INL/RPT-22-70382

Light Water Reactor Sustainability Program

Development of Plant Reload Optimization Platform Capabilities for Core Design and Fuel Performance Analysis

December 2022

U.S. Department of Energy

Office of Nuclear Energy

INL/RPT-22-70382

Development of Plant Reload Optimization Platform Capabilities for Core Design and Fuel Performance Analysis

Yong-Joon Choi[1]
Mohammad Abdo[1]
Congjian Wang[1]
Jarrett Valeri[2]
Cesare Frepoli[2]
Khang Nguyen[3]
Jason Hou[3]

[1]Idaho National Laboratory,
[2]FPoliSolutions LLC,
[3]North Carolina State University

December 2022

Prepared for the
U.S. Department of Energy
Office of Nuclear Energy
Under DOE Idaho Operations Office
Contract DE-AC07-05ID14517

EXECUTIVE SUMMARY

The United States (U.S.) nuclear industry faces a challenge in maintaining required levels of safety while ensuring economic competitiveness to stay in business. Safety remains a key parameter for all aspects of light water reactor (LWR) nuclear power plant (NPP) operations. Safety can become more economical by using a risk-informed ecosystem, such as the one being developed by the Risk-Informed Systems Analysis (RISA) Pathway under the U.S. Department of Energy (DOE) Light Water Reactor Sustainability (LWRS) Program. The LWRS Program promotes a wide range of research and development activities with the goal of maximizing both the safety and economic efficiency of NPPs through improved scientific understanding, especially given many plants are now considering second license renewals.

The RISA Pathway has two main goals: (1) deploy methodologies and technologies that better represent safety margins and cost and safety factors and (2) develop advanced applications that enable cost-effective plant operation.

The Plant Reload Optimization Platform development project aims to build a reactor core design tool that includes reactor safety and fuel performance analyses, and also uses artificial intelligence to support optimization of core design solutions.

This report summarizes Fiscal Year 2022 (FY-22) activity in platform capability developments in RAVEN. This platform performs simulations using industry codes for core design (i.e., PARCS) and fuel performance (i.e., TRANSURANUS) which will allow expansion of the capabilities to include advanced fuel designs such as accident-tolerant fuel (ATF)s with high burnup.

Page intentionally left blank

CONTENTS

FIGURES

TABLES

ACRONYMS

ATF	accident tolerance fuel
DBA	design basis accident
DNBR	departure nucleate boiling rate
DOE	U.S. Department of Energy
FY	fiscal year
GA	genetic algorithm
HFP	hot full power
INL	Idaho National Laboratory
ITU	Institute for Transuranium Elements
LOCA	loss-of-coolant accident
LWR	light water reactor
LWRS	Light Water Reactor Sustainability
PARCS	Purdue Advanced Reactor Core Simulator
PWR	pressurized water reactor
RAVEN	Risk Analysis and Virtual Environment
RELAP	Reactor Excursion and Leak Analysis Program
RISA	Risk-Informed Systems Analysis
U.S.	United States

1. INTRODUCTION

The United States (U.S.) Department of Energy (DOE) Light Water Reactor Sustainability (LWRS) Program Risk-Informed Systems Analysis (RISA) Pathway Plant Reload Optimization project aims to develop an integrated, comprehensive platform offering an all-in-one solution for reactor core reload evaluations with a special focus on optimization of core design considering feedback from system safety analysis (i.e., thermal-hydraulics) and fuel performance [1].

The RISA Pathway optimization platform is mainly driven by the Idaho National Laboratory (INL)-developed Risk Analysis and Virtual Environment (RAVEN) [2] computer software which gives unlimited flexibility in using modern artificial intelligence techniques such as Genetic Algorithm (GA). This GA method is a proven technology for fuel reload optimization purpose [3].

RAVEN's capability is not just limited to optimization. It can also provide input decks to other physical codes and perform post-processing of simulation results. This extensibility of RAVEN facilitates coupling with other physical codes for core design, fuel performance, and systems analysis, which can lead to a unified framework that considers physical phenomena. Hence, using RAVEN as a controller of the GA method allows a "tool-independent" one-stop plant reload optimization platform with easy access for users.

The optimization platform can set multiple objectives and constraints such as fuel cycle length (e.g., an extension from 18 to 24 months), fuel enrichment, burnable poisons, core design limits (e.g., peaking factors and boron concentration), safety parameters (e.g., peak cladding temperature and departure of nucleate boiling rate [DNBR]). To do this, the RISA Pathway GA-based optimization platform uses the following individual computational tools coupled with RAVEN to provide safety feedback during core designing:

- PARCS for core design
- RELAP5-3D for system response analysis
- TRANSURANUS for fuel performance analysis.

During the core design code benchmark study by the RISA pathway, the PARCS and TRANSURANUS codes showed high degrees of flexibility and efficiency in the RISA pathway GA-based optimization platform [1].

Figure 1-1 gives a snapshot of the optimization platform. Initial core design is given by RAVEN, and PARCS generates the equilibrium core which is the required input for RELAP5-3D limiting design basis accidents (DBA) analyses. Once the core design is found acceptable by RELAP5-3D analyses, fuel performance is assessed by TRANSURANUS for final confirmation of an acceptable core design. This process is controlled by RAVEN along with an uncertainty analysis performed by RELAP5-3D. The choice of the analytical tools herein is for demonstrative purposes. The platform is designed as "plug-and-play" where individual tools can be replaced, provided the proper interfaces with RAVEN are developed.

The uncertainties can be quantified by RAVEN during the multi-physics simulation. However, the propagation of uncertainties across the different physics calculations may increase complexity of the algorithm, computational burden, and applicability in practical use. In some circumstances it could be more convenient and efficient to bound values from one discipline before proceeding to the next step in the simulation stream, especially when the potential loss in analytical margin is small compared to the added complexity.

Note that for analyses directly supporting plant licensing basis, additional or potentially different tools may be needed. For example, the reactor subchannel analysis is typically modeled by another thermal-hydraulics code which solves the details of the heat transfer within the fuel assembly. This is necessary for the evaluation of critical heat flux or DNBR which has associated limits tracked in the safety analyses. The subchannel code is typically validated with fuel-product-specific data, often from the fuel vendors' proprietary data.

This report describes coupling between RAVEN, PARCS, and TRANSURANUS as well as presented demonstrations for verification of the developed GA optimization platform. As shown in Figure 1-1, TRANSURANUS needs data from both PARCS and RELAP5-3D through RAVEN. Coupling between RAVEN and RELAP5-3D is already completed [2].

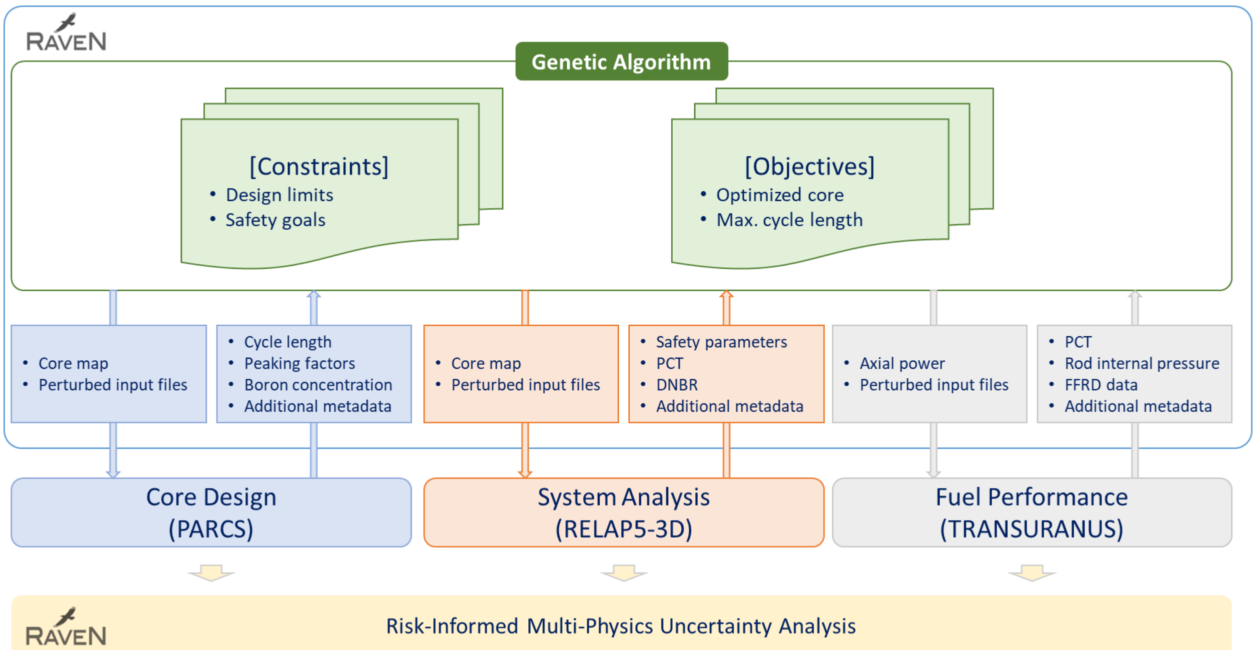

Figure 1-1. High level flow chart of LWRS-developed fuel reload optimization platform.

2. RAVEN INTERFACE FOR PARCS

2.1 Overview of PARCS

Developed by Purdue university, PARCS is a three-dimensional (3D) reactor core simulator designed to solve both the steady-state and time-dependent multigroup neutron diffusion equations and low-order transport equations in orthogonal and non-orthogonal geometries [4]. The cross section library is processed by independent module called GENPMAXS [[5]] by using the data generated from the lattice physics codes such as TRITON [[6]], HELIOS [7], or CASMO [8] into the PMAXS format readable by PARCS. PARCS also has coupling capabilities with thermal-hydraulics system codes such as TRACE [9] and RELAP5 [10]. The major features of the PARCS code are eigenvalue calculations, transient (kinetics) calculations, xenon transient calculations, decay heat calculations, pin power calculations, depletion calculations, and adjoint calculations.

2.2 Development of PARCS/RAVEN Coupling Interface

The interface development aims to use RAVEN to generate input and execute PARCS by applying a GA optimizer in RAVEN. Figure 2-1 shows data flow in the PARCS/RAVEN coupling interface. It is noted that the interface was designed to solve 17 × 17 reactor core of pressurized water reactor (PWR), thus, an additional update is needed for other types of reactor or core configurations.

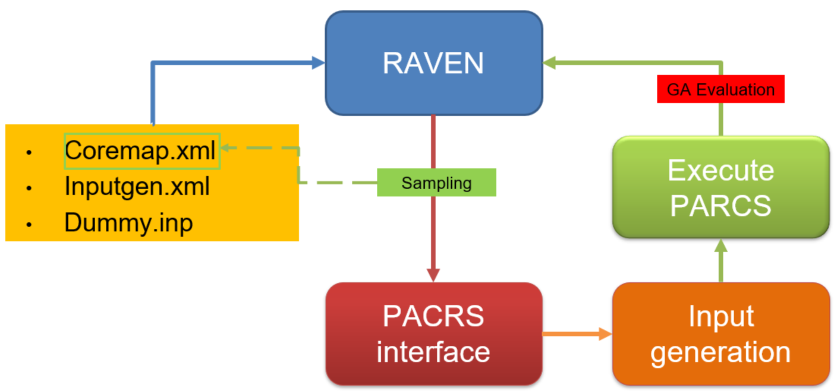

Figure 2-1. Data flow of the PARCS/RAVEN interface.

At the beginning of the simulation, user input files (`coremap.xml` `inputgen.xml` and `dummy.inp`) are checked by RAVEN for any input errors. With a GA optimizer, RAVEN samples fuel assembly location mapping and updates the `coremap.xml` file. Once sampling is completed, RAVEN sends input files to PARCS for execution. The resulting PARCS simulation is evaluated by the GA optimizer fitness function to produce the population of the next generation. This iterative process continues until the maximum number of generations, as specified in the RAVEN input, is reached. The simulation will then provide a final core design to the user which is the best possible solution out of all simulated possibilities.

The PARCS/RAVEN interface was built based on Python 3 computer language including following three files. The source script is not publicly available.

- `PARCSinterface.py`: Connects and interacts with the RAVEN main module
- `PARCSData.py`: Collects and extracts data from the depletion file and the pin power distribution file generated after each PARCS simulation. The list of data that can be extracted are:
    - Time-dependent multiplication factor k_{eff}

- Time-dependent F_Q
- Time-dependent $F_{\Delta H}$
- Time-dependent critical boron concentration
- Cycle length determined by the critical boron concentration being 10 ppm
- Time-dependent relative pin power distribution.

- `SpecificParser.py`: Generates PARCS input from sample loading pattern provided by RAVEN-GA module.

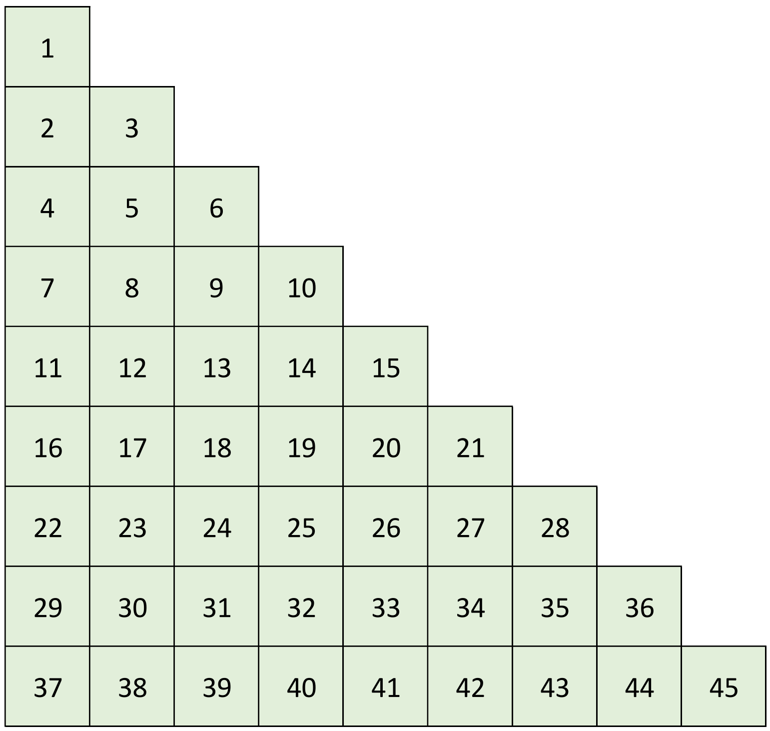

Figure 2-2. Index scheme example for 1/8 core-loading pattern.

The PARCS/RAVEN interface needs the following three input files:

- `coremap.xml`: The fuel assembly index scheme. An example of 1/8 core-loading pattern is shown in Figure 2-2 above.
- `dummy.inp`: Placeholder for perturbed PARCS input deck.
- `inputgen.xml`: Fuel assembly definition, cross section information and other specifications for PARCS.

2.3 Demonstration of PARCS/RAVEN Coupling Interface

2.3.1 Description PWR core optimization problem

A generic three-loop Westinghouse PWR core optimization problem was used for the PARCS/RAVEN coupling interface demonstration. The 17 × 17 core model consists of 157 fuel assemblies of five different fuel types as shown in Table 2-1.

Table 2-1. Fuel assembly inventory for the initial PWR core (Fuel type 1 refers to the reflector).

Fuel type	2	3	4	5	6
Enrichment (w/o)	2.0	2.5	2.5	3.2	3.2
Burnable poison	None	None	16 Gd rods	None	16 Gd rods

Figure 2-3 shows the fuel assembly design with and without Gd burnable poison loading. An initial 1/4 core-loading pattern with five different fuel types is illustrated in Figure 2-4. Fuel type index 1 refers to the reflector.

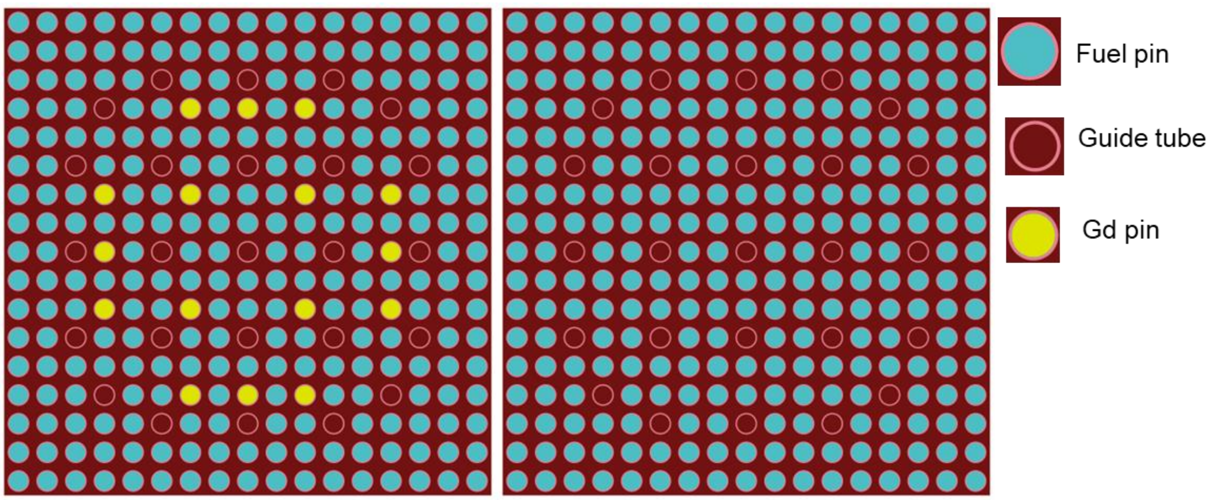

Figure 2-3. Pin map of a fuel assembly with (left) and without Gd burnable poison (right).

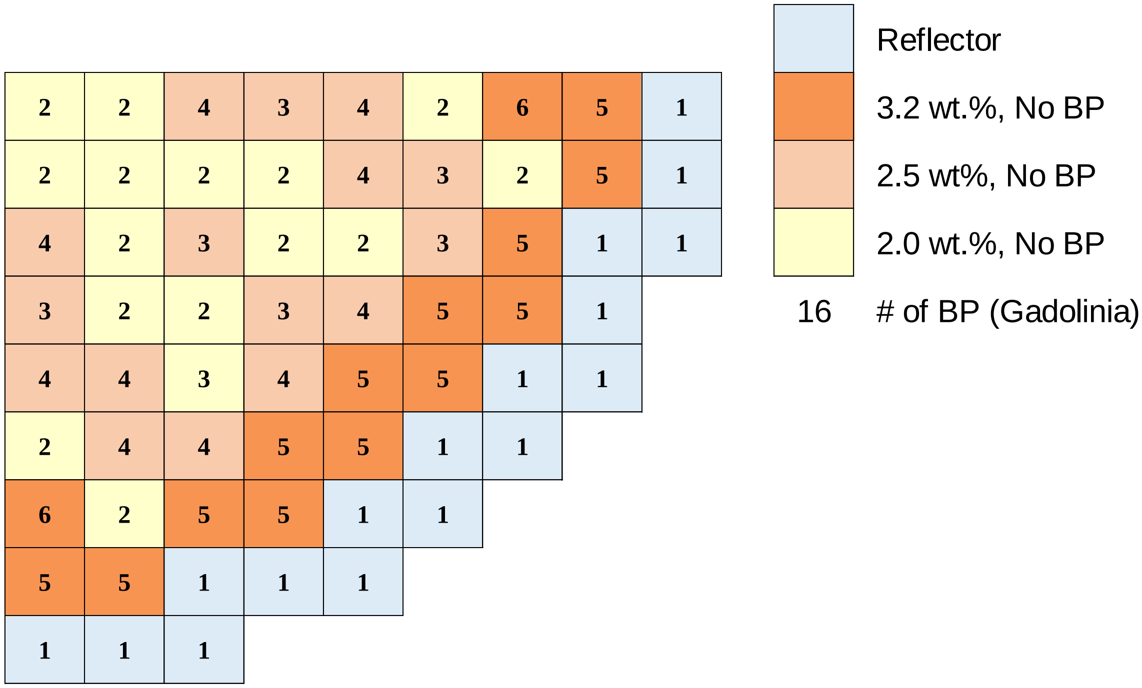

Figure 2-4. An initial 1/4 PWR core layout (the number indicates fuel type shown in Table 2-1).

The goals of the optimized core were set to minimize both peaking factors (i.e., $F_{\Delta H}$ and F_Q) and hot full power (HFP) boron concentration while maximizing cycle energy. The target constraint values for the design are $F_Q < 2.1$, $F_{\Delta H} < 1.48$ and HFP boron concentration <1300 ppm[1]. A single objective optimization method was used as a variable named `<target>`, representing the fitness function from following equation:

$$target = 1.0 \times L - 400.0 \times \max(0, F_{\Delta H} - 1.48) - 400.0 \times \max(0, F_Q - 2.1) - 1.0 \times \max(0, B - 1300)$$

where L is the cycle length and B is the peak critical boron concentration.

A weight factor of 400.0 was given to both F_Q and $F_{\Delta H}$ while 1.0 was given to boron concentration rate. The weight factor acts as a penalty that reduces the fitness of each proposed solution which violates the constraints of $F_Q < 2.1$ and $F_{\Delta H} < 1.48$ making it more important than violating the boron concentration constraint. Hence, this weight factor helps reducing the selection of constraints violated fitness in the next generation. It is noted that, in this fitness function definition, the weight assigned for each constraint is intuitively based on the evaluation of the user and heavily dependent on the nature of the problem.

The `inputgen.xml` of this problem follows:

```
<PARCS-input-gen>
   <THFlag> F </THFlag>
   <power> 100 </power>
   <coretype> PWR </coretype>
   <initialBoron> 1000 </initialBoron>
   <XSdir> Xsdir </XSdir>
   <Depdir> PARCSDEP </Depdir>
   <DepHistory> 1 1 1 1 18*30 </DepHistory>
```

[1] Current design limit of peaking factors of typical PWR are: $F_Q < 2.31$ and $F_{\Delta H} < 1.635$ [11].

```
    <NFA> 9 </NFA>
    <NAxial> 14 </NAxial>
    <FA_Pitch> 21.50 </FA_Pitch>
    <FA_Power> 16.3 </FA_Power>
    <Geometry> QUATER </Geometry>
    <grid_x> 1*10.75 8*21.50 </grid_x>
    <grid_y> 1*10.75 8*21.50 </grid_y>
    <grid_z> 30.48  12*30.48  30.48 </grid_z>
    <neutmesh_x> 1*1 8*1 </neutmesh_x>
    <neutmesh_y> 1*1 8*1 </neutmesh_y>
    <BC> 0 2 0 2 2 2 </BC>
    <FA-list>
     <FA name='FA1'  FAid ='0'  type ='20' structure = '1*6 12*1 1*7 FUEL'/>
     <FA name='FA2'  FAid ='1'  type ='30' structure = '1*6 12*2 1*7 FUEL'/>
     <FA name='FA3'  FAid ='2'  type ='40' structure = '1*6 12*3 1*7 FUEL'/>
     <FA name='FA4'  FAid ='3'  type ='50' structure = '1*6 12*4 1*7 FUEL'/>
     <FA name='FA5'  FAid ='4'  type ='60' structure = '1*6 12*5 1*7 FUEL'/>
     <FA name='REF'  FAid ='5'  type ='10' structure = '1*6 12*8 1*7 REFL'/>
     <FA name='NONE' FAid ='-1' type ='00' structure = ''/>
    </FA-list>
    <XS-list>
     <XS id='1' name='xs_g200_gd_0_bp_0'/>
     <XS id='2' name='xs_g250_gd_0_bp_0'/>
     <XS id='3' name='xs_g250_gd_16_bp_0'/>
     <XS id='4' name='xs_g320_gd_0_bp_0'/>
     <XS id='5' name='xs_g320_gd_16_bp_0'/>
     <XS id='6' name='xs_gbot'/>
     <XS id='7' name='xs_gtop'/>
     <XS id='8' name='xs_grad'/>
    </XS-list>
</PARCS-input-gen>
```

2.3.2 PWR core optimization results and analysis

The GA optimization was performed for two cases: 1) population size of 25 with 25 generations, and 2) population size of 50 with 50 generations[2]. It is noted that the number of populations is identical to the number of generations in the RISA pathway GA optimization platform. The comparison demonstrated that an increase in population and generation sizes results in better convergence of the results for parameters of interest.

Figure 2-5, Figure 2-6, and Figure 2-7 show results comparisons of fitness values, cycle lengths, boron concentrations, and peaking factors, respectively. Generally, convergence is observed as the number of generation (thus population) increases. In the case of fitness comparison, the gap between the worst (minimum) and the best (maximum) becomes narrower as iteration progresses, as shown in Figure 2-5. This indicates that the GA optimizer needs sufficiently large numbers of generation and population to provide optimal solutions.

[2] Population means a set of potential solutions. Generation means the number of iterations. More detail can be found in reference [1].

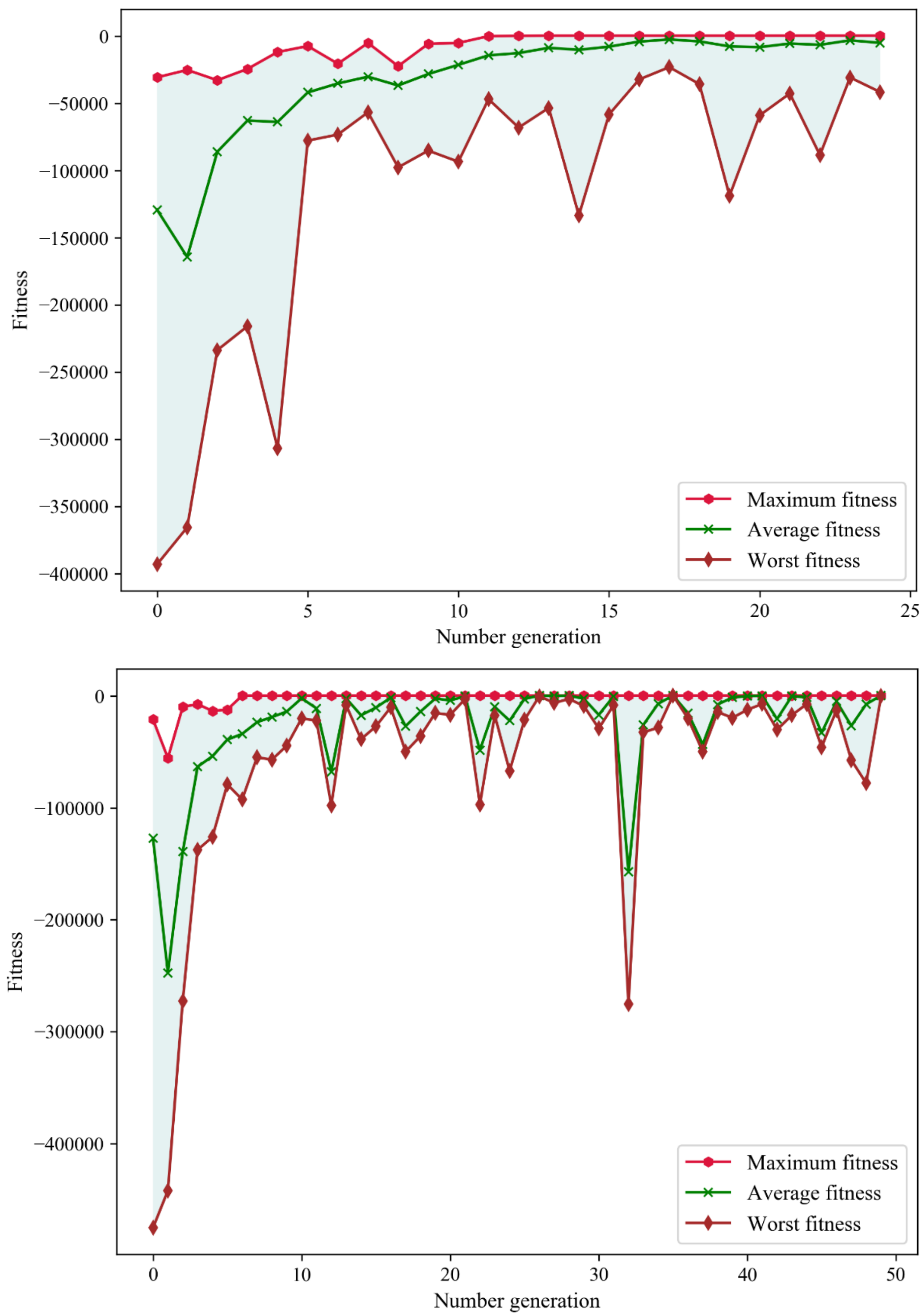

Figure 2-5. Fitness values as a function of 25 (top) and 50 (bottom) generation cases.

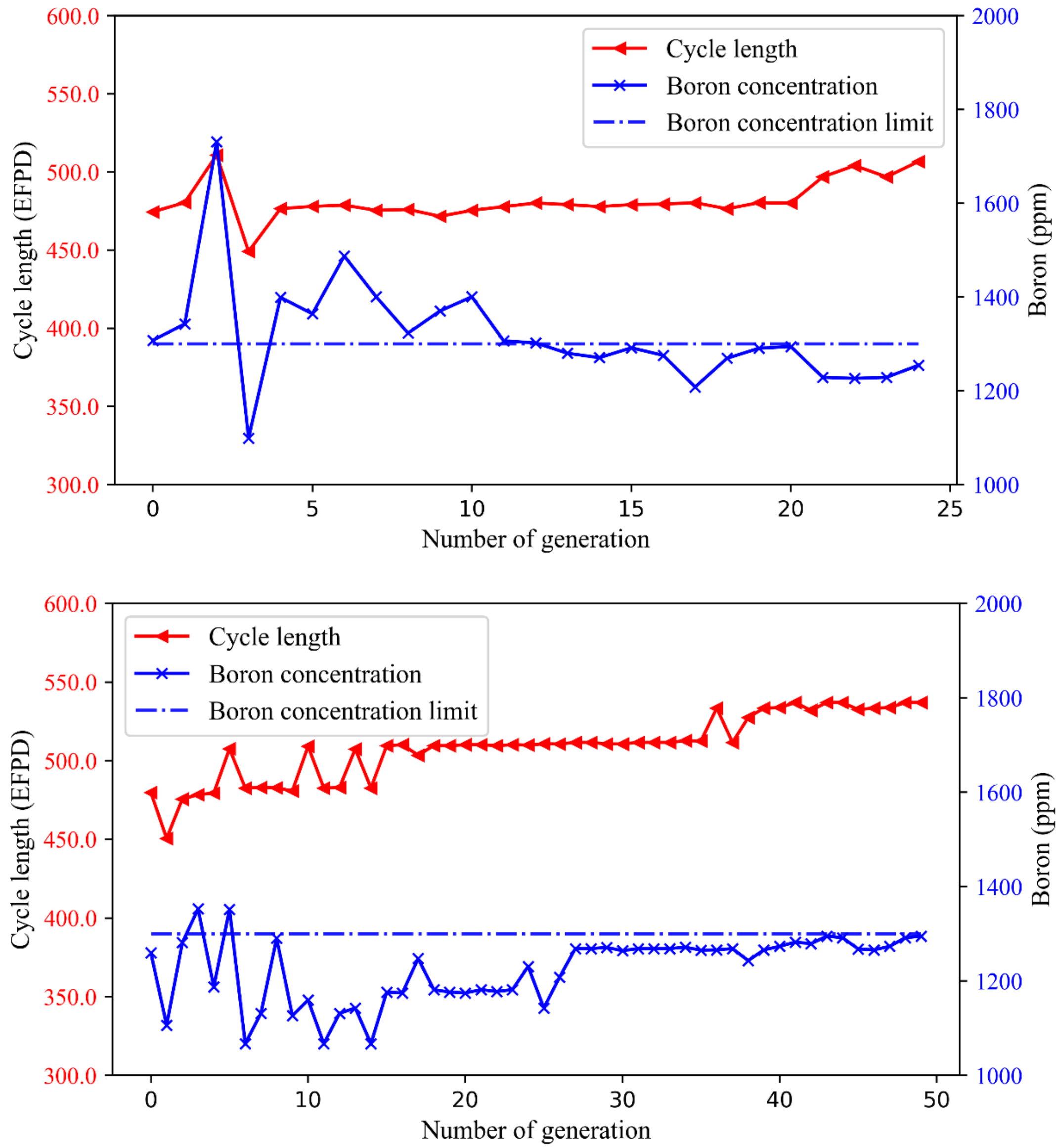

Figure 2-6. Cycle length and boron concentration in the best solution for 25 (top) and 50 (bottom) generation cases.

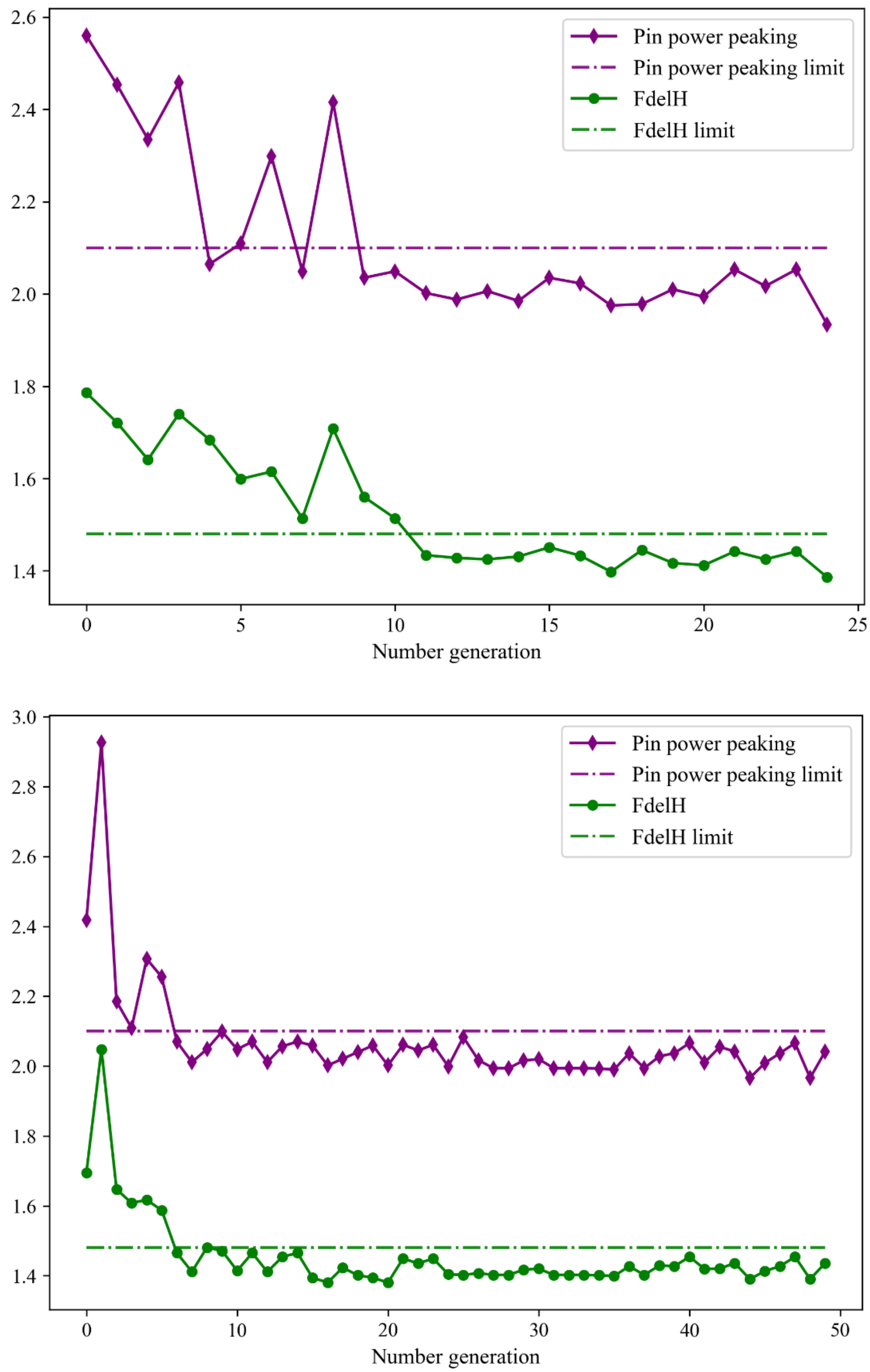

Figure 2-7. Pin power peaking and $F_{\Delta H}$ values in the best solution for 25 (top) and 50 (bottom) generation cases.

The convergence of constraint parameters (e.g., boron concentration and peaking factors) is also observed as the number of generations increases. The convergence starts earlier and with smaller oscillation sizes, as shown in Figure 2-6 and Figure 2-7.

The final value of the maximum cycle length with 50 generation cases was 537 effective full power days (EFPD) with an optimal core-loading pattern shown in Figure 2-8. For comparison, the initial core designed used for this problem allowed only 433 EFPD.

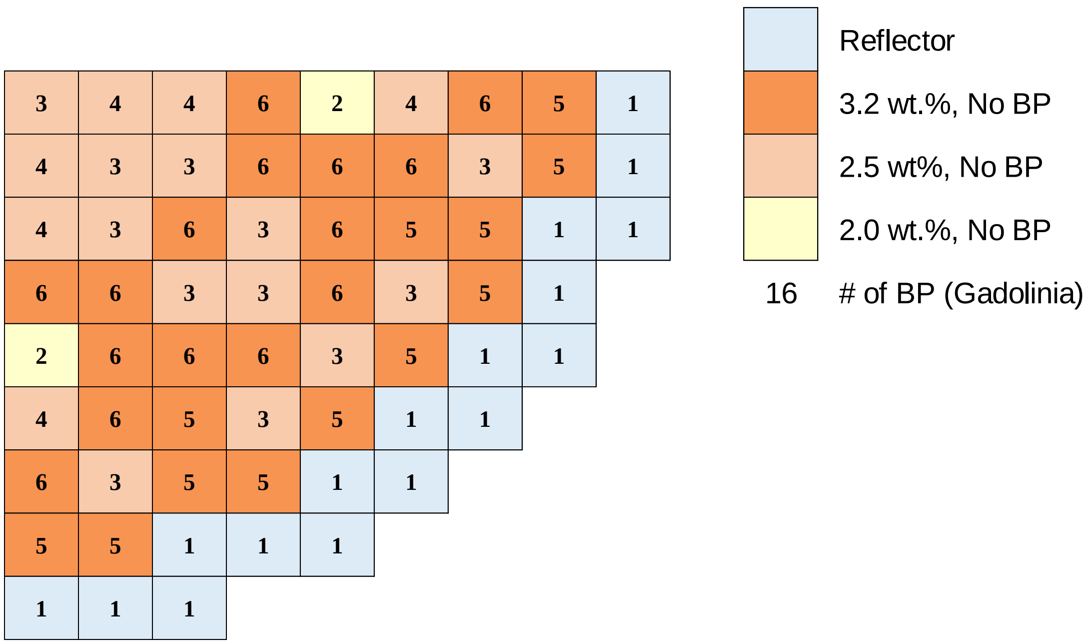

Figure 2-8. Optimal loading pattern from 50 generation cases (the number indicates fuel type shown in Table 2-1).

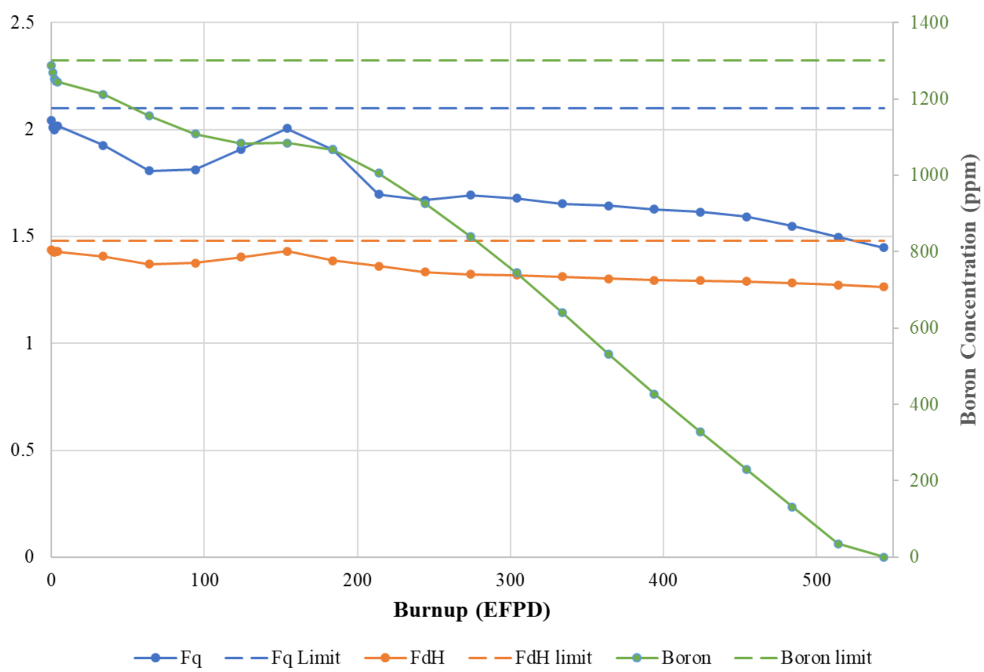

Figure 2-9. Core parameters change with depletion.

Figure 2-9 above shows the history of the major core design parameters as depletion time (i.e., burn up) progresses with all the parameters remaining below safety limitations during the fuel cycle.

The radial and axial relative power distributions of the optimal core are shown in Figure 2-10 and Figure 2-11, respectively. The power distributions is used for the system and fuel performance analyses as an initial and boundary conditions as shown in Figure 1-1.

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
1	1.21	1.03	1.02	1.09	0.94	0.74	0.70	0.49	0.00
2	1.03	1.34	1.37	1.19	1.06	0.94	0.78	0.42	0.00
3	1.02	1.37	1.39	1.44	1.19	1.29	0.84	0.00	0.00
4	1.09	1.19	1.44	1.43	1.15	0.99	0.58	0.00	
5	0.94	1.06	1.19	1.15	0.99	0.66	0.00	0.00	
6	0.74	0.94	1.29	0.99	0.66	0.00	0.00		
7	0.70	0.78	0.84	0.58	0.00	0.00			
8	0.49	0.42	0.00	0.00	0.00				
9	0.00	0.00	0.00						

Figure 2-10. Average radial relative power distribution at HFP (maximum value of 1.43 at position [3,6]).

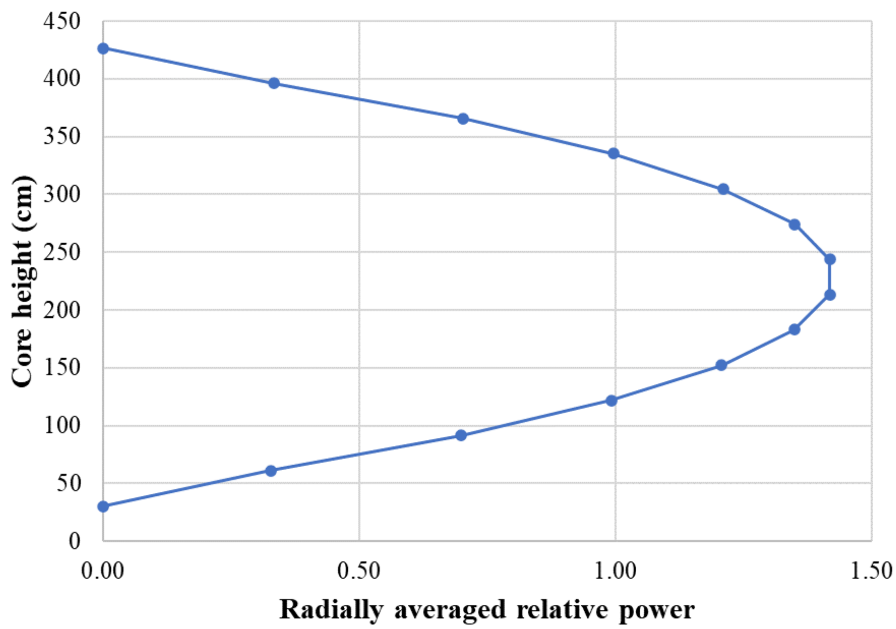

Figure 2-11. Axial relative power distribution at HFP.

Figure 2-12 shows full core optimized loading pattern at 537 EFPD.

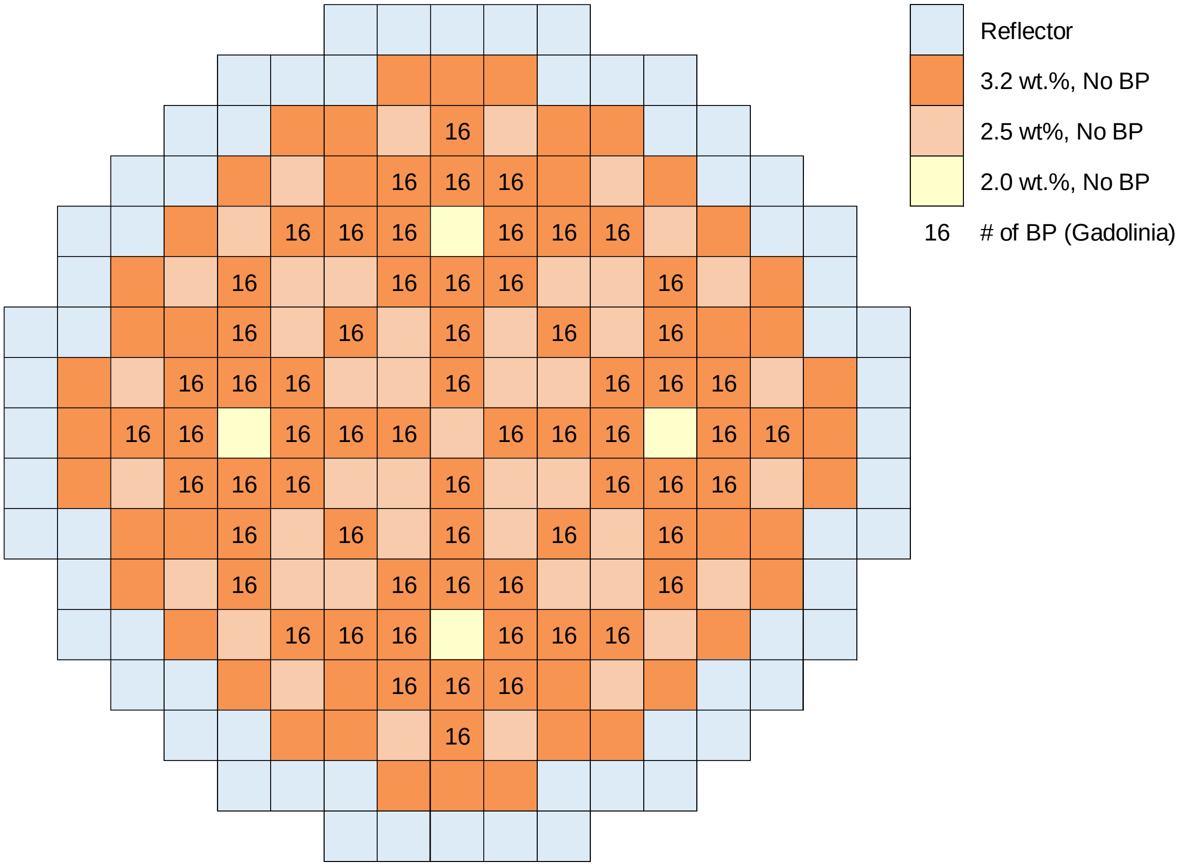

Figure 2-12. Optimized full core loading pattern at 537 EFPD (the numbers indicate number of Gadolinia burnable poisons).

3. RAVEN INTERFACE FOR TRANSURANUS

3.1 Overview of TRANSURANUS

TRANSURANUS is a fuel performance code developed at the Joint Research Centre's Institute for Transuranium Elements (ITU) in Karlsruhe, Germany [12]. The code approximates the fuel rod behavior with an axisymmetric, axially stacked, and one-dimensional radial approaches. The code can be employed for both steady-state and transient analyses and incorporates models that account for different and interrelated phenomena occurring in the fuel rod. The modeling of the fission gas behavior is a crucial aspect of nuclear fuel analysis in view of the related effects on the thermomechanical performance of the fuel rod, which can be particularly significant during transients.

During a loss-of-coolant accident (LOCA), the fuel cladding will further oxidize due to the exothermic reaction between the cladding and the steam. The run-away reaction is controlled by regulatory limits imposed on the maximum peak clad temperature of 2200ºF (from 10 CFR 50.46 acceptance criteria). The transient local oxidation can also be calculated by TRANSURANUS. Note that, in the case of a cladding burst, a double-sided oxidation may occur. That calculation also accounts for clad thinning due to the computed strain.

3.2 Development of TRANSURANUS/RAVEN Coupling Interface

TRANSURANUS input needs both core physics and thermal-hydraulics data. Hence, two sets of coupling interfaces were developed through RAVEN: TRANSURANUS / PARCS and TRANSURANUS / RELAP5-3D. The coupling was done as “loose coupling” to solve different disciplines or physics in each time step, thus, TRANSURUNUS and RELAP5-3D are interconnected via RAVEN but not dependent each other

For the core design, condition I, normal operation and operational transients, and condition II, fault of moderate frequency, events are typically considered safety aspects [13]. The power excursions occurring during these events may lead to xenon transients, which will translate into a spectrum of power shapes and peaking factors that must be considered in the safety analysis downstream. The limiting DBAs, which are mostly condition III, infrequent faults, and condition IV, limiting faults events, are then assessed based on core design results [13]. Finally, the fuel performance is analyzed based on both core design and DBA results. The analysis may include different modes of operation and specific reactor control mechanisms to assess the effect of power maneuvering.

The interface was built based on Python 3 computer language, including the following file. The source code script is not publicly available.

- `TUinterface.py`: Connects and interacts with RAVEN main module.

1.1.1 Coupling interface for PARCS

TRANSURANUS needs the following input data from the PARCS core design results to be used in the input file:

- Pin power histories (for each pin in every assembly modeled) in TRANSURANUS input file format[3]. An example TRANSURANUS input file is shown in Figure 3-1 which is required from full core design result. The breakdown pin power histories are:
    - Time-dependent, axially dependent linear heat rate (W/mm)[4]

[3] The required information from a RAVEN data object will be extracted following this structure

[4] A consistent set of axial power shape between PARCS and RELAP5-3D is necessary

- Time-dependent, axially dependent fast flux (neutrons/cm^2-s)
- Time-dependent steady-state system condition data including[5]
  - Coolant pressure
  - Coolant temperature
  - Coolant mass flow rate
  - Thermal-hydraulic diameter of the fuel rod channel.

- Two additional XML format input files that include fuel assembly data about what rods to simulate, each rod type (including enrichments and burnable poison content), and a pin power history map. It is noted that the file name could be defined by user.
  - `assembly_core.xml`: List of assemblies in the core and associated metadata including fuel assembly name and fuel assembly type or region.
  - `sample_region.xml`: Information of specific fuel assembly type (region) including pin maps denoting which pins have burnable poison or different enrichment levels.

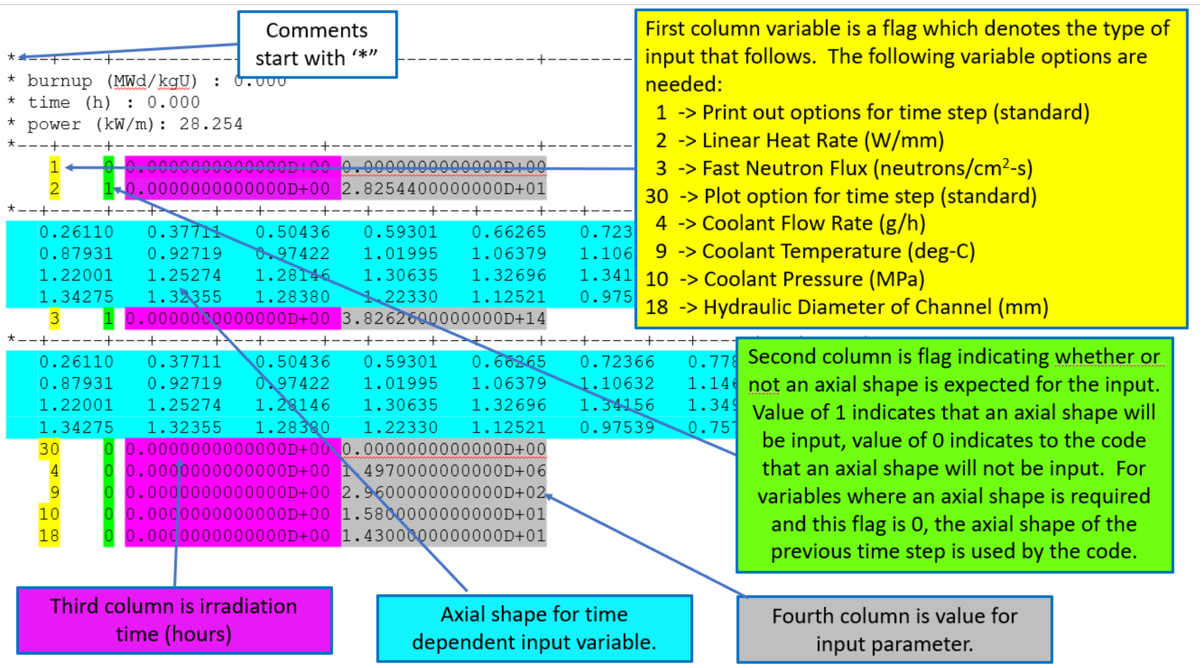

Figure 3-1. Sample input data for TRANURANUS/RAVEN coupling.

An alternative approach is to obtain the power shape information from the core design and translate the data into peaking factors which will be then used to define axial power shapes to feed into both the RELAP5-3D simulation and the TRANSURANUS calculation.

3.2.1 Coupling interface for RELAP5-3D

The necessary data from RELAP5-3D varies based on the limiting DBA scenario. In case of a LOCA analysis, TRANURANUS requires a value of F_Q, core average power, and burnup of each rod and average burnup. The list of necessary data needs to be identified for each limiting DBA scenario.

[5] These data could be propagated by RAVEN.

For a specific power history of a representative fuel pin generated from core design, the following data are needed from the RELAP5-3D simulation results. All these parameters are time-dependent inputs (i.e., derived for each time step).

- Coolant inlet temperature is calculated from the heat flux (`HTRNR`), the wall heat transfer coefficient (`HTHTC`), and the wall surface temperature (`HTTEMP`) by using the following equation[6]:

$$T_{cool} = \frac{-HTRNR}{HTHTC} + HTTEMP - 273.15$$

- Coolant pressure (not position dependent)
- Lower plenum temperature from the first value of `TEMPG`
- Upper plenum temperature from the second value of `TEMPG`.

3.2.2 Additional data for TRANSURANUS

For precise simulation results from TRANSURANUS, additional thermomechanical inputs may be necessary. This includes fuel as-built specifications such as geometry, materials, enrichment, and burnable absorber content data. A separate as-built specification input file can be created (referred to as a TRANSURANUS input header file) and mapped to individual rods based on the characteristics of fuel types (e.g., Westinghouse UO_2 rod, AREVA UO_2 rod, Gd rod, accident-tolerant fuel [ATF], high burnup, high enrichment, low enrichment).

3.2.3 Input and output control and plotting

The interface of TRANSURANUS was created using the generic code interface of the RAVEN XML input format. The current capabilities of the TRANSURAUS/RAVEN coupling interface are:

- Keyword-based substitution of fixed format inputs
- Running TRANSURANUS
- Running TuPlot, the TRAUNSURANUS plotting tool, and collecting time-dependent outputs.

The keyword-based input substitution allows the user to associate sampled parameters with keywords in the TRANSURANUS ASCII input file. The interface also requires that the card fixed format input specification is given. This allows the interface to perform the substitution while ensuring that the inputs will not produce errors by re-alignment. For example, `$rab` in the following line is the keyword and was set to represent a numerical input format of 8F10.5.

```
0.0       $rab      4.72000   5.43000   0.00160   0.0003
```

After the input substitution to real number 4.65598E0, the input becomes,

```
*** Editing line for keyword $rab
0.0       4.655980E04.72000   5.43000   0.00160   0.0003
```

One key element of the interface is that the replacement will appropriately round to fill the space. Note that in fixed format input, no spaces are needed between the input flags, as the inputs are simply read from a specific range of columns. The interface takes this into account and ensures that all inputs are written to coincide with the correct format.

[6] The axial node position of TRANSURANUS and RELAP5-3D needs to be the same, or the coolant inlet temperature needs to be re-meshed.

TuPlot is preferred for plotting the TRANSURANUS simulation considering following the aspects.

- TRANSURANUS outputs are nearly all multidimensional (functions of radius, axial section—also referred to as “slices”—and time) for a wide variety of output variables, therefore, designing a new user interface to pull data is somewhat challenging.
- TuPlot already has an input interface with an accompanying user’s manual
- TuPlot is generally included with TRANSURANUS distributions
- TRANSURANUS users would be expected to already have familiarity with TuPlot.

As such, the interface requires the user to supply a TuPlot input file and executable path, then it simply runs TuPlot with the outputs of the TRANSURANUS execution. Currently, the interface only supports time-dependent plots, but can be extended in the future for axial-section-dependent and radial-section-dependent plots.

The outputs are saved using the naming convention for TuPlot. The time-dependent plots can be uniquely identified with diagram numbers, curve numbers (these are essentially just different expressions of the plotted quantity), and axial section numbers. Therefore, contact pressure between fuel and cladding for the axial section can be saved by the interface to the output CSV file.

3.3 Demonstration of TRANSURANUS/RAVEN Coupling Interface

3.3.1 PWR UO2 fuel performance analysis

A demonstration case was performed with a typical PWR UO2 fuel. A single fuel rod was nodalized into 32 slices to simulate a steady-state test case. Two steps were included:

1. Using a Monte Carlo sampler in RAVEN, 20 cases were sampled following three normally distributed variables:
    - Outer radius of the fuel
    - Fuel local grain diameter
    - Fuel grain size diameter, averaged over the radius and burnup
2. Created the following two plots for verification:
    - Deformation of the fuel inner radius
    - Contact pressure between fuel and cladding (rod internal pressure).

The simulations and output gathering were completed successfully. Figure 3-2 and Figure 3-3 shows the deformation and internal pressure at Slice #3 which is mostly in the bottom part of the fuel rod. The figures show 20 different results from Monte Carlo sampling. Additional verification confirmed that sampled parameters were correctly substituted and the outputs were collected correctly.

It is noted that the demonstration case only focuses on the verification of using RAVEN to control TRANSURANUS. The result in Figure 3-2 and Figure 3-3 does not provide any meaningful representation of physical phenomena.

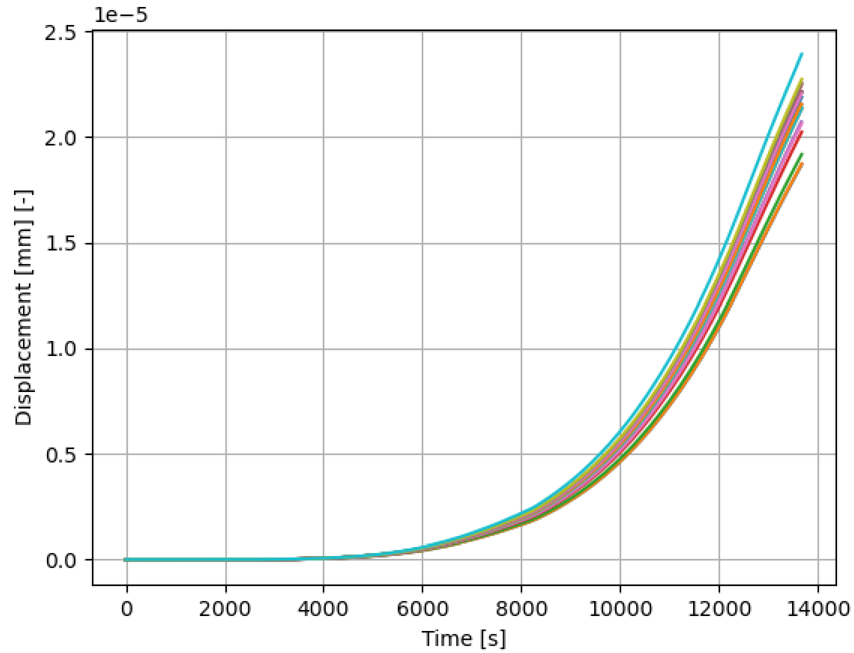

Figure 3-2. Deformation history of the fuel inner radius at Slice #3 (20 different data lines from Monte Carlo sampling).

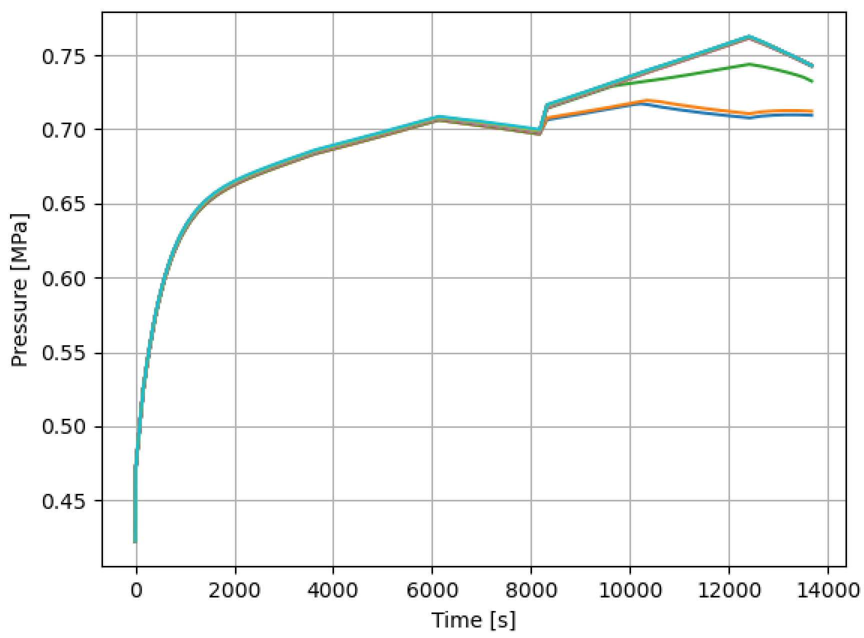

Figure 3-3. Contact pressure between fuel and cladding at Slice #3 (20 different data lines from Monte Carlo sampling).

4. SUMMARY AND FUTURE WORKS

The Python-based coupling interfaces were implemented in RAVEN to enable communication with the core simulator PARCS and fuel performance code TRANSURANUS. This feature is fully applicable to any type of fuel and fuel cycle configuration.

For the PARCS/RAVEN coupling, the GA optimizer was used to solve the core-loading pattern optimization problem. The required input data for the interface includes an initial core map with a proposed index scheme, simulation parameters for a PARCS input file, and a template file for perturbed input from RAVEN. To verify the PARCS/RAVEN coupling, a generic three-loop Westinghouse PWR 17 × 17 core with 157 fuel assemblies and five different fuel types was introduced to find an optimized core with design constraints such as peaking factors and boron concentration. It was obvious that a larger number of generations showed better optimization performance than smaller number of generations.

For TRANSURANUS/RAVEN coupling, the developed interface focused on RAVEN's capability to transition PARCS and RELAP5-3D output data into the TRANSURANUS input. The TRANURANUS/RAVEN coupling interface successfully controlled the TRANSURANUS simulation and post-processing by using TuPlot.

The following needs were identified, and these activities will be conducted during Fiscal Year 2023 (FY-23):

- Develop coupling interface between PARCS and RELAP5-3D. The current PARCS/RAVEN coupling interface is only focused on controlling PARCS and RAVEN and applying the GA optimizer. RELAP5-3D needs core design data directly from PARCS to perform a limiting DBA analysis and provide feedback to PARCS to verify that the designed meets safety criteria during accident conditions.
- The GA optimizer will be improved to allow using multiple objectives such as fuel cycle length, enrichment, burnable poisons, and multiple constraints including core design limits and system safety parameters. .
- Currently, the TRANSURANUS interface only saves output data for time-dependent variables. This needs to be extended to include radial-location-dependent and axial-location-dependent information.

5. REFERENCES

[1] Choi, Y.-J. et al., “Development and Demonstration of a Risk-Informed Approach to the Regulatory Required Fuel Reload Safety Analysis,” INL/EXT-22-68628, Idaho National Laboratory, 2022. https://doi.org/10.2172/1885790.

[2] Alfonsi, A., et al., “RAVEN Theory Manual,” INL/EXT-16-38178, Rev. 4, Idaho National Laboratory, Idaho Falls, ID, USA, 2021. https://doi.org/10.2172/1260312.

[3] Andersen, B. D., “A Machine-Learning Based Approach to Minimize Crud Induced Effects in Pressurized Water Reactors,” Dissertation, North Carolina State University, 2021. https://repository.lib.ncsu.edu/handle/1840.20/38656.

[4] Downar T., Xu Y., and Seker V., “PARCSv3.1 Theory Manual,” UM-NERS-09-001, October, 2012. https://www.nrc.gov/docs/ML1016/ML101610117.pdf.

[5] Ward A., Xu Y., and Downar T., “GenPMAXS v6.2, Code for Generating the PARCS Cross Section Interface File PMAXS,” U. of Michigan, 2016.

[6] Wieselquist W. A, et al., “SCALE Code System”, ORNL/TM-2005/39, Rev 6.2.4., April 2020.

[7] Wemple C. A., et al., “The HELIOS-2 Lattice Physics Code”, PHYSOR Conference, 2008.

[8] Smith K. S., et al., “SIMULATE-3: Advanced Three-Dimensional Two-Group Reactor Analysis Code User’s Manual”, Studsvik Scandpower, Studsvik/SOA-95/15 Rev 2, 1995.

[9] Miller, R. and Downar, T., “Completion Report for the TRAC-M-SPECIFC Data Map Routine in the Coupled TRAC-M/PARCS code,” PU/NE-99-9, Purdue University, June 1998.

[10] Barber, D. and Downar, T., “Completion Report for the Coupled RELAP5/PARCS Code,” PU/NE-98-31, Purdue University, Nov. 1998.

[11] Morita, T., et al., “Topical report: power distribution control and load following procedures.” Westinghouse Electric Corporation. September 1974. http://doi.org/10.2172/4242391.

[12] TRANSURANUS HANDBOOK, Version V1M2J19, January 2019. (Available to licensee only)

[13] U.S. Nuclear Regulatory Commission, “Standard Review Plan for the Review of Safety Analysis Reports for Nuclear Power Plants: LWR Edition — Transient and Accident Analysis”, NUREG-0800, Chapter 15, 2007

[14] Bordelon, F.M., et al., “Westinghouse reload safety evaluation methodology.” (WCAP—9273). Westinghouse Electric Corp., Pittsburgh, PA, 1978.