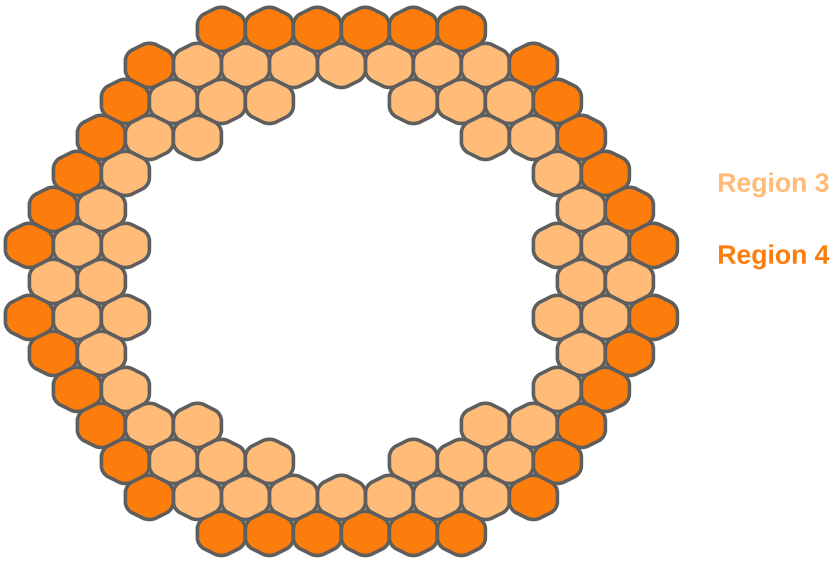


Figure 3.10: The horizontal cross-sectional view of the GTHTR300 region.

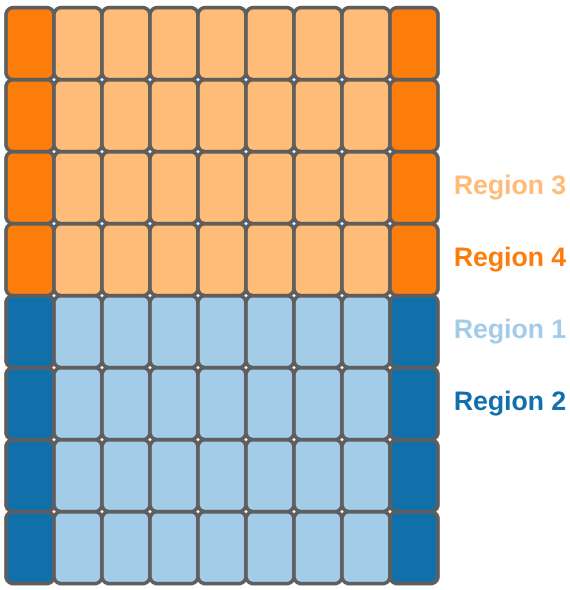


Figure 3.11: The axial cross-sectional view of the GTHTR300 region.

3.3.3 Objective Function

The objective funcitons to evaluate the minimization is set destruction rate of the TRU nuclides i, D_i, and the mass of TRU and U-235 nuclides at the BOC. The objective functions and constraints for waste minimization are defined as:

$$\begin{aligned} \max.\ \mathrm{D_i} &= 1 - \frac{m_{i,EOC}}{m_{i,BOC}}, i = 1, ..., N && (3.4) \\ \max.\ \mathrm{m_{TRU,BOC}} && (3.5) \\ \min.\ \mathrm{m_{U-235,BOC}} && (3.6) \\ k_{\mathrm{eff,BOC}} &\geq 1.0 && (3.7) \\ \rho(t) &< 0. && (3.8) \end{aligned}$$

where $m_{i,EOC}$ is the total mass of TRU nuclide i at the EOC, $m_{i,BOC}$ is the total mass of TRU nuclide i at the BOC, and N is the total number of target TRU nuclides. $m_{TRU,BOC}$ is the total mass of TRU at the BOC. $m_{U-235,BOC}$ is the total mass of TRU at the BOC. $k_{\mathrm{eff,BOC}}$ is effective neutron multiplication factor at the BOC. $\rho(t)$ is a reactivity coefficient.

The destruction rates of TRU nuclides are individually evaluated instead of the total destruction rate, and it can model a surrogate model more accurately and consider the toxicity characteristics of each TRU nuclides. The total mass of TRU with TRU destruction rate can evaluate how much TRU nuclides can be destroyed in a cycle. The mass of U-235 can evaluate how efficient the expensive U-235 is used for the TRU incineration. The high enrichment of U-235 and low burnup with a large U-235 is not economically practical. These objective functions are trade-offs so that the search for optimal points becomes more complicated than that without U-235 and TRU objectives.

3.3.4 Correlation Check and Sensitivity Analysis

The relationship between design parameters and objective values is investigated to see trends. Fig. 3.12 - 3.25 show the total TRU destruction rate, each TRU nuclide's destruction rate, U-235, and TRU at the BOC for the different combinations of design parameters.

For each figure, one design parameter is changed in its design range, and the other design parameters are fixed at the specific values. The fixed values for this analysis are shown in Table 3.10. The fixed point is set at the median point of the range except for the Uranium enrichment. The

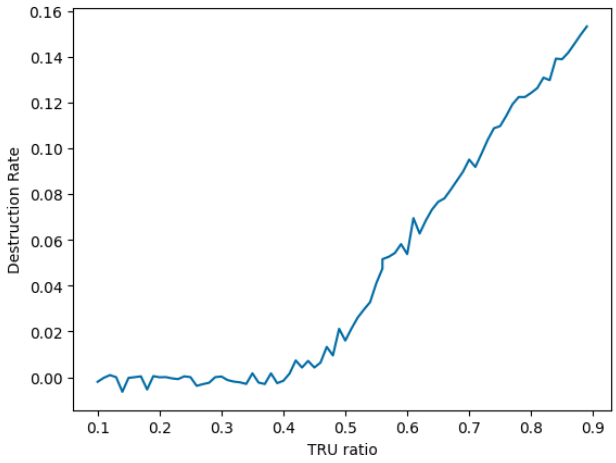
Table 3.10: The default design parameters for the sensitivity analysis.

	Fixed Value
TRU ratio	50 %
Radius of fuel kernel	287 μ m
Packing fraction	20 %
Uranium enrichment in region 1	12 %
Uranium enrichment in region 2	12 %
Uranium enrichment in region 3	12 %
Uranium enrichment in region 4	12 %
Boron enrichment	3 %

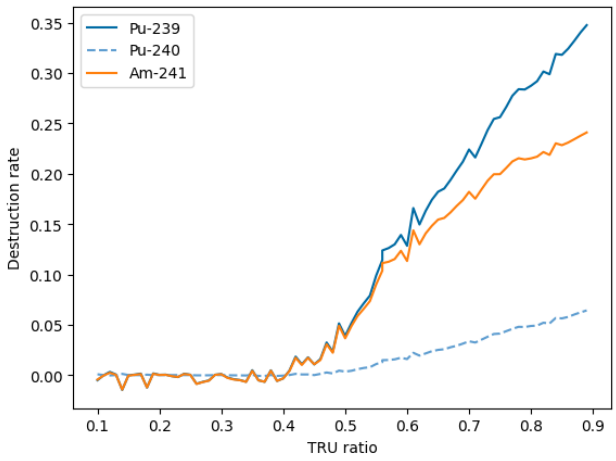
fixed values of each Uranium enrichment are 12%, which is lower than the middle point.

The range of TRU ratio between 0.1 to 0.4 shows a shallow destruction rate in Fig. 3.12. It means this combination of design parameters cannot keep the criticality long, and the burnup is very short. In a higher TRU ratio, some fissile nuclides included in the TRU can help the criticality, and the depletion can continue enough time to destroy the TRU nuclides. The higher TRU ratio increases the destruction rate because of the higher burnup. Even though some non-fissile nuclides can be destroyed eventually after a couple of neutron capture reactions and decays, there is much more buildup due to the neutron capture reactions of lower actinides.

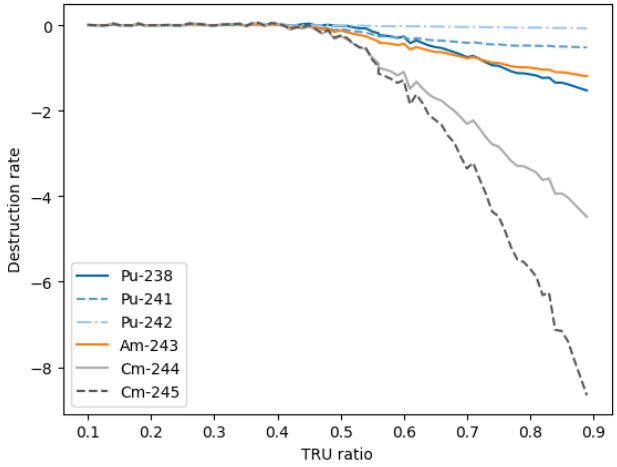
The balance of fission and capture reactions determines whether the destruction rate is positive or negative. For example, Am-241 is not a fissile nuclide, but their capture resonance peak is very high, so the destruction rate is positive. On the other hand, Pu-241 is a fissile nuclide, but the capture cross-section of the Pu-240 is very high, so the net destruction rate is slightly negative. Some nuclides with negative destruction rates such as Cm-243 show shallow values compared with the positive destruction rate of Pu-239(-14.0 vs. 0.35 at 90% of the TRU ratio). However, the fuel's



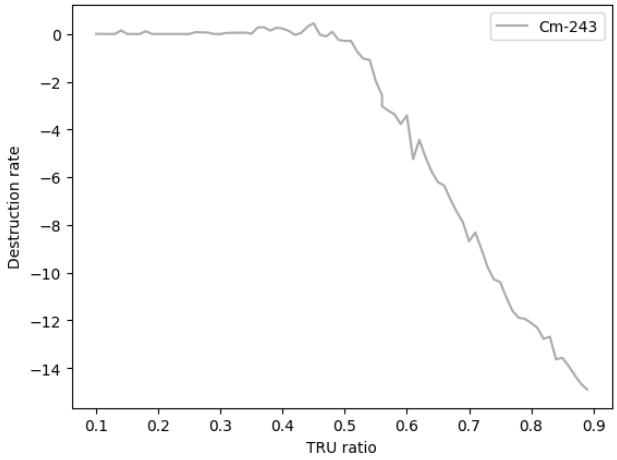
(a) Destruction rate of TRU.



(b) Destruction rate of decreased nuclides.



(c) Destruction rate of increased nuclides.

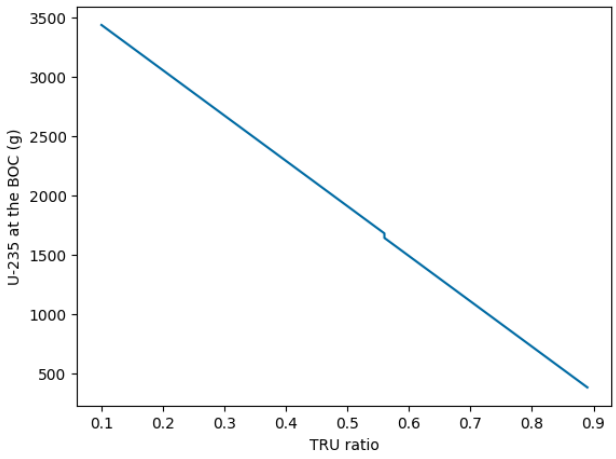


(d) Destruction rate of increased nuclides.

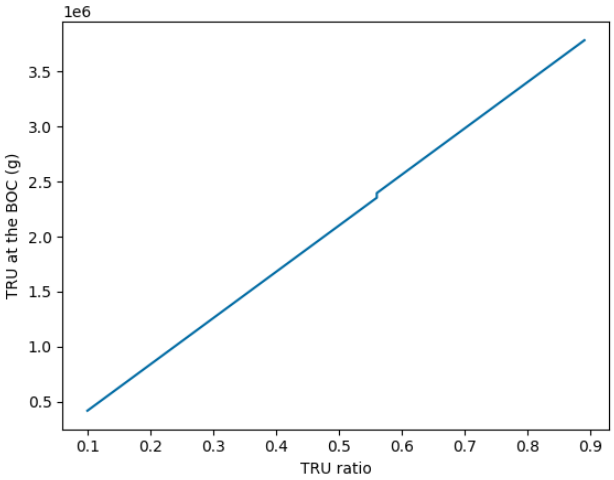
Figure 3.12: The trend of destruction rates depending on the TRU ratio.

initial inventory of these nuclides is minimal compared with Pu-239. Hence, the net destruction rate can be positive. When the TRU ratio increases, the portion of uranium fuel decreases. So, the mass of U-235 also decreases. A higher TRU ratio looks preferable if it can meet the constraints from these characteristics.

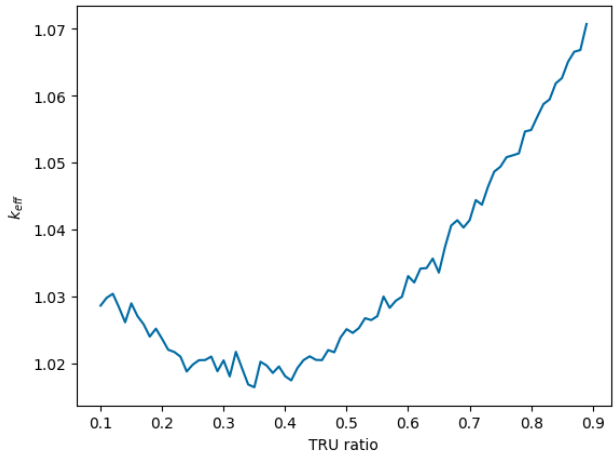
Fig. 3.13 shows that the depletion period steadily increases when the TRU ratio rises, and the low TRU ratio cannot keep the criticality in this combination of design parameters. For the criticality, higher Uranium enrichment is necessary, for example.



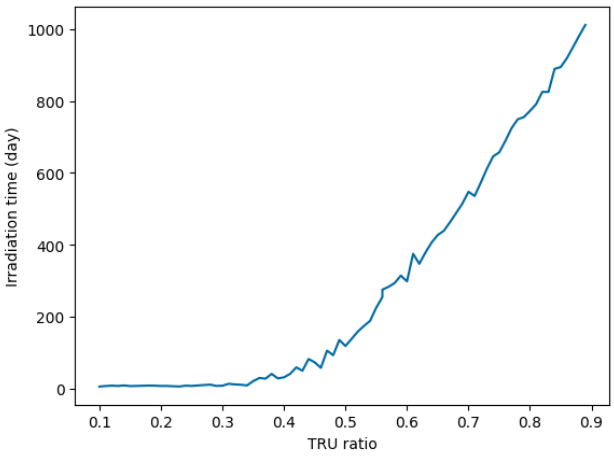
(a) Loaded U-235 at the BOC.



(b) Loaded TRU at the BOC.

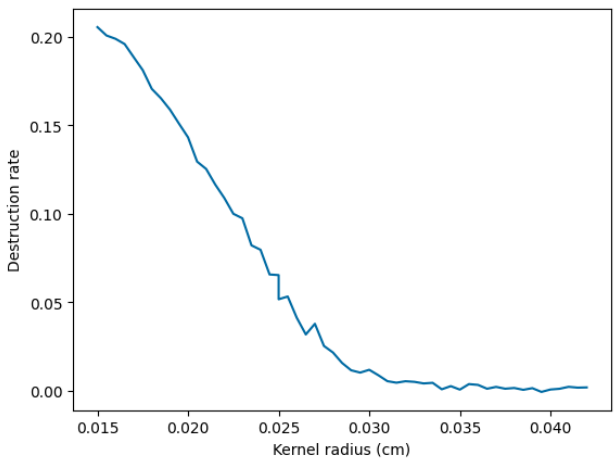


(c) Neutron multiplication factor at the BOC.

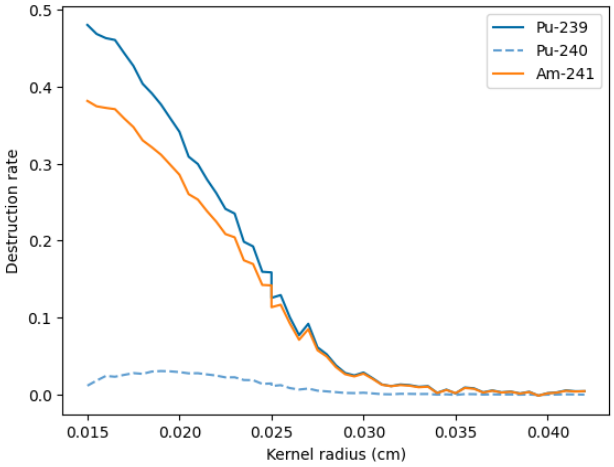


(d) Irradiation time.

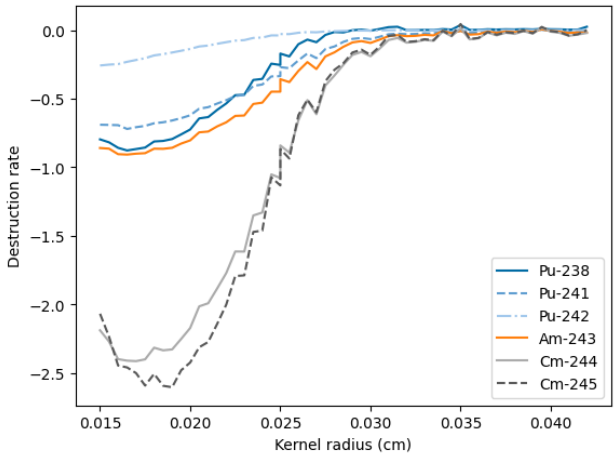
Figure 3.13: The trend of objective values and important values depending on the TRU ratio.



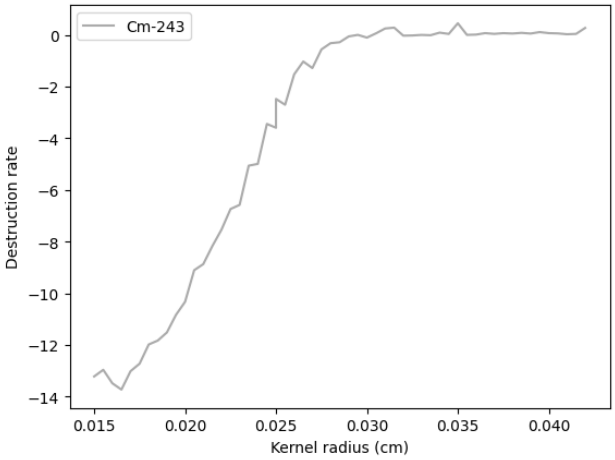
(a) Destruction rate of TRU.



(b) Destruction rate of decreased nuclides.



(c) Destruction rate of increased nuclides.

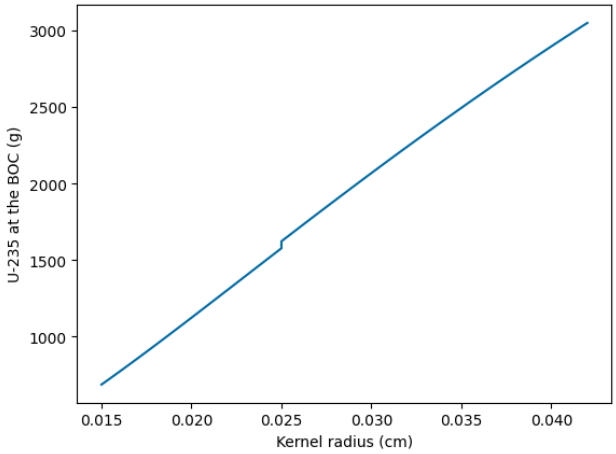


(d) Destruction rate of increased nuclides.

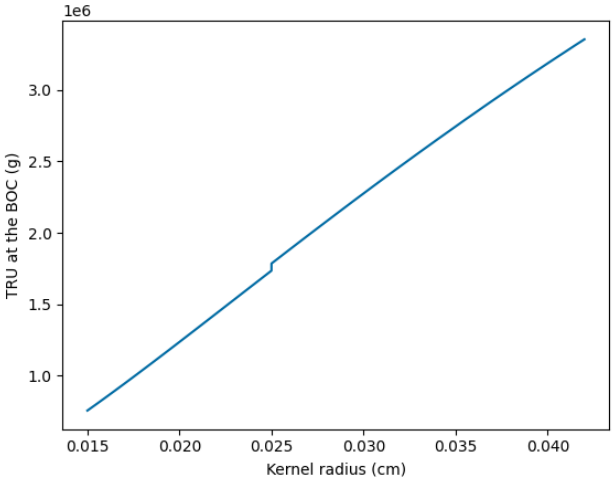
Figure 3.14: The trend of destruction rates depending on the fuel kernel radius.

Fig. 3.14 shows more interesting trends regarding kernel radius. In the trends of kernel radius, a longer radius has a subcriticality in this combination of the design parameters. The depletion can only continue short, and the TRU destruction rate becomes close to zero. The performance of the TRU destruction rate is good for the small fuel kernel. The destruction rate of decreased nuclides such as Pu-239 and Am-241 also shows similar trends. Regarding the destruction rates of nuclides which have negative destruction rates, their destruction rates increase as the kernel radius decreases, but they start to increase again around less than 220 μm of the kernel radius.

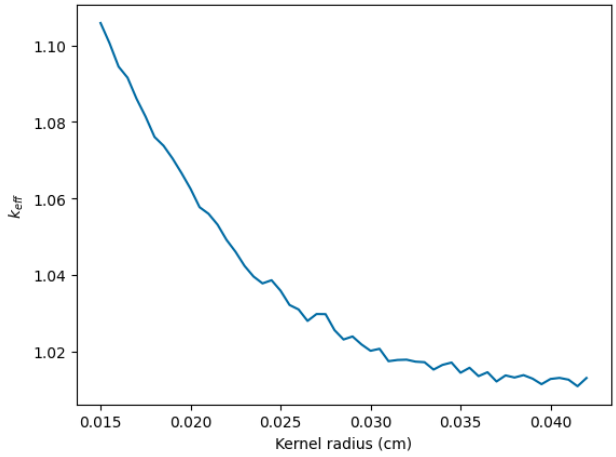
The fuel kernel size can significantly affect the neutronics behavior because of the moderation



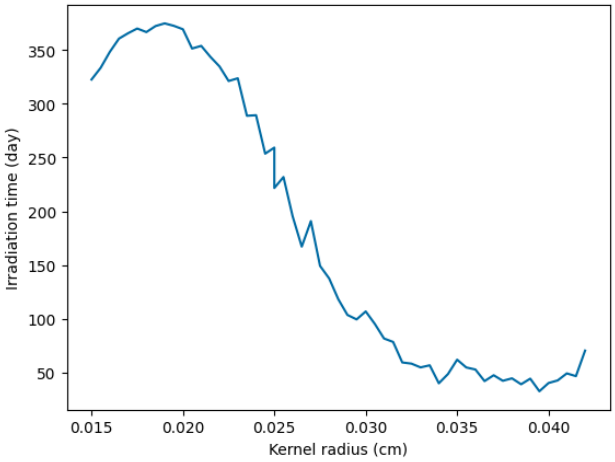
(a) Loaded U-235 at the BOC.



(b) Loaded TRU at the BOC.



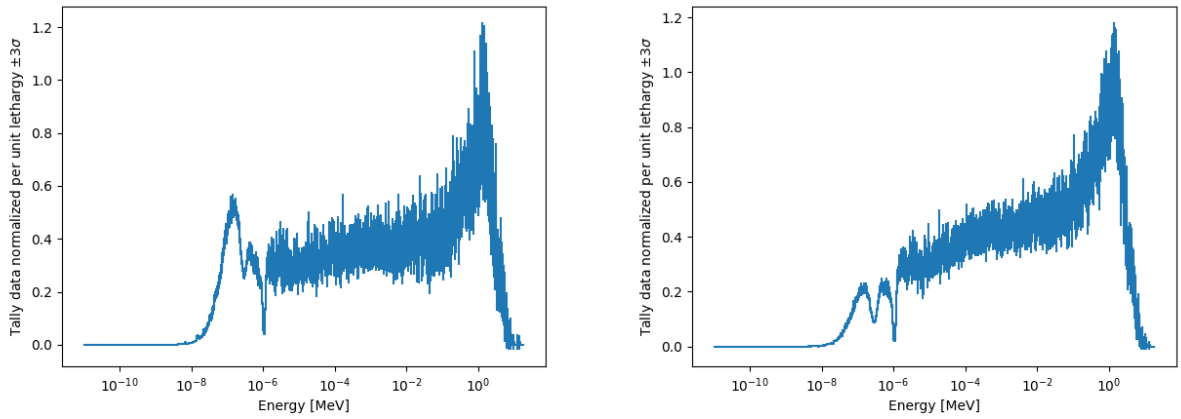
(c) Neutron multiplication factor at the BOC.



(d) Irradiation time.

Figure 3.15: The trend of objective values and important values depending on the fuel kernel radius.

and resonance absorption at the thermal spectrum. When the kernel size becomes smaller, the graphite moderates neutrons more, and the neutron spectrum becomes softer. The energy spectrum in the case of 150 μm and 260 μm of kernel radius are shown in Fig. 3.16. There is a higher peak of around 0.1eV of neutron energy in the case of 150 μm fuel kernel radius. For the fissile nuclides, The fission cross-section at the thermal energy spectrum is relatively high in resonance absorption, so their destruction rates improve. However, most of the capture cross-sections of both fissile and non-fissile nuclides are also high at the resonance peak, leading to the higher actinides' buildup.



(a) 150 μ m fuel kernel radius.

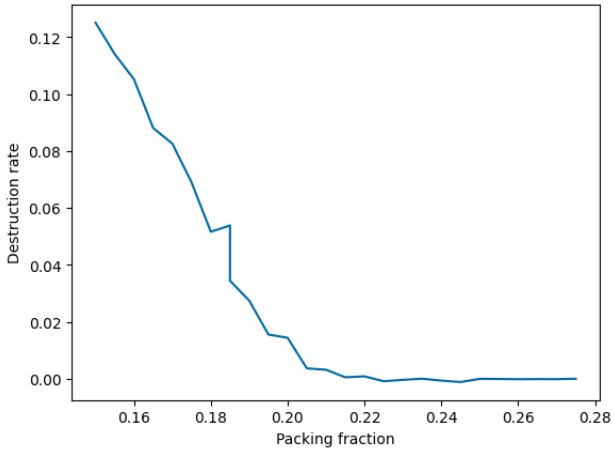
(b) 260 μ m fuel kernel radius.

Figure 3.16: The energy spectrum of GTHTR-300 in the case of 150 μm and 260 μm fuel kernel radius.

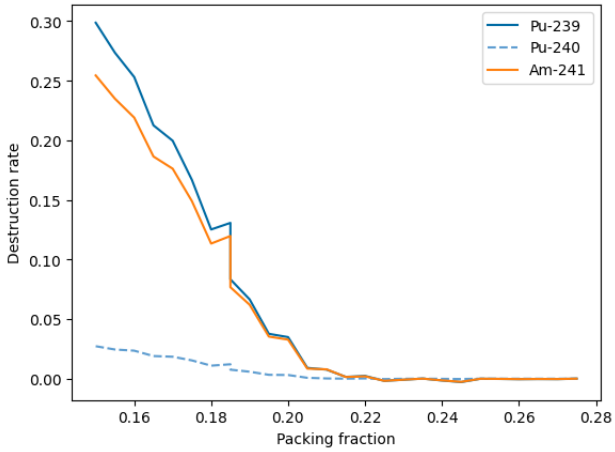
Some part of Pu-239 is transmuted to the higher actinides. Since the amount of Pu-239 is much higher than other TRU nuclides, the effect of this buildup looks large in the rate.

On the other hand, a large fuel kernel has some advantages. The large kernel enhances self-shielding and minimizes the fissile nuclides' capture/fission reaction value. The fissile nuclides are used more efficiently to keep the criticality, increasing the total burnup. As for the total amount of TRU, the small fuel kernel size is a trade-off because the amount of loaded TRU at the BOC decrease in the smaller fuel kernel size. This trend shows the difficulty of finding the optimal point between one design parameter and the outputs.

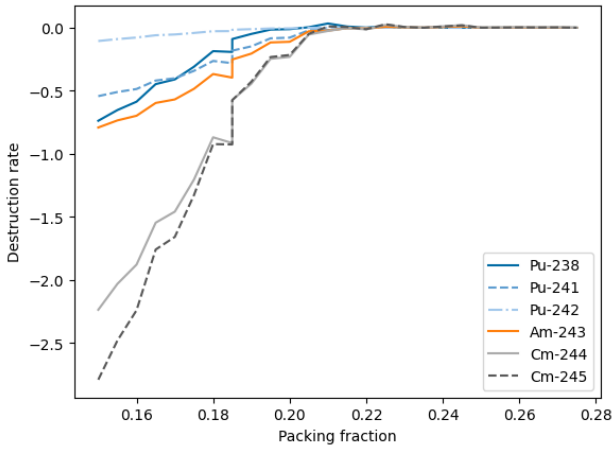
Fig. 3.15 shows the total irradiation time is decreased at 0.015 cm of the fuel kernel radius. This explains that the minimal fuel kernel is unsuitable for the capture/fission ratio, and the fissile nuclides are not efficiently used. Because of the shorter irradiation time, the trend of destruction rate changes at the 0.015 cm of fuel kernel radius.



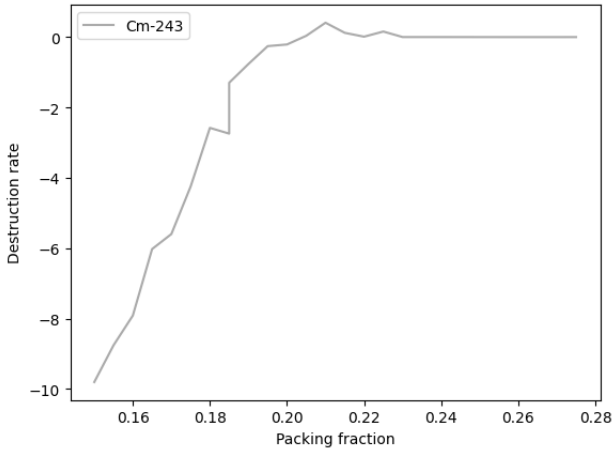
(a) Destruction rate of TRU.



(b) Destruction rate of decreased nuclides.



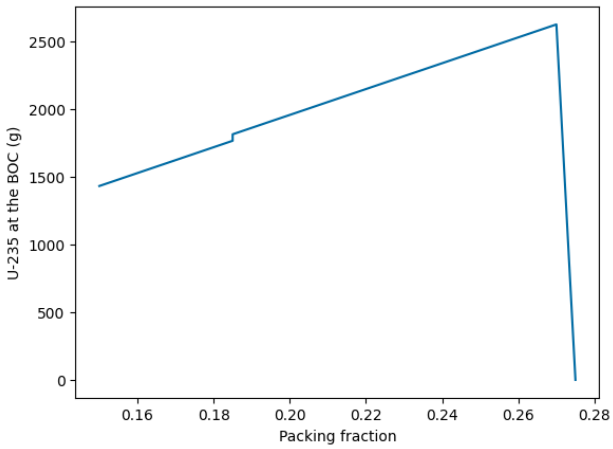
(c) Destruction rate of increased nuclides.



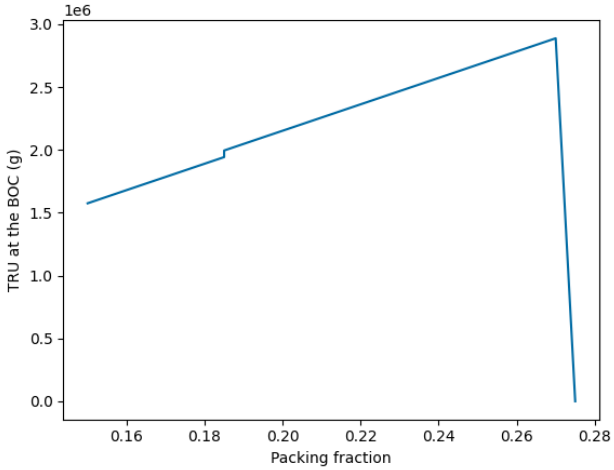
(d) Destruction rate of increased nuclides.

Figure 3.17: The trend of destruction rates depending on the packing fraction.

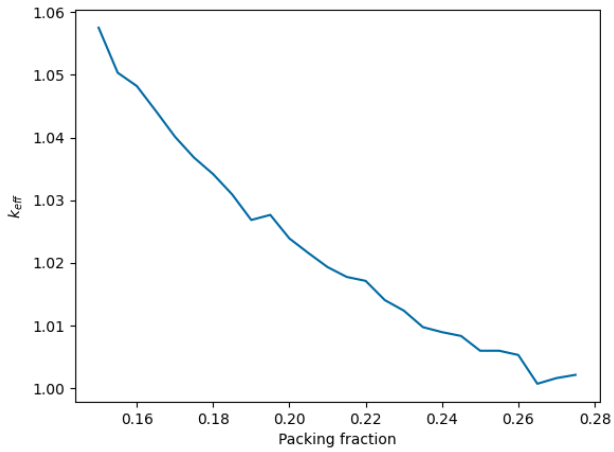
In the case of the packing fraction trend, as shown in Fig. 3.17, the reactor is entirely subcritical at a point larger than 27.5 % of the packing fraction. Hence, all parameters are evaluated as zero over that point. Packing fraction has a similar neutronics impact with kernel radius, and higher packing fraction leads to the under-moderation of the neutron, while lower packing fraction can be over-moderation. Hence, the capability of TRU destruction regarding neutronics can be determined by balancing these two design parameters. This figure also shows that the feasible region of a design parameter is a very narrow range depending on the other combinations of the design parameters.



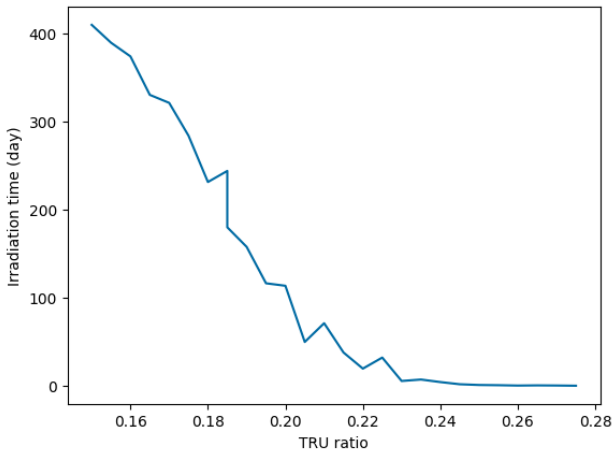
(a) Loaded U-235 at the BOC.



(b) Loaded TRU at the BOC.

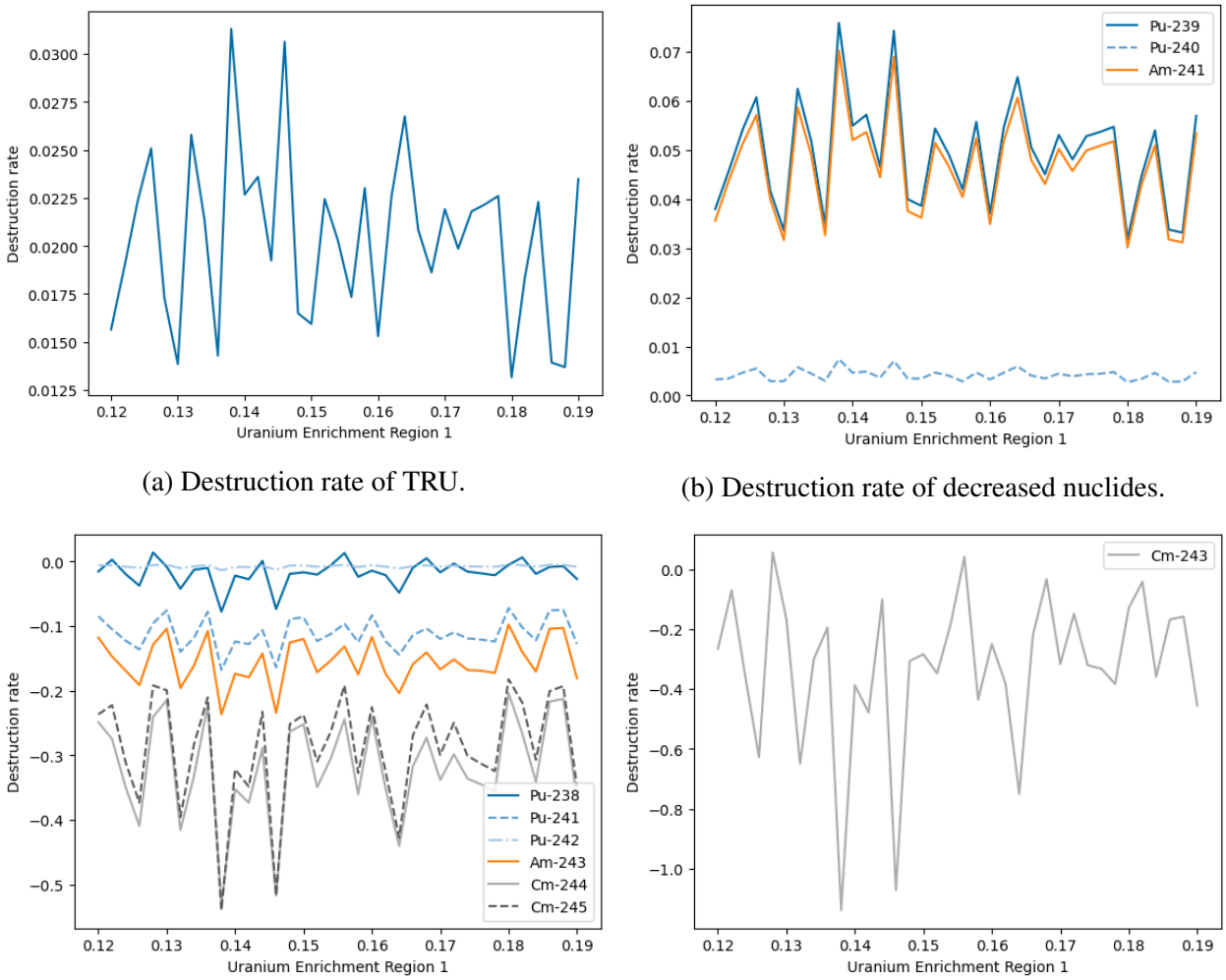


(c) Neutron multiplication factor at the BOC.



(d) Irradiation time.

Figure 3.18: The trend of objective values and important values depending on the packing fraction.



(a) Destruction rate of TRU.

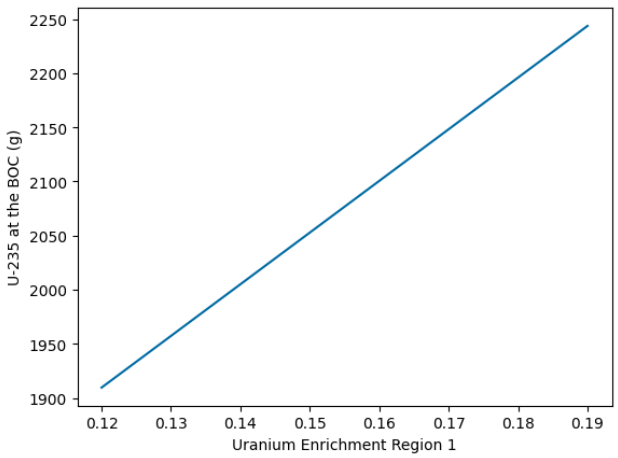
(b) Destruction rate of decreased nuclides.

(c) Destruction rate of increased nuclides.

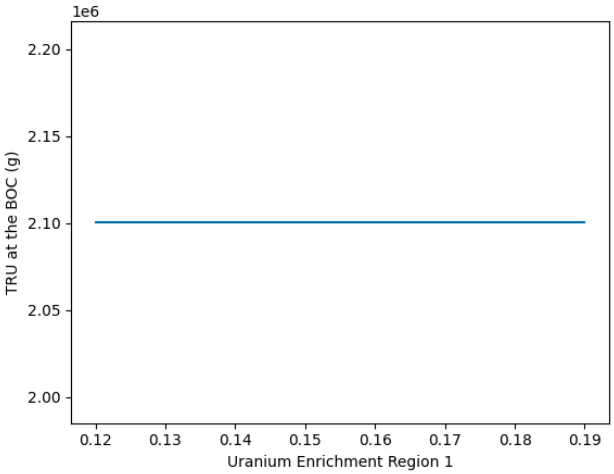
(d) Destruction rate of increased nuclides.

Figure 3.19: The trend of destruction rates depending on the Uranium enrichment at region 1.

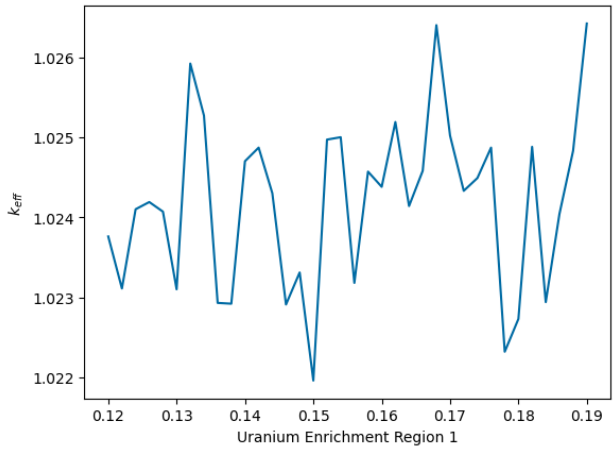
Fig. 3.19 - 3.23 show the trend of Uranium enrichment in each region. No clear trend can be observed from the figures with statistical noise, but the uranium enrichment directly affects the longer burnup. So, depending on the other design parameters, Uranium enrichment can improve the destruction rate. However, the higher enrichment of Uranium requires a more extensive amount of U-235 and also high enrichment of High-Assay Low-Enriched Uranium (HALEU) takes cost in terms of the production and proliferation concern. That makes another trade-off, and the optimization problem makes it more difficult.



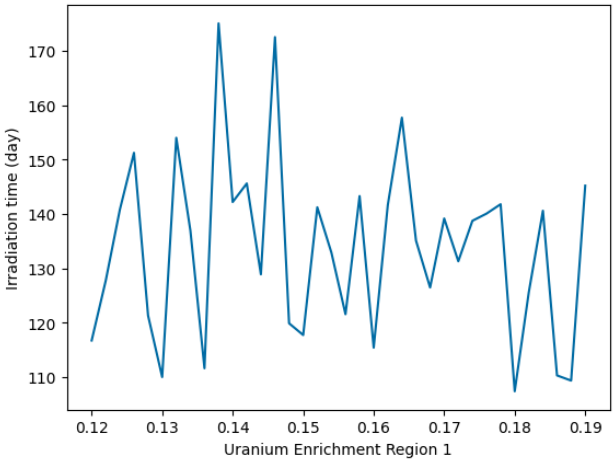
(a) Loaded U-235 at the BOC.



(b) Loaded TRU at the BOC.

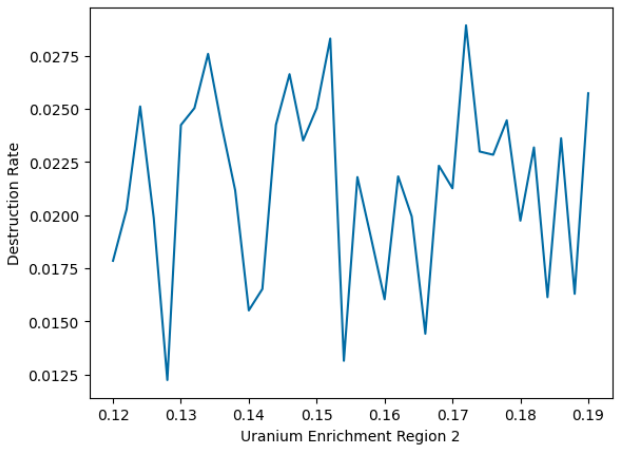


(c) Neutron multiplication factor at the BOC.

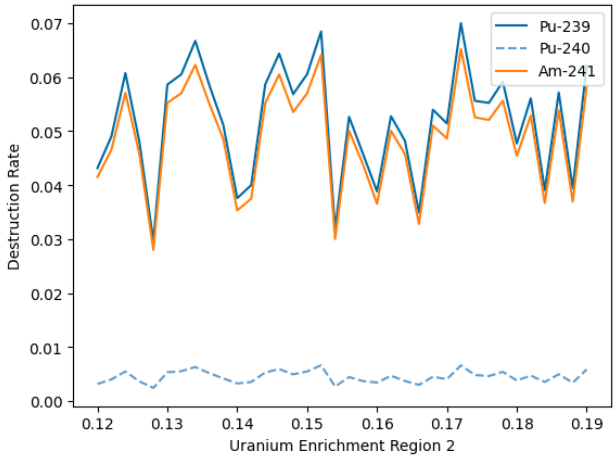


(d) Irradiation time.

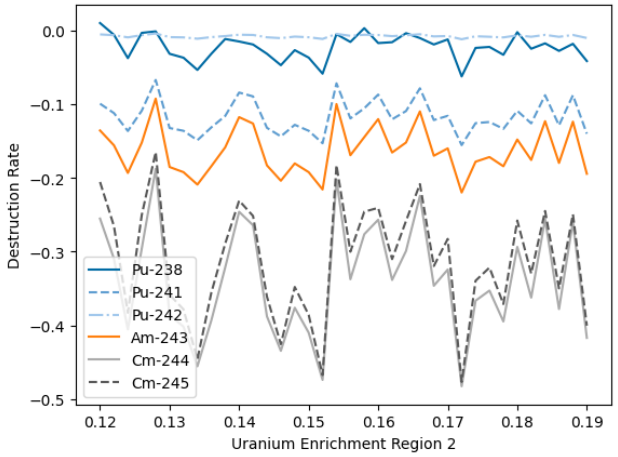
Figure 3.20: The trend of objective values and important values depending on the Uranium enrichment at region 1.



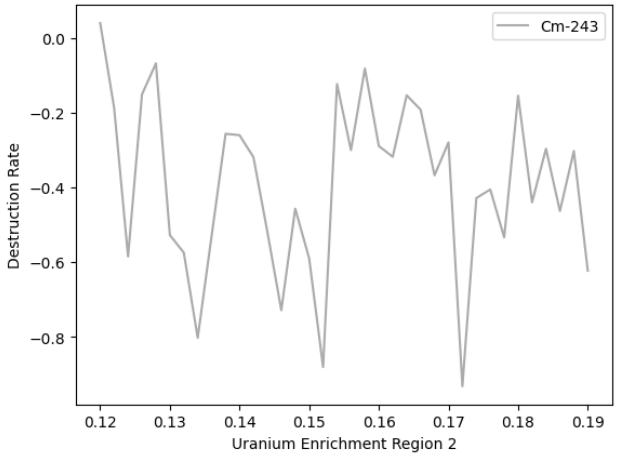
(a) Destruction rate of TRU.



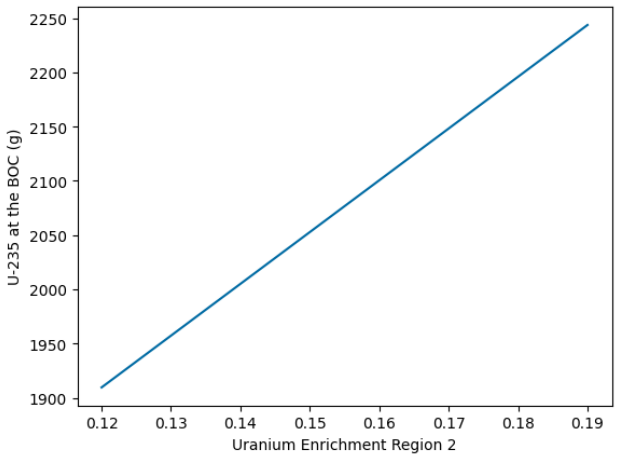
(b) Destruction rate of decreased nuclides.



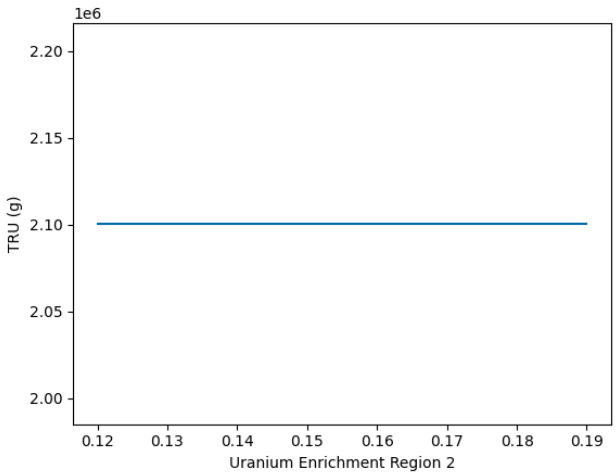
(c) Destruction rate of increased nuclides.



(d) Destruction rate of increased nuclides.

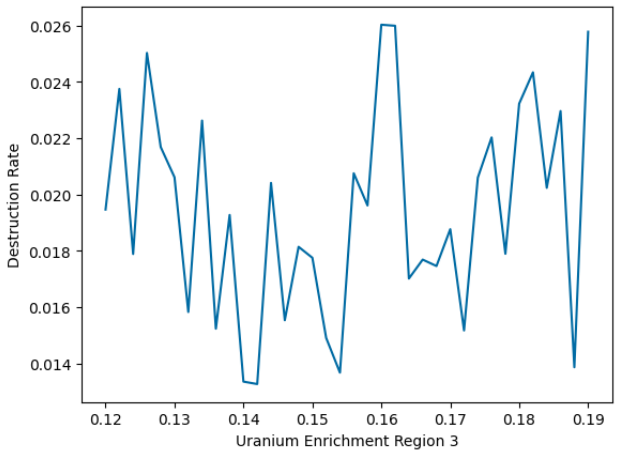


(e) Loaded U-235 at the BOC.

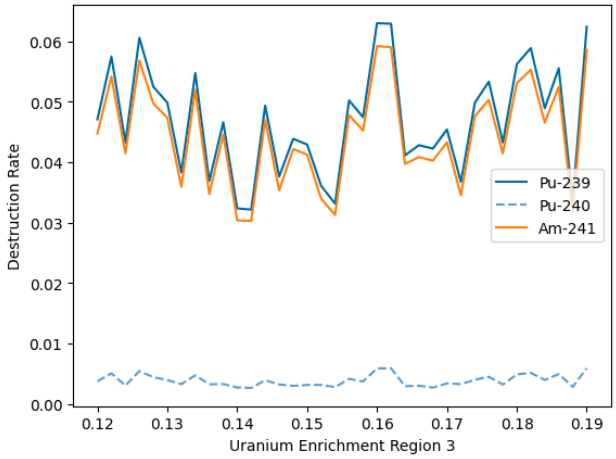


(f) Loaded TRU at the BOC.

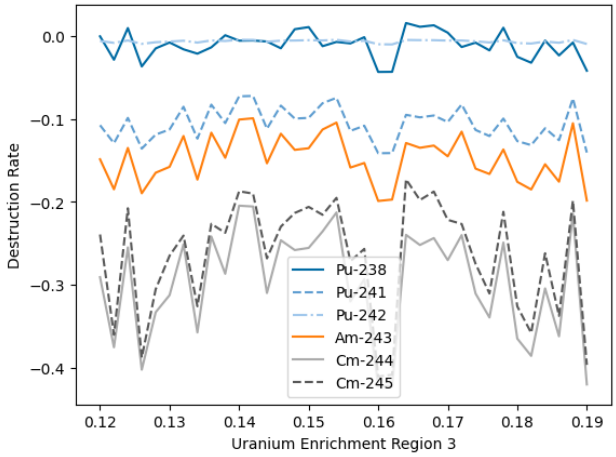
Figure 3.21: The trend of objective values depending on the Uranium enrichment at region 2.



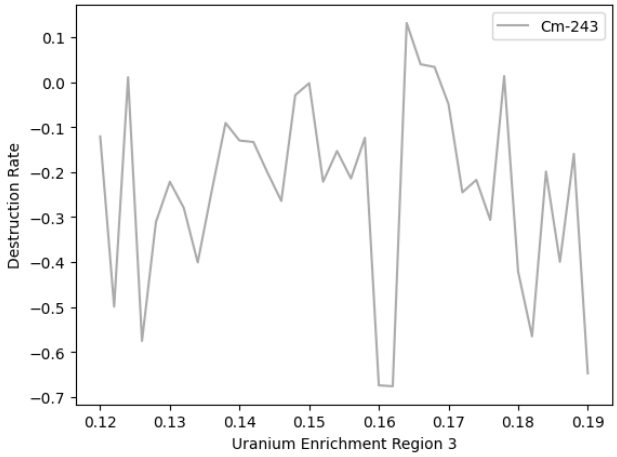
(a) Destruction rate of TRU.



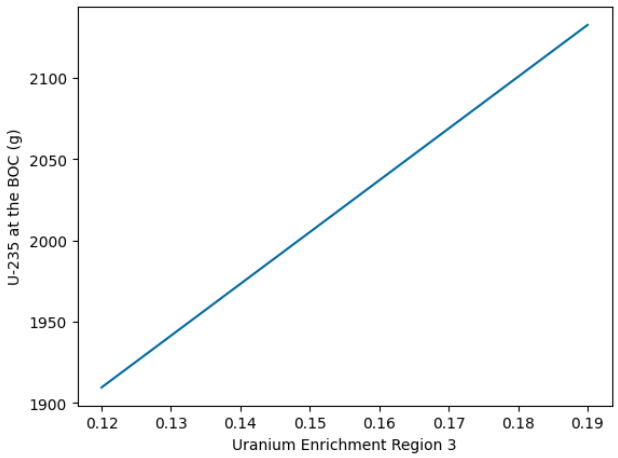
(b) Destruction rate of decreased nuclides.



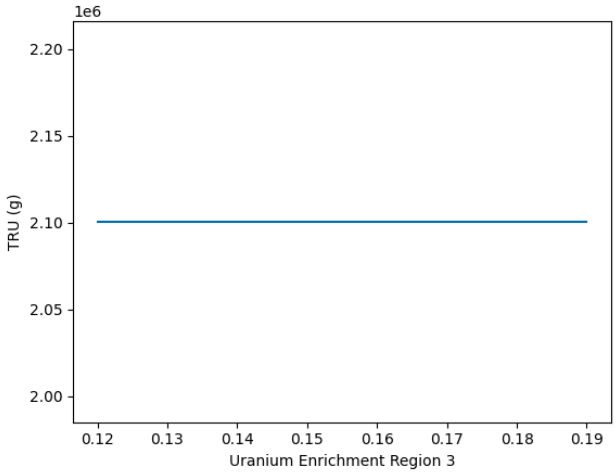
(c) Destruction rate of increased nuclides.



(d) Destruction rate of increased nuclides.

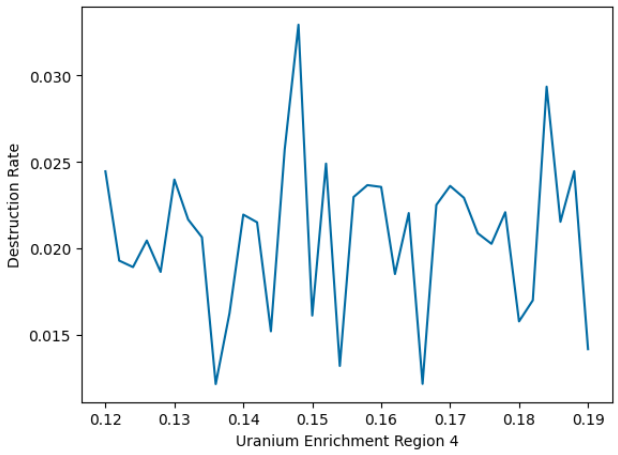


(e) Loaded U-235 at the BOC.

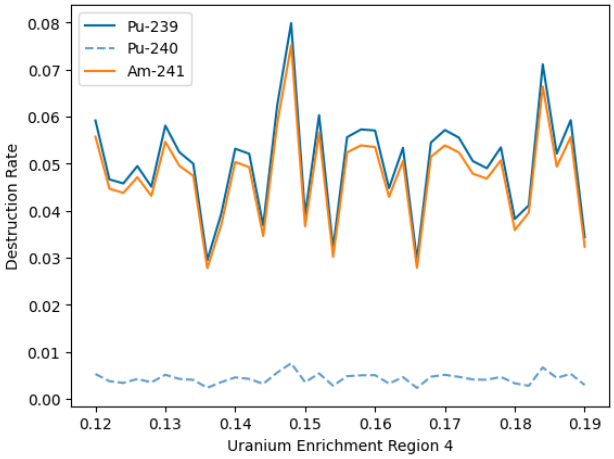


(f) Loaded TRU at the BOC.

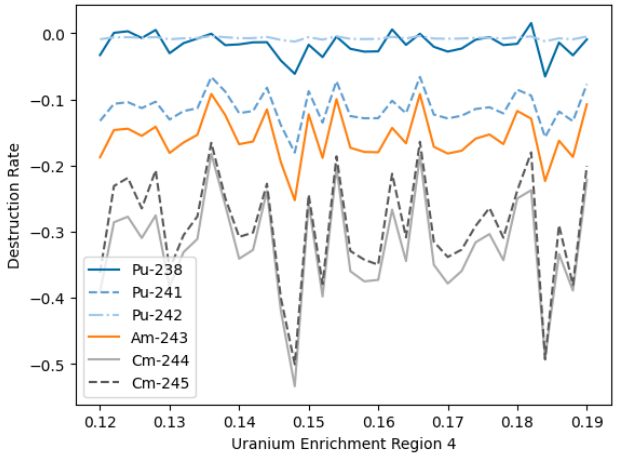
Figure 3.22: The trend of objective values depending on the Uranium enrichment at region 3.



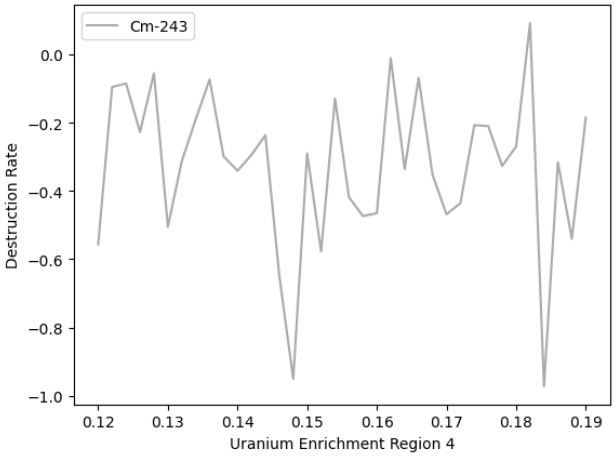
(a) Destruction rate of TRU.



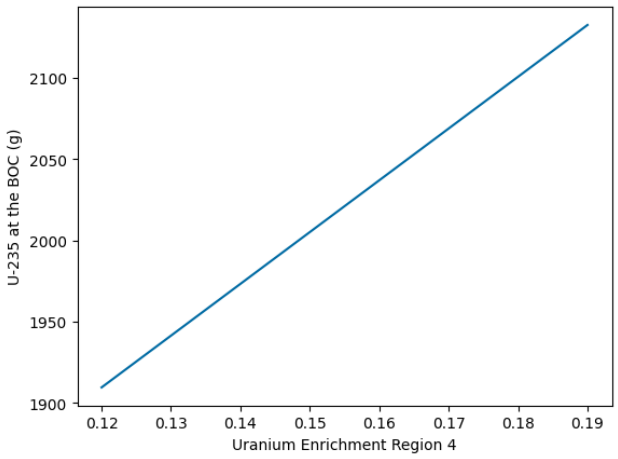
(b) Destruction rate of decreased nuclides.



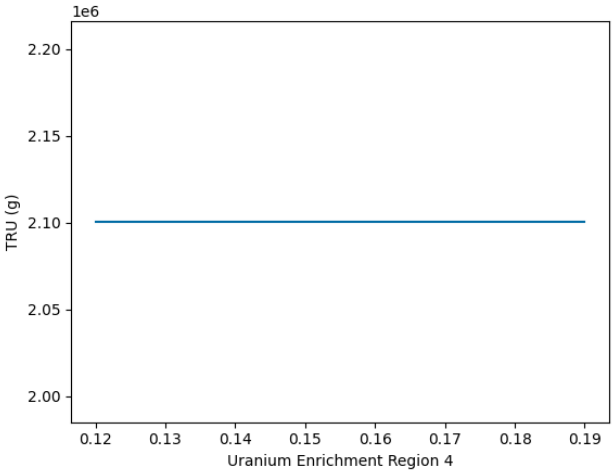
(c) Destruction rate of increased nuclides.



(d) Destruction rate of increased nuclides.

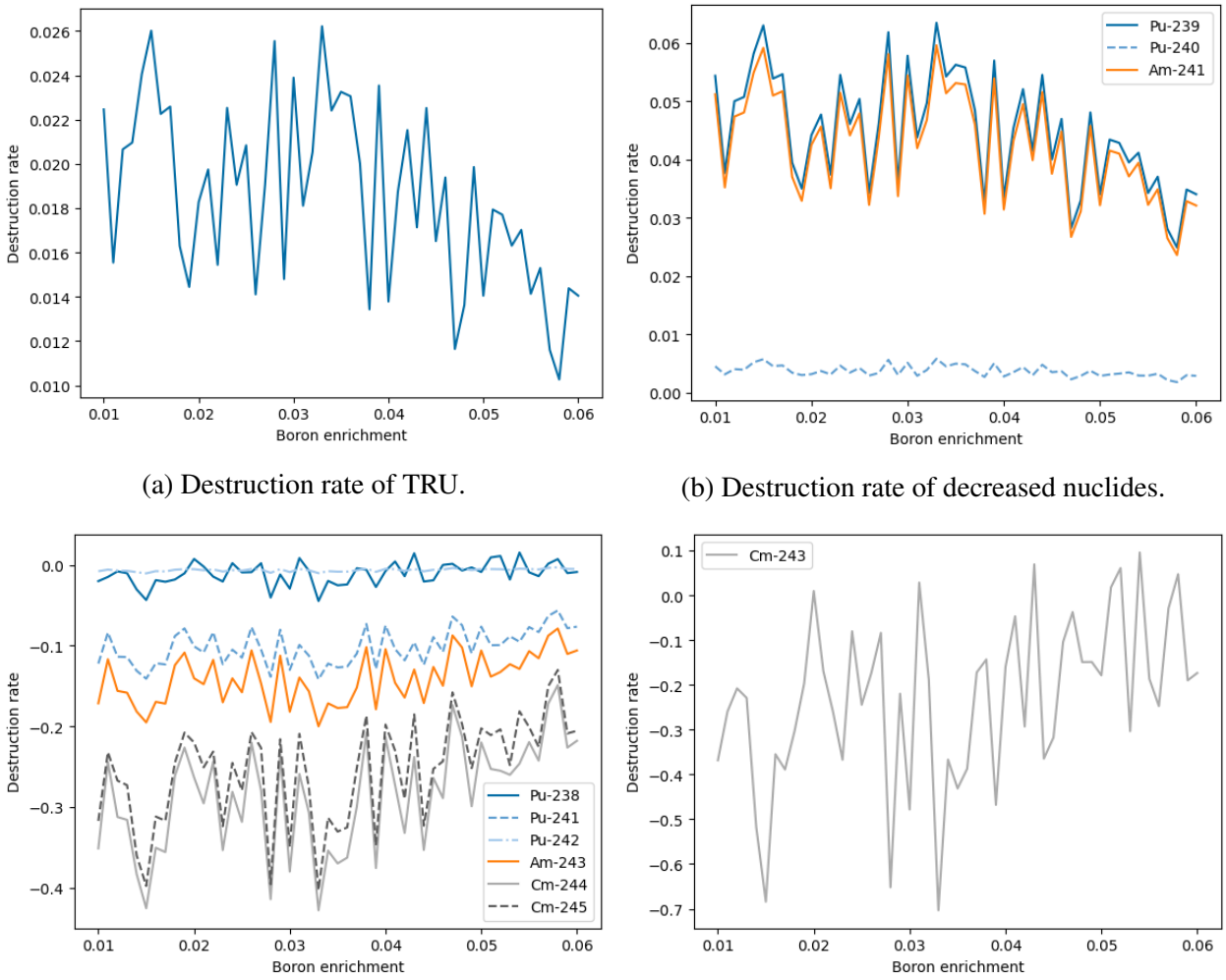


(e) Loaded U-235 at the BOC.



(f) Loaded TRU at the BOC.

Figure 3.23: The trend of objective values depending on the Uranium enrichment at region 4.



(a) Destruction rate of TRU.

(b) Destruction rate of decreased nuclides.

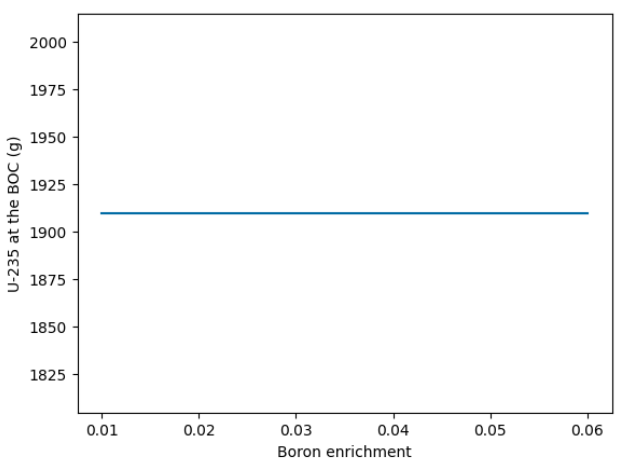
(c) Destruction rate of increased nuclides.

(d) Destruction rate of increased nuclides.

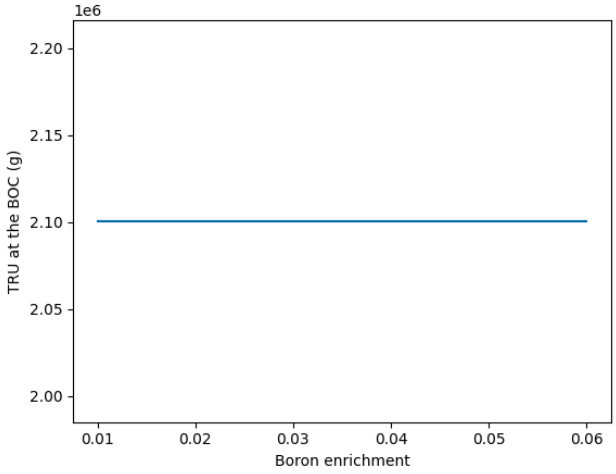
Figure 3.24: The trend of destruction rates depending on the boron enrichment.

Fig. 3.24 and 3.25 show the trend between burnable poison and destruction rate. It also includes sizeable statistical noise, but there are weak correlations. Burnable poison does not directly affect the net destruction rate significantly, but Burnable poison also affects the constraints. The neutron multiplication factor and the irradiation time also have tremendous noises. However, there are negative correlations between them.

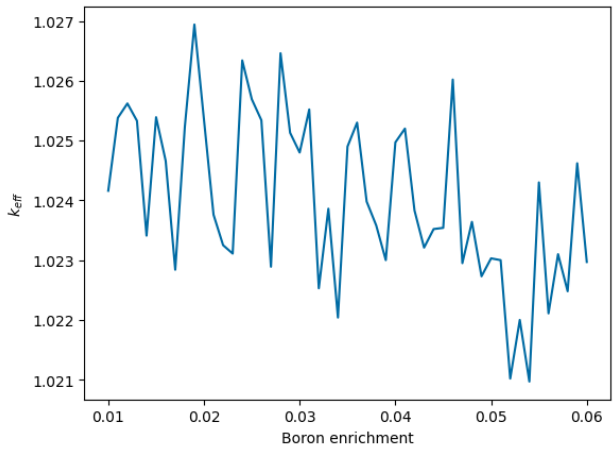
From these results, it can be clarified that finding the optimal design point for the TRU incineration is very difficult even if reactor physics knowledge is applied to the decision-making. Mathematical optimization is essential with domain knowledge to efficiently search the optimal



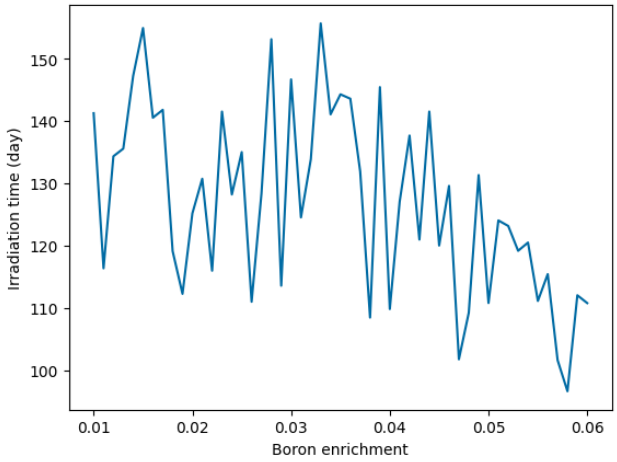
(a) Loaded U-235 at the BOC.



(b) Loaded TRU at the BOC.



(c) Neutron multiplication factor at the BOC.



(d) Irradiation time.

Figure 3.25: The trend of objective values and important values depending on the boron enrichment.

design parameters within the tolerable computation time.

The Pearson correlation coefficients between these design parameters and TRU destruction rate is shown in Table 3.11 As confirmed in each figure, the TRU ratio, fuel kernel radius, and packing fraction have strong positive or negative correlations, and burnable poison enrichment has a moderate correlation.

Table 3.11: The Pearson correlation coefficient of design parameters and TRU destruction rate.

Parameters	Correlations
TRU ratio	0.93
Radius of fuel kernel	-0.91
Packing fraction	-0.87
Uranium enrichment in region 1	-0.12
Uranium enrichment in region 2	0.03
Uranium enrichment in region 3	0.13
Uranium enrichment in region 4	0.03
Boron enrichment	-0.49

3.3.5 Feasible Search and Regression Analysis

In this GT-HTR300 TRU minimization problem, approximately 30% of the combination of design parameters is infeasible if the design parameters are randomly selected. Also, a large portion of the infeasibility is due to the subcriticality of the BOC. Compared with randomly selecting the design parameters and directly solving the optimization problem, feasible search efficiently investigated the feasible region.

Then, surrogate models are built using the gaussian process regression algorithm. The combination of design parameters and objective values is normalized before making a surrogate model using the regression algorithm. So, the normalization of input data and inverse modification of output data is also necessary to use the surrogate model for the prediction. The hyperparameter was tuned for the kernel consisting of constant, RBF, and noise. This hyperparameter is optimized for every nuclides' surrogate model. Twelve surrogate models are built for each nuclide's destruction rate and the amount of TRU and U-235. In the case of infeasibility, the output parameters are assumed to be 0.

The setting for the optimization of the hyperparameters is shown in Table 3.12. The α is the values added to the diagonal of the kernel matrix for preventing numerical problems. The kernel of Gaussian process regression is the combination of RBFs, constant, and the noise for all except for the U-235 amount. For the U-235, Matern kernel with ν = 2.5 (twice differentiable functions)

Table 3.12: The setting for the optimization of the regression algorithm's hyperparameters.

Setting	Values
Length scale bounds for RBF	1.0e-11 - 1.0e+10
Constant value bounds	1.0e-09 - 1.0e+03
Noise level bounds p_c	1.0e-15 - 1.0e+03
α	0.5
Optimizer	Limited memory BFGS-B
Number of restart	80

are selected. Both RBF kernel and Matern kernel uses the anisotropic kernel for length scale.

The optimized hyperparameters for each objective value are shown in Table 3.13.

Table 3.13: The regression algorithm's hyperparameters.

Objective values	Length scale of each parameters for RBF								Constant values	Noise level
	TRU ratio	Kernel radius	Packing fraction	Uranium enrich 1	Uranium enrich 2	Uranium enrich 3	Uranium enrich 4	Boron enrich		
Pu-238	2.29e-01	7.17e-03	1.13e+09	5.73e-03	8.85e-09	2.05e+02	2.52e+06	1.38e+05	1.58e-01	1.60e-15
Pu-239	1.40e-01	7.54e-03	5.38e-02	6.05e-03	1.00e+10	1.00e+10	8.11e-02	5.07e+06	1.78e-01	8.35e-07
Pu-240	1.98e-01	4.87e-03	6.25e-02	9.75e-03	1.17e+05	1.95e+01	1.63e-02	1.27e-02	1.59e-01	9.52e-08
Pu-241	2.28e-01	6.24e-03	4.77e+05	5.85e-03	5.36e-06	8.07e+02	1.39e+02	5.19e+04	8.51e-02	7.23e-08
Pu-242	1.57E-01	3.91E-03	1.21E-02	1.00E+10	9.03E-03	1.47E+03	2.40E+05	4.82E+01	3.16E-05	1.00E-15
Am-241	2.61E-01	6.23E-03	5.73E-02	8.74E-03	1.12E-02	2.64E+04	7.75E-03	2.74E+09	1.10E-01	2.59E-15
Am-243	1.91E-01	6.08E-03	6.12E-02	9.23E-03	1.76E+09	1.00E+10	1.44E-02	1.60E-02	1.78E-01	9.37E-08
Cm-243	1.95E-01	6.12E-03	5.68E-02	9.86E-03	6.16E+03	1.00E+10	1.48E-02	1.61E-02	2.20E-01	8.50E-09
Cm-244	1.40E-01	6.58E-03	2.63E+01	4.82E-03	1.64E+03	5.20E+00	1.55E-02	1.43E-02	1.68E-01	2.93E-07
Cm-245	1.97E-01	1.10E-08	2.68E+04	3.88E+01	1.92E-02	9.25E+01	2.33E+03	3.23E-02	3.92E-05	5.40E-14
TRU	2.06E-01	1.60E-03	3.78E-02	1.29E-02	1.16E+02	1.77E+01	1.04E+01	9.22E-03	3.16E-05	9.16E-15
U-235	1.57E-01	2.04E-02	1.26E-02	4.70E-02	1.49E-02	1.30E-02	3.49E+02	1.03E+02	3.16E-05	1.00E-15
	ν									
U-235	2.5									

3.3.6 Metaheuristic Optimization Search

The metaheuristics algorithms such as genetic algorithms and simulated annealing typically require multiple hyperparameters, depending on the problem characteristics. The hyperparameters of the GA-SA hybrid algorithm are shown in Table 3.14. The first two design parameters are

Table 3.14: The hyperparameter of the GA-SA hybrid algorithm.

Parameter	Values
Number of generation	90
Number of chromosomes per generation	100
Crossover probability p_c	0.8
α	0.1
w_1	0.5
w_2	0.5
g	10000
ζ	2

related to the computation time of the GA-SA hybrid algorithm, and the other parameters affect the balance of diversification and intensification.

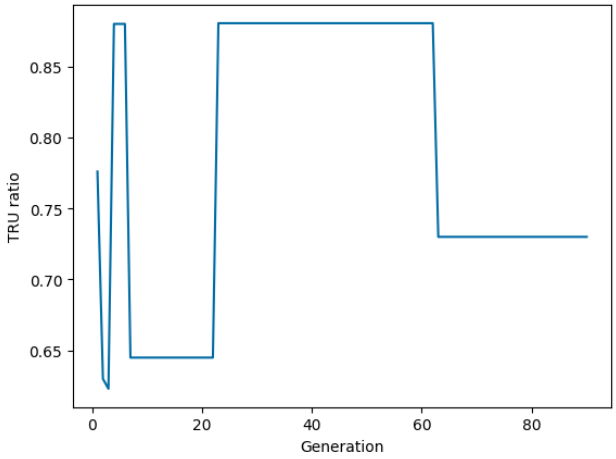
For multi-objective optimization, the weighted sum method is selected. The weight is selected based on the initial amount of TRU in the fuel inventory for each TRU nuclide. In this research, the primary focus of the objective is still TRU destruction rate even though considering other factors. So, the possible differences between the minimum and maximum TRU and U-235 inventory values are evaluated as the same weight for 2% of the destruction rate. The weight of each objective value are shown in Table 3.15

In each iteration of adding data, building a surrogate model, and searching the optimal values in the surrogate model, 50 sets of data are added. In these additional data, 30 % of the data is selected from the $\pm$ 5 % range of optimal points achieved in the previous iteration. Another 30 % of the data is selected from the 10 % range of optimal points achieved in the previous iteration. The other data is randomly selected using Latin hypercube sampling from all design spaces.

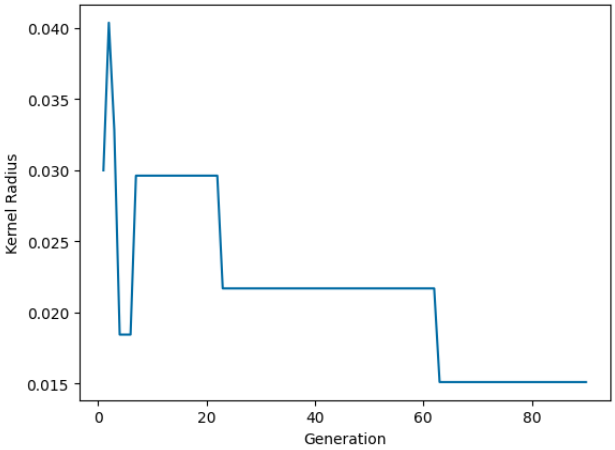
Table 3.15: The weight of the each objective values.

Objective	Weight
Pu-238	7.88e-3
Pu-239	5.28e-1
Pu-240	2.87e-1
Pu-241	6.83e-2
Pu-242	4.66e-2
Am-241	5.22e-2
Am-243	5.64e-3
Cm-243	9.79e-6
Cm-244	8.91e-4
Cm-245	4.90e-5
TRU	2.00e-8
U-235	5.00e-6

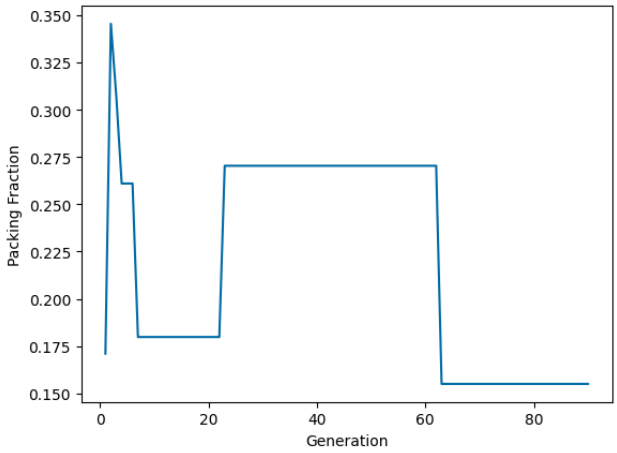
The accuracy threshold of the surrogate model regarding cross-validation is set at 6% of the relative errors. The test thresholds regarding the random point and the optimal point are set 6% and 3% of the relative errors, respectively.



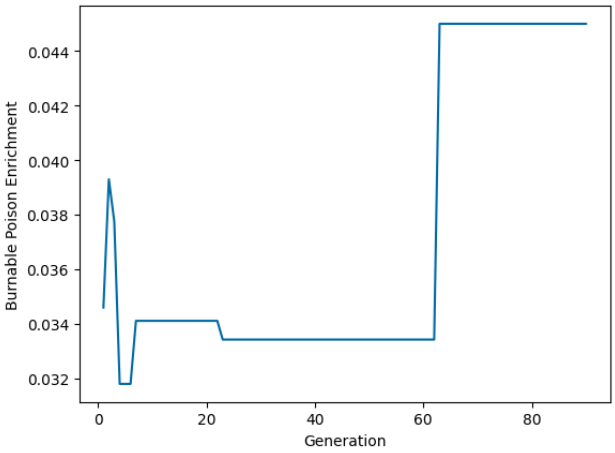
(a) TRU ratio.



(b) Fuel kernel radius.



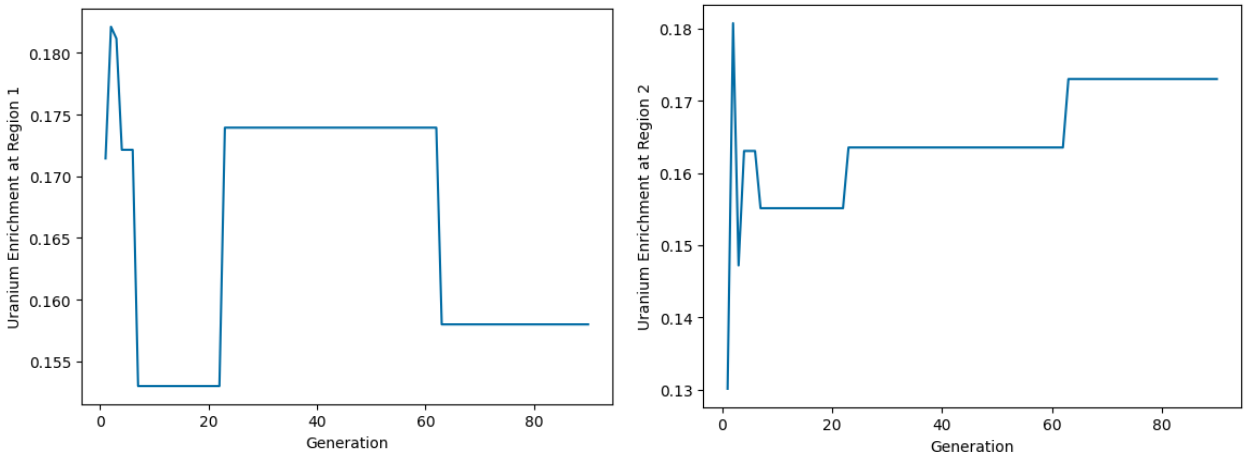
(c) Packing fraction.



(d) Boron enrichment

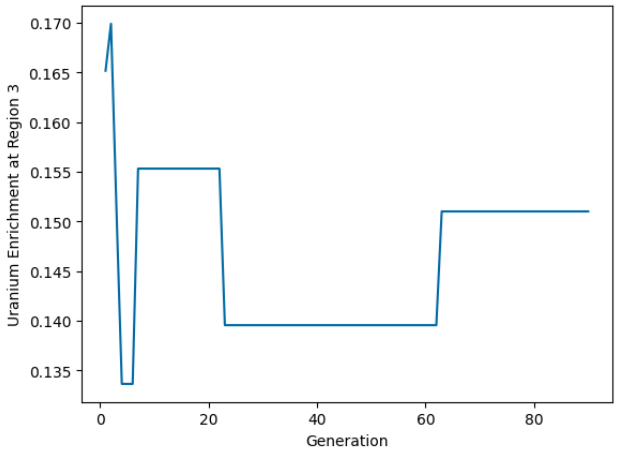
Figure 3.26: The last iteration of design parameters in the GA-SA hybrid algorithm.

Fig. 3.26a - 3.28 show each parameter's design change: TRU ratio, fuel kernel radius, packing fraction, boron enrichment, and Uranium enrichment at region one to four and the best objective values through the last iteration of the GA-SA hybrid algorithm. The last iteration is the seventh iteration. The TRU ratio changes in the higher space of the range between about 60% and 90%. This trend is reasonable because a high TRU ratio is advantageous for all objective values. A high TRU ratio can destroy more fissile nuclides, load more total TRU at the BOC, and contain fewer U-235. Fuel kernel radius and packing fraction also tend to select a lower range of values, and these trends match the result of the correlation check. The change of boron enrichment does

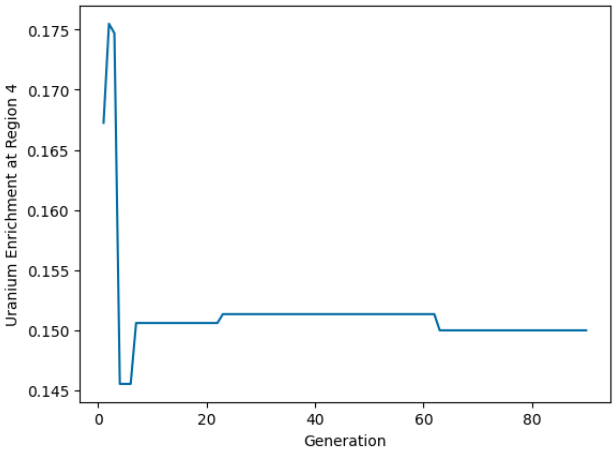


(a) Uranium enrichment at region one.

(b) Uranium enrichment at region two.



(c) Uranium enrichment at region three.



(d) Uranium enrichment at region four.

Figure 3.27: The last iteration of Uranium enrichment parameters in the GA-SA hybrid algorithm.

not show a meaningful trend. Regarding the Uranium enrichment of the different regions, regions two and four are outside the reactor core, and regions one and three are inside the core. From these changes, there are no noteworthy trends. The initial part of the generation changes more parameters because of the room to improve the weighted objective values and the chance of the crossover and the mutation. In the later generation, the updates of the best performance are few, and theoretically, relatively few changes should happen more frequently than significant changes. This last iteration case did not behave like that: the last change is the significant shift around 60 generations. There are no parameter changes after that change around 60 generations due to the

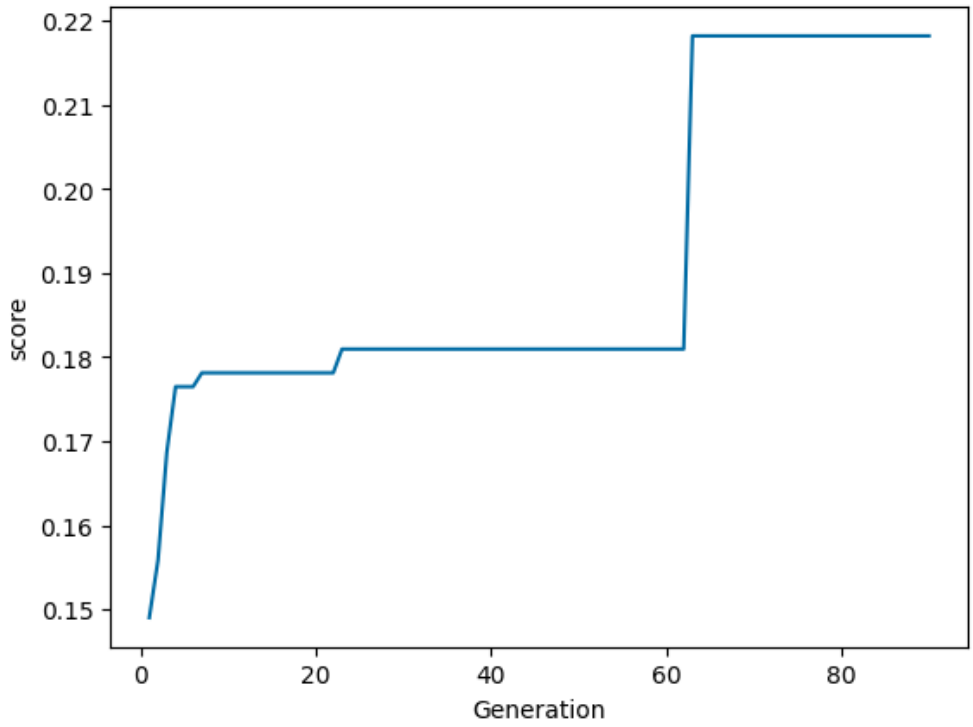


Figure 3.28: The last iteration of weighted objective values in the GA-SA hybrid algorithm.

stochastic process. The optimal design parameters are shown in Table 3.16. Each parameter is

Table 3.16: The optimal design parameters.

Parameters	Values
TRU ratio	73 %
Kernel radius	151 μ m
Packing fraction	15.5 %
Uranium enrichment at region 1	15.8 %
Uranium enrichment at region 2	17.3 %
Uranium enrichment at region 3	15.1 %
Uranium enrichment at region 4	15.0 %
Boron enrichment	4.5 %

defined at 3.3.2 and 3.3.3. The raw values have a long fractional part, and they are rounded.

The TRU ratio is high, and the kernel radius and packing fraction are close to the lower boundaries. This result is similar to the analysis as shown in the correlations between the two values. In the middle of the generation, the TRU ratio is close to the high boundary, and the kernel radius and packing fraction were higher values than the later generation. It infers that the minimum kernel radius and packing fraction are very important for the high objective values, but the TRU ratio does not affect compared with them. So, the chromosome with a shallow kernel radius and packing fraction is selected even though the TRU ratio decreases. For the actual optimal point, the TRU ratio should be much higher. So, more generation is necessary to obtain that value. Even though the combination of low packing fraction and small fuel kernel decreases the total TRU amount significantly, the high TRU ratio decreases the amount of U-235 in addition to the amount of TRU. Hence, the sanction posed by these two objective values is not severe. The uranium enrichment reached the middle range, not depending on the region. The four regions of fuel area are not enough to see the trend of the regional effect of fuel enrichment, and it is necessary to subdivide the region more. The burnable poison enrichment also does not significantly affect the results.

The last iteration of each objective value in the surrogate model with their errors and the simulated optimal design performance using the optimal design parameters are shown in Table 3.17. The weighted objective values are evaluated summation of all objective values using the weighted sum method. Note that the TRU destruction rate is not used to calculate weighted objective values. The fissile nuclides decrease the amount(positive destruction rate), and the non-fissile nuclides increase the amount(negative destruction rate) through the depletion. The error of weighted objective values is below 3.00%, and most of the nuclides are in the same order of errors or less. Pu-241 and Cm-243 destruction rates have a higher order of errors. Significantly, the effect of Pu-241 is relatively crucial due to the weight of this nuclide. The average relative error of the surrogate model is 5.83%. In the design parameters for the best weighted objective values, the TRU destruction rate recorded 25.9%, and the Pu-239 destruction rate recorded 58.7%.

Even if U-235 and TRU amount is not considered, the accuracy of TRU destruction rate, which is directly estimated, is slightly better than estimating each nuclide individually (2.32e-02 vs.

Table 3.17: The optimal design performance.

Objectives	Surrogate model values	Simulation values	Errors
Pu-238 destruction rate	-1.18e+00	-1.23e+00	-4.39e-02
Pu-239 destruction rate	5.90e-01	5.87e-01	4.53e-03
Pu-240 destruction rate	2.52e-02	2.53e-02	-8.49e-04
Pu-241 destruction rate	-6.47e-01	-7.19e-01	-1.11e-01
Pu-242 destruction rate	-3.43e-01	-3.35e-01	2.21e-02
Am-241 destruction rate	4.74e-01	4.59e-01	3.25e-02
Am-243 destruction rate	-1.02e+00	-1.11e+00	-8.92e-02
Cm-243 destruction rate	-1.67e+01	-1.86e+01	-1.17e-01
Cm-244 destruction rate	-2.92e+00	-3.14e+00	-7.82e-02
Cm-245 destruction rate	-3.20e+00	-3.39e+00	-6.02e-02
TRU destruction rate	2.65e-01	2.59e-01	2.32e-02
TRU amount at the BOC[g]	8.96e+05	8.65e+05	3.47e-02
TRU amount at the EOC[g]		6.41e+05	
U-235 amount at the BOC[g]	3.75e+02	3.88e+02	-3.60e-02
U-235 amount at the EOC[g]		3.17e+02	
Weighted objective values	2.85e-01	2.77e-01	2.97e-02
k_{eff} at the BOC		1.04e+00	
Time at the EOC[day]		4.24e+02	

2.37e-02). Still, the individual calculation has some advantage in seeing each nuclides' characteristics change and weighing the importance for specific aspects such as toxicity, but the direct estimation of the TRU destruction may be efficient in a simple case.

3.3.7 Discussion

3.3.7.1 Efficiency

In the case of the proposed framework, 600 times simulation is executed to achieve the optimal values. In the case of brute force search for all design space, even 2500 times of simulation did not achieve the same performance of the optimal values. So, the proposed framework can be efficient. However, brute force search is not applied in any context, Practically. The design search space is tried to be minimized by analysis and engineer's judgment. Hence the improved efficiency of the proposed framework is essential. The efficiency of the algorithms depends on the hyperparameters of hybrid GA-SA algorithms and regression algorithms, the threshold of the accuracy, the number

of additional simulation data for building the surrogate model in each outer iteration. For each iteration of regression algorithms and hybrid GA-SA algorithms, the optimization of hyperparameters is not so tricky. It can be done based on cross-validation or grid search.

The limitation of this research scope is that some hyperparameters are not optimized in a strong logic. For example, the number of additional simulation data will highly affect the total iteration of the loop. The higher number of data addition in each iteration may achieve a low iteration number, but the possibility of local optima increases. Also, it may repeat the inefficient iteration. On the other hand, the small number of additions in each iteration may take much more time to execute the hybrid GA-SA algorithm. It is difficult to decide these parameters by the grid search or cross-validation because of the longer computation time. Hence, some efficient sets of optimized hyperparameters can be achieved, but there remains some room for improvement.

3.3.7.2 Accuracy

The algorithm's accuracy was confirmed by the optimal point and averaged multiple random points as relative errors. There are two criteria because the whole surrogate model does not require very high accuracy compared with the area close to the optimal point because the trend of objective values is enough to see the design space trend. The threshold of optimal point accuracy is set at 3.00% because there is no significant improvement in the accuracy in terms of iteration and the original simulation precision. In the vast design space, decreasing the errors of the surrogate model is costly, but it can be a reasonable data size if it is focused more on the optimal point.

4. CONCLUSIONS AND FUTURE STUDY

This chapter shows the conclusions of the dissertation and the future study to expand from this dissertation.

4.1 Summary and Conclusions

In this dissertation, the optimization framework for the nuclear reactor design problem was developed and tested for the nuclear waste incineration problem utilizing a high fidelity HTTR and GT-HTR300 reactor model. The framework consists of multiple modules to realize the optimization search efficiently. The framework simulates the necessary input and output data pairs for building a surrogate model, searches a feasible region using a trust region-based method, builds a surrogate model using a Gaussian process regression method, and searches the surrogate model using a hybrid of the genetic algorithms and simulated annealing. For the nuclear waste incineration problems, two serpent models: HTTR and GT-HTR300, were modeled from scratch. Sub-modules for the plotting and evaluating the output results were also developed for the convenient simulation. Surrogate models were built using regression algorithms with optimized hyperparameters. The nested iterations of the hybrid metaheuristics and surrogate model improvement worked in terms of accuracy and efficiency.

The two cases of the reactor design optimization loading TRU were conducted for TRU incineration. In the case of HTTR, the optimal design to maximize the TRU destruction rate was investigated. TRU was from the spent fuel of AP1000. As a result of the optimization framework, the 5.1 % of TRU destruction rate is obtained with the design parameters of 13% of TRU ratio, 410 μm of fuel kernel radius, 15% of packing fraction, 15% of uranium enrichment, and 4.9% of boron enrichment as shown in Table 3.5. Even though the optimal case, the TRU destruction rate is not high because of the unfavorable neutron flux for the TRU incineration purpose. However, the framework selected the possible optimal TRU incineration design. In the case of GTHTR-300, the optimal design to maximize the TRU destruction rate considering initial TRU

loading and U-235 loading was investigated. TRU was also from the spent fuel of AP1000, but Neptunium was extracted before fabrication of the TRU fuel kernel. As a result of the optimization process of the framework, the 25.9% of TRU destruction rate with 865kg of TRU loading and 388g of U-235 loading was achieved, as shown in Table 3.17. The selected optimal design parameters for that destruction rate are 73% of TRU ratio, 151μm of fuel kernel radius, 15.5% of packing fraction, 15.8%, 17.3%, 15.1%, and 15.0% of Uranium enrichment at regions 1, 2, 3, and 4, respectively, and the 4.5% of boron enrichment as shown in Table 3.16. The error of the TRU destruction rate was 2.32%. The optimal value of the fuel kernel radius and packing fraction were close to the lower boundary, and the TRU ratio is in the higher range of the possible value. This infers that the achieved optimal point is close to the actual optimal point considering the parameter sensitivity analysis, but more generations for the search can find the better optimal point. For efficiency, the framework executed 600 times of simulation to achieve the optimal values. On the other hand, more than 2500 times of simulations using the brute force search did not achieve the same performance. The proposed framework showed the applicability and performance to the waste minimization problem.

4.2 Further Study

The current research showed more systematic ways for the nuclear waste minimization problem using mathematical optimization. The applicability of the optimization framework to the nuclear waste incineration problem was clarified. However, there are potentially more ways to improve the framework and offer more efficient and accurate results to the optimization problem. This section briefly introduces a set of suggestions.

4.2.1 Non-Stochastic Search Algorithm

To search for a surrogate model, the metaheuristic hybrid GA-SA algorithm was selected in this research. A metaheuristic is a robust approach, but the computation time is costly. The recent summary of the surrogate modeling for the black box and gray box optimization[69] summarizes some latest research not using genetic algorithms to search the surrogate model. These approaches

may improve the efficiency of the framework.

4.2.2 Fuel Shuffling, Design Parameters, and Constraints

For a higher TRU destruction rate, several batches of depletion are necessary[26]. For the simple estimation, there is an equation to predict the burnup[70].

$$B_{n-batch} = \frac{2n}{n+1} B_{one-batch}, \tag{4.1}$$

where $B_{n-batch}$ is the discharged burn-up of n batch, $B_{one-batch}$ is the discharged burn-up of one batch, and n is the number of batches. The potential TRU destruction rate can be roughly estimated using it. However, it is necessary to simulate each batch for accurate calculation, and it takes a lot of computation cost. If the surrogate modeling supports it, the computation time can significantly decrease.

Additional design parameters such as more detailed uranium enrichment distribution and the core size are helpful for design optimization. Also, the weight of each objective value can be adjusted more suitable for specific reasons, such as the toxicity of TRU nuclides.

In this research scope, only the fundamental reactor physics values were evaluated as the constraints of design values; however, there are more constraints, such as the thermal-hydraulics aspects. To consider these aspects will make the framework more practical and beneficial for the superiority of the knowledge-based optimization because of more complexity of the problem.

4.2.3 Regression Algorithm

The Gaussian process regression was selected for building a surrogate model in this research for the balance of accuracy and computational cost. There are several different models for the choice of the surrogate model. For example, there are some basic functions for simpler modeling, while there are gradient-boosted tree algorithms and artificial neural networks for more complex models. Each iteration of building the surrogate model and searching objective values, required accuracy may vary. So, using another regression algorithm can be a better candidate instead of the Gaussian process regression.

4.2.4 Hyperparameters Optimization

Since the hybrid GA-SA algorithm considers the balance of diversification and intensification, each iteration performs efficiently to some extent. However, some hyperparameters are not entirely optimized as described at 3.3.7.1. The number of additional simulation data and how to select the additional data in each interation will affect the framework performance significantly, but the optimization of those parameters were not performed due to the long computation time for the sensitivity analysis. So, these parameter improvements are going to achieve better performance in terms of efficiency and accuracy.

Regarding the hyperparameters of the hybrid GA-SA algorithm, there are two kinds of parameters: one is for adjusting the diversification and intensification, the other is for adjusting the computation time and accuracy in addition to the diversification and intensification. The latter is the number of generations and number of chromosomes in this research. It is imperative to take enough generations to search for the surrogate model with diversification and intensification because their focus is shifted from diversification to intensification through the generations. If the hyperparameter tuning is leaned to one of them, fewer generations are potentially risky. Also, the hybrid algorithm expanding the genetic algorithm tends to weigh the diversification rather than the intensification. Furthermore, hyperparameter tuning is performed at the first iteration so that the model accuracy is not so well. Considering these factors, it can be helpful to weigh more intensification by adjusting the parameters.

REFERENCES

[1] "Status and trends in spent fuel and radioactive waste management," Tech. Rep. No. NF-T-4.6, International Atomic Energy Agency, VIENNA, January 2009.

[2] "IAEA safety standards for protecting people and the environment classification of radioactive waste," Tech. Rep. No. GSG-1, International Atomic Energy Agency, VIENNA, November 2009.

[3] U.S. NRC, "High-Level Waste," March 2020. https://www.nrc.gov/waste/high-level-waste.html.

[4] World Nuclear News, "Biden nominee confirms opposition to Yucca Mountain," January 2021. https://world-nuclear-news.org/Articles/Biden-nominee-confirms-opposition-to-Yucca-Mountai.

[5] Congress.gov, "H.R.4091 - 116th congress (2019-2020): ARPA–E reauthorization act of 2019," June 2020. https://www.congress.gov/bill/116th-congress/house-bill/4091.

[6] "Status of minor actinide fuel development," Tech. Rep. No. NW-T-1.14, International Atomic Energy Agency, VIENNA, December 2018.

[7] C. Schneidmiller, "Energy dept. starts 'small' used-fuel reprocessing effort," January 2020. https://www.exchangemonitor.com/doe-starts-small-used-fuel-reprocessing-effort/.

[8] C. Schneidmiller, "DOE nuclear energy chief reaffirms interest in foreign reprocessing for U.S. spent fuel," May 2020. https://www.exchangemonitor.com/doe-nuclear-energy-chief-reaffirms-interest-foreign-reprocessing-u-s-spent-fuel/.

[9] C. H. Piercy, "RE: public meeting on "status of spent fuel reprocessing rulemaking" (held march 4, 2020, ML20063L785)," May 2020. https://www.ans.org/file/1571/1/05.28.20%20-%20ANS%20Response%20to%20Reprocessing%20Rulemaking.pdf.

[10] J. Ahlswede and M. B. Kalinowski, "Germany's current and future plutonium inventory," *The Nonproliferation Review*, vol. 19, no. 2, pp. 293–312, 2012.

[11] S. Krikorian, "France's efficiency in the nuclear fuel cycle: What can 'oui' learn?," September 2019. https://www.iaea.org/newscenter/news/frances-efficiency-in-the-nuclear-fuel-cycle-what-can-oui-learn.

[12] European Nuclear Society, "Reprocessing plants, world-wide." https://web.archive.org/web/20150622040852/http://www.euronuclear.org/info/encyclopedia/r/reprocessing-plants-ww.htm.

[13] World Nuclear News, "Regulator confirms safety of Japanese reprocessing plant," May 2020. https://www.world-nuclear-news.org/Articles/Regulator-approves-safety-of-Japanese-reprocessing.

[14] World Nuclear Association, "China's nuclear fuel cycle," October 2019. https://www.world-nuclear.org/information-library/country-profiles/countries-a-f/china-nuclear-fuel-cycle.aspx.

[15] World Nuclear Association, "Russia's nuclear fuel cycle," June 2020. https://www.world-nuclear.org/information-library/country-profiles/countries-o-s/russia-nuclear-fuel-cycle.aspx.

[16] E. Law, "Cleaning up our nuclear past: faster, safer and sooner," October 2018. https://nda.blog.gov.uk/2018/10/23/what-is-reprocessing/.

[17] F. Heidet, T. Kim, and T. Taiwo, "Impact of minor actinide recycling on sustainable fuel cycle options," *Nuclear Engineering and Design*, vol. 323, pp. 434 – 462, 2017.

[18] K. Ishii, M. Yamaoka, Y. Moriki, T. Oomori, Y. Tsuboi, K. Arie, and M. Kawashima, *Development of Uranium-Free TRU Metallic Fuel Fast Reactor Core*, pp. 155–167. Tokyo: Springer Japan, 2015.

[19] R. Takeda, J. Miwa, and K. Moriya, "BWRS for long-term energy supply and for fissioning almost all transuranium," in *Proceedings of GLOBAL 2007 conference on advanced nuclear fuel cycles and systems*, 2007.

[20] R. Versluis, F. Venneri, D. Petti, L. Snead, and D. McEachern, "Project Deep-Burn: development of transuranic fuel for high-temperature helium-cooled reactors," in *Proceedings of the 4th International Topical Meeting on High Temperature Reactor Technology (HTR2008)*, (Washington, DC, USA), September 2008.

[21] R. Wigeland, T. Taiwo, H. Ludewig, M. Todosow, W. Halsey, J. Gehin, R. Jubin, J. Buelt, S. Stockinger, K. Jenni, and B. Oakley, "Nuclear fuel cycle evaluation and screening - fianl report," Tech. Rep. INL/EXT-14-31465, Idaho National Laboratories, October 2014.

[22] S. Bays, H. Zhang, and M. Pope, "Deep burn fuel cycle integration: Evaluation of twotier scenarios," Tech. Rep. INL/EXT-09-15915, Idaho National Laboratories, May 2009.

[23] C. Rodriguez, A. Baxter, D. McEachern, M. Fikani, and F. Venneri, "Deep-burn: making nuclear waste transmutation practical," *Nuclear Engineering and Design*, vol. 222, no. 2, pp. 299 – 317, 2003. HTR-2002 1st international topical meeting on High Temperature reactor technology.

[24] D. E. Ames II, "High fidelity nuclear energy system optimization towards an environmentally benign, sustainable, and secure energy source," Tech. Rep. SAND2010-6684, Sandia National Laboratory, Albuquerque, NM, USA, October 2010.

[25] D. E. Ames II, *High-fidelity nuclear energy system optimization towards an environmentally benign, sustainable, and secure energy source*. PhD thesis, College Station TX, 2010.

[26] Y. Kim and F. Venneri, "Optimization of one-pass transuranic deep burn in a modular helium reactor," *Nuclear science and engineering*, vol. 160, no. 1, pp. 59–74, 2008.

[27] T. G. Lewis, *Physics-Based 3D Multi-Directional Reloading Algorithm for Deep Burn HTR Prismatic Block Systems*. PhD thesis, College Station TX, 2010.

[28] P. V. Tsvetkov, T. G. Lewis, A. B. Alajo, and D. E. Ames, ""used fuel" vectors and waste minimization strategies for vhtrs operating without refueling," *Nuclear Engineering and Design*, vol. 240, no. 10, pp. 2458–2465, 2010. 4th International Topical Meeting on High Temperature Reactor Technology (HTR 2008), with Regular Papers.

[29] Y. Fukaya, M. Goto, H. Ohashi, Y. Tachibana, K. Kunitomi, and S. Chiba, "Proposal of a plutonium burner system based on htgr with high proliferation resistance," *Journal of Nuclear Science and Technology*, vol. 51, no. 6, pp. 818–831, 2014.

[30] Y. Fukaya, M. Goto, H. Ohashi, X. Yan, T. Nishihara, Y. Tsubata, and T. Matsumura, "Uranium-based TRU multi-recycling with thermal neutron HTGR to reduce environmental burden and threat of nuclear proliferation," *Journal of Nuclear Science and Technology*, vol. 55, no. 11, pp. 1275–1290, 2018.

[31] C. Darwin, *On the Origin of Species by Means of Natural Selection*. London: Murray, 1859. or the Preservation of Favored Races in the Struggle for Life.

[32] J. H. Holland, *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. University of Michigan Press, 1975.

[33] D. Goldberg, G. David Edward, D. Goldberg, and V. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Artificial Intelligence, Addison-Wesley Publishing Company, 1989.

[34] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.

[35] A. Conn, N. Gould, and P. Toint, *Trust Region Methods*. MPS-SIAM Series on Optimization, Society for Industrial and Applied Mathematics (SIAM, 3600 Market Street, Floor 6, Philadelphia, PA 19104), 2000.

[36] J. H. Friedman, "Greedy function approximation: A gradient boosting machine.," *Ann. Statist.*, vol. 29, pp. 1189–1232, 10 2001.

[37] T. Chen and C. Guestrin, "XGBoost," *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug 2016.

[38] C. M. Pereira and C. M. Lapa, "Coarse-grained parallel genetic algorithm applied to a nuclear reactor core design optimization problem," *Annals of Nuclear Energy*, vol. 30, no. 5, pp. 555–565, 2003.

[39] C. M. do Nascimento Abreu Pereira, R. Schirru, and A. S. Martinez, "Basic investigations related to genetic algorithms in core designs," *Annals of Nuclear Energy*, vol. 26, no. 3, pp. 173–193, 1999.

[40] B. V. Haibach and M. A. Feltus, "A study on the optimization of integral fuel burnable absorbers using the genetic algorithm based cigaro fuel management system," *Annals of Nuclear Energy*, vol. 24, no. 6, pp. 439–448, 1997.

[41] H. Sayyaadi and T. Sabzaligol, "Various approaches in optimization of a typical pressurized water reactor power plant," *Applied energy*, vol. 86, no. 7-8, pp. 1301–1310, 2009.

[42] J. L. C. Chapot, F. Carvalho Da Silva, and R. Schirru, "A new approach to the use of genetic algorithms to solve the pressurized water reactor's fuel management optimization problem," *Annals of Nuclear Energy*, vol. 26, no. 7, pp. 641–655, 1999.

[43] M. D. DeChaine and M. A. Feltus, "Nuclear fuel management optimization using genetic algorithms," *Nuclear Technology*, vol. 111, no. 1, pp. 109–114, 1995.

[44] R. Omori, Y. Sakakibara, and A. Suzuki, "Applications of genetic algorithms to optimization problems in the solvent extraction process for spent nuclear fuel," *Nuclear technology*, vol. 118, no. 1, pp. 26–31, 1997.

[45] F. Alim, K. Ivanov, and S. H. Levine, "New genetic algorithms (GA) to optimize PWR reactors: Part I: Loading pattern and burnable poison placement optimization techniques for PWRs," *Annals of Nuclear Energy*, vol. 35, no. 1, pp. 93–112, 2008.

[46] X. Ingremeau, P. Dumaz, D. Plancq, P. Richard, D. Haubensack, B. Iooss, G. Rimpault, M. Zabiego, and S. David, "FARM: A new tool for optimizing the core performance and safety characteristics of gas cooled fast reactor cores," 2010.

[47] E. Hourcade, X. Ingremeau, P. Dumaz, S. Dardour, D. Schmitt, S. Massara, and G. Darmet, "Innovative methodologies for fast reactor core design and optimization," 2011.

[48] J. E. Bevins and R. N. Slaybaugh, "Gnowee: A hybrid metaheuristic optimization algorithm for constrained, black box, combinatorial mixed-integer design," *Nuclear Technology*, vol. 205, no. 4, pp. 542–562, 2019.

[49] A. Kumar and P. V. Tsvetkov, "A new approach to nuclear reactor design optimization using genetic algorithms and regression analysis," *Annals of Nuclear Energy*, vol. 85, pp. 27 – 35, 2015.

[50] K. Zeng, N. E. Stauff, J. Hou, and T. K. Kim, "Development of multi-objective core optimization framework and application to sodium-cooled fast test reactors," *Progress in Nuclear Energy*, vol. 120, p. 103184, 2020.

[51] O. Fabbris, S. Dardour, P. Blaise, J.-H. Ferrasse, and M. Saez, "Surrogates based multicriteria predesign methodology of sodium-cooled fast reactor cores–application to cfv-like cores," *Nuclear Engineering and Design*, vol. 305, pp. 314–333, 2016.

[52] I. Bajaj, S. S. Iyer, and M. Faruque Hasan, "A trust region-based two phase algorithm for constrained black-box and grey-box optimization with infeasible initial point," *Computers Chemical Engineering*, vol. 116, pp. 306 – 321, 2018. Multi-scale Systems Engineering – in memory honor of Professor C.A. Floudas.

[53] P.-H. Chen and S. M. Shahandashti, "Hybrid of genetic algorithm and simulated annealing for multiple project scheduling with multiple resource constraints," *Automation in Construction*, vol. 18, no. 4, pp. 434–443, 2009.

[54] S.-H. Ling and F. H. Leung, "An improved genetic algorithm with average-bound crossover and wavelet mutation operations," *Soft computing*, vol. 11, no. 1, pp. 7–31, 2007.

[55] A. E. Johnson, D. Kotlyar, S. Terlizzi, and G. Ridley, "serpenttools: A python package for expediting analysis with serpent," *Nuclear Science and Engineering*, vol. 194, no. 11, pp. 1016–1024, 2020.

[56] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.

[57] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, "Array programming with NumPy," *Nature*, vol. 585, pp. 357–362, Sept. 2020.

[58] Wes McKinney, "Data Structures for Statistical Computing in Python," in *Proceedings of the 9th Python in Science Conference* (Stéfan van der Walt and Jarrod Millman, eds.), pp. 56 – 61, 2010.

[59] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.

[60] S. Saito, T. Tanaka, and Y. Sudo, "Design of high temperature engineering test reactor (HTTR)," tech. rep., Japan Atomic Energy Research Inst., 1994.

[61] X. Yan, K. Kunitomi, T. Nakata, and S. Shiozawa, "GTHTR300 design and development," *Nuclear Engineering and Design*, vol. 222, no. 2-3, pp. 247–262, 2003.

[62] J. Leppänen, M. Pusa, T. Viitanen, V. Valtavirta, and T. Kaltiaisenaho, "The serpent monte carlo code: Status, development and applications in 2013," *Annals of Nuclear Energy*, vol. 82, pp. 142–150, 2015. Joint International Conference on Supercomputing in Nuclear Applica-

tions and Monte Carlo 2013, SNA + MC 2013. Pluri- and Trans-disciplinarity, Towards New Modeling and Numerical Simulation Paradigms.

[63] "Evaluation of High Temperature Gas Cooled Reactor Performance: Benchmark Analysis Related to Initial Testing of the HTTR and HTR-10," Tech. Rep. IAEA-TECDOC-1382, International Atomic Energy Agency, VIENNA, January 2003.

[64] J. D. Bess, N. Fujimoto, B. H. Dolphin, L. Snoj, and Z. Stsushi, "Evaluation of the start-up core physics tests at Japan's high temperature engineering test reactor (fully loaded core)," Tech. Rep. INL/EXT-08-14767, Idaho National Laboratories, March 2009.

[65] M.-H. M. Cuvelier and P. V. Tsvetkov, "TRU management and ^{235}U consumption minimization in fuel cycle scenarios with AP1000 and VHTRs," *Annals of Nuclear Energy*, vol. 55, pp. 137 – 150, 2013.

[66] M.-H. M. Cuvelier, "Advanced fuel cycle scenarios with AP1000 pwrs and VHTRs and fission spectrum uncertainties," Master's thesis, College Station TX, 2012.

[67] Y. Fukaya, K. Kunitomi, and M. Ogawa, "Study on reduction of potential radiotoxicity for spent fuel by using HTGR," *Nippon Genshiryoku Gakkai Wabun Ronbunshi (Online)*, vol. 14, no. 3, pp. 189–201, 2015.

[68] Y. Fukaya, M. Goto, and T. Nishihara, "Development on nuclear design modelfor detailed design of clean burn HTGR," Tech. Rep. JAEA-Technology 2015-017, Japan Atomic Energy Agency., 2015.

[69] I. Bajaj, A. Arora, and M. M. F. Hasan, *Black-Box Optimization: Methods and Applications*, pp. 35–65. Cham: Springer International Publishing, 2021.

[70] H. W. Graves Jr, *Nuclear fuel management*. John Wiley Sons, 1979.