Indira Gandhi Centre for Atomic Research
Department of Atomic Energy

Simulation and Modelling of Uranium Burnup Using Neutron Reaction Networks and ODEs

TRAINING REPORT

submitted by

ABHINAND SATISH

under the guidance of

Mr. Anupam Chakraborty
RSS/RS&ATSD/EPG/RDG

Project Overview:

This project details the development and implementation of a numerical simulation to model the burnup of Uranium chain-based fuel and its subsequent transmutation chain. How this would happen is by leveraging data from the neutron flux distributions, reaction cross-sections, and decay constant, a system of coupled first-order differential equations derived from Stacey's "Nuclear Reactor Physics" book. The final model of this project will allow the user to input a file where a list of isotopes and their subsequent concentrations and neutron flux which would compute the evolution of the isotopic concentrations over time. With that in mind here is a list of requirements for the simulations to run:

- All 19 differential equations from Stacey's textbook
- Neutron cross sections for each reaction type for the isotopes that are involved
- Materials of Fuel and their concentrations
- Using an input flux to calculation the reaction rate of each reaction

The above is basically a layout for the process that I used to calculate the isotopes produced and the specific outputs will be explained. Keep in mind that the goal of this project is to be able to show what the isotopes are and their concentrations in a particular time frame. This is a very common requirement as Nuclear Reactions have burnup and cool down times, so it is important to understand what your fuel looks like.

Stacey's Ordinary Differential equations

In chapter 6.1 – changes in Fuel composition, page 199, we can see the Fuel Depletion-Transmutation-Decay Equations. The concentrations of the various fuel isotopes in a reactor are described by a coupled set of production-destruction equations. In the following equations three is going to be a two-digit superscript to identify isotopes in which the first digit is the last digit in the atomic number and the second digit is the last digit in the atomic mass. We represent the neutron reaction rate by $\sigma^{nm}{}_x \varphi n^{nm}$, although the actual calculation may involve a sum over energy groups of such terms. Here is the list of isotopic concentrations are described by:

1. $\frac{\partial n^{24}}{\partial t} = -\sigma_a{}^{24} \emptyset n^{24}$
2. $\frac{\partial n^{25}}{\partial t} = \sigma_\gamma{}^{24} \emptyset n^{24} - \sigma_a{}^{25} \emptyset n^{25}$
3. $\frac{\partial n^{26}}{\partial t} = \sigma_\gamma{}^{25} \emptyset n^{25} - \sigma_a{}^{26} \emptyset n^{26} + \lambda^{36}{}_{ec} n^{36}$
4. $\frac{\partial n^{27}}{\partial t} = \sigma_\gamma{}^{26} \emptyset n^{26} + \sigma_{n,2n}{}^{28} \emptyset n^{28} - \lambda^{27} n^{36}$
5. $\frac{\partial n^{28}}{\partial t} = -\sigma_a{}^{28} \emptyset n^{28}$
6. $\frac{\partial n^{29}}{\partial t} = \sigma_\gamma{}^{28} \emptyset n^{28} - (\lambda^{29} + \sigma_a{}^{29} \emptyset) n^{29}$
7. $\frac{\partial n^{36}}{\partial t} = \sigma_{n,2n}{}^{37} \emptyset n^{37} - (\lambda^{36} + \sigma_a{}^{36} \emptyset) n^{36}$
8. $\frac{\partial n^{37}}{\partial t} = \lambda^{27} n^{27} - \sigma_a{}^{27} \emptyset n^{37}$
9. $\frac{\partial n^{38}}{\partial t} = \sigma_\gamma{}^{37} \emptyset n^{37} - (\lambda^{38} + \sigma_a{}^{38} \emptyset) n^{38}$
10. $\frac{\partial n^{39}}{\partial t} = \lambda^{29} n^{29} - (\lambda^{39} + \sigma_a{}^{39} \emptyset) n^{39}$

11. $\frac{\partial n^{48}}{\partial t} = \lambda^{38} n^{38} - {\sigma_a}^{48} \emptyset n^{48}$
12. $\frac{\partial n^{49}}{\partial t} = \lambda^{39} n^{39} - {\sigma_a}^{49} \emptyset n^{49} + {\sigma_\gamma}^{48} \emptyset n^{48}$
13. $\frac{\partial n^{40}}{\partial t} = {\sigma_\gamma}^{49} \emptyset n^{49} - {\sigma_a}^{40} \emptyset n^{40} + {\sigma_\gamma}^{29} \emptyset n^{29} + {\sigma_\gamma}^{39} \emptyset n^{39}$
14. $\frac{\partial n^{41}}{\partial t} = {\sigma_\gamma}^{40} \emptyset n^{40} - (\lambda^{41} + {\sigma_a}^{41} \emptyset) n^{41}$
15. $\frac{\partial n^{42}}{\partial t} = {\sigma_\gamma}^{41} \emptyset n^{41} - {\sigma_a}^{42} \emptyset n^{42}$
16. $\frac{\partial n^{43}}{\partial t} = {\sigma_\gamma}^{42} \emptyset n^{42} - (\lambda^{43} + {\sigma_a}^{43} \emptyset) n^{43}$
17. $\frac{\partial n^{51}}{\partial t} = \lambda^{41} n^{41} - (\lambda^{51} + {\sigma_a}^{51} \emptyset) n^{51}$
18. $\frac{\partial n^{52}}{\partial t} = {\sigma_\gamma}^{51} \emptyset n^{51} - {\sigma_a}^{52} \emptyset n^{52}$
19. $\frac{\partial n^{53}}{\partial t} = \lambda^{43} n^{43} - {\sigma_a}^{53} \emptyset n^{53} + {\sigma_\gamma}^{52} \emptyset n^{52}$

These 19 differential equations are 19 isotopes that are found in a Uranium Decay chain. Here is a list of all the isotopes and the codes to them so it is easy to correspond each equation to each isotope.

Isotope	**Code**
U234	24
U235	25
U236	26
U237	27
U238	28
U239	29
Np236	36
Np237	37
Np238	38
Np239	39
Pu238	48
Pu239	49
Pu240	40
Pu241	41
Pu242	42
Pu243	43
Am241	51
Am242	52
Am243	53

So how we interact with these differential equations is by building a function that can include production and loss terms based on the isotope. For instance, we have an input isotope and then build production and loss terms based on that isotope from data stored. This would be easier than having to solve each differential equation every time rather than building the equations on a case-by-case basis.

Flux normalization

The flux input was read from a file (`Flux.txt`) containing raw flux values across 175 energy groups. These raw values were then normalized to create a flux spectrum that describes the relative distribution of neutrons across energy:

```
flux = pd.read_csv('Flux.txt', header=None).squeeze().values
flux_sum = np.sum(flux)
spectrum = flux[:] / flux_sum
```

This normalized spectrum ensures that the shape of the flux is preserved (important for spectral interactions), but the absolute magnitude can still be controlled independently via a scalar value from the input file (the total flux).

Applying time dependent flux

The user-provided `Total flux` and `Flux fractions` were used to reconstruct effective flux for each time interval. This is important because different intervals in the depletion simulation may experience different irradiation intensities. This was implemented as:

```
interval_fluxes = (spectrum * total_flux).reshape(spectrum.shape[0], 1) *
np.array([flux_fractions])
```

This results in a 175×n matrix, where n is the number of intervals. Each column represents the full multigroup flux for a one time period.

Cross section implementation with flux

Each reaction type (absorption, capture, (n,2n)) has group-wise microscopic cross-section data. To apply these cross-sections properly, they were dot-multiplied with the effective flux for the current time interval. This gives an effective reaction rate per nucleus for that reaction type and isotope:

```
(nuclear_data[rxn][iso] * evaluation_flux).sum()
```

This summation across all 175 groups yields a scalar representing the group-collapsed reaction rate, incorporating both spectral shape and total neutron intensity. This was repeated for each reaction channel in the ODE.

ODE System Construction

The ODEs represent a coupled system where each isotope's concentration evolves due to both production and loss terms. The production terms include contributions from decay or neutron reactions from parent isotopes. The loss terms include decay and neutron reactions that deplete the isotope.

In the ODE function:

* Gain terms are calculated based on the reaction cross sections of the parent isotope multiplied by its concentration and the evaluated flux.
* Loss terms are calculated by applying the flux-weighted cross section for the isotope being evaluated and multiplying by its concentration.

```
for iso, rxn in production:
if rxn == 'decay':
gain += nuclear_data[rxn][iso] * y[index_map[iso]]
else:
gain += (nuclear_data[rxn][iso] * evaluation_flux).sum() * y[index_map[iso]]
```

This approach maintains full fidelity to energy-dependent behavior while simplifying the equations to a 0-D (no spatial dependence) burnup calculation.

Solving the ODEs and Generating Results

The ODE system was solved interval by interval using `solve_ivp` with the `BDF` method for stiff equations. The concentrations obtained from the solution were converted from atoms/cm^3 to grams using molar masses:

result_dict = {iso: (N[index_map[iso],:] / 6.022e23) * molar_masses[iso] for iso in isotope_list}

This allowed the model to track the mass of each isotope over time. These outputs were plotted and saved for interpretation and comparison.

Results

I have used three scenarios as inputs for my code to test. 1st is natural uranium cycle, 2nd is plutonium cycle which is used in PFBR reactors and 3rd is a mixed composition fuel. There three inputs are as follows:

Example 1 input file

Isotopes : U235, U238

Isotope masses (g) : 71.0, 9929.0

Time units : d

Time intervals : 180, 60, 180, 60, 180, 60

Time step : 10

Flux fractions : 1.0E-00, 1.0E-10, 1.0E-00, 1.0E-10, 1.0E-00, 1.0E-10

Total flux : 8.0E+15

Example 2 input file

Isotopes :Pu239, Pu240, Pu241, Pu242

Isotope masses (g) : 6500.0, 2500.0, 700.0, 300.0

Time units : d

Time intervals : 180, 60, 180, 60, 180, 60

Time step : 10

Flux fractions : 1.0E-00, 1.0E-10, 1.0E-00, 1.0E-10, 1.0E-00, 1.0E-10

Total flux : 8.0E+15

Example 3 input file

Isotopes : U235, U238, Pu239, Pu240, Pu241, Pu242

Isotope masses (g) : 56.3, 7873.7, 1345.5, 517.5, 144.9, 62.1

Time units : d

Time intervals : 180, 60, 180, 60, 180, 60

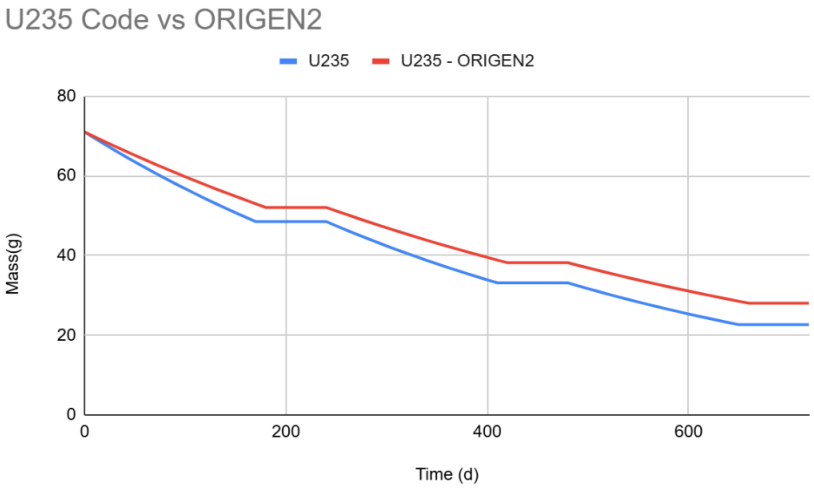
Time step : 10

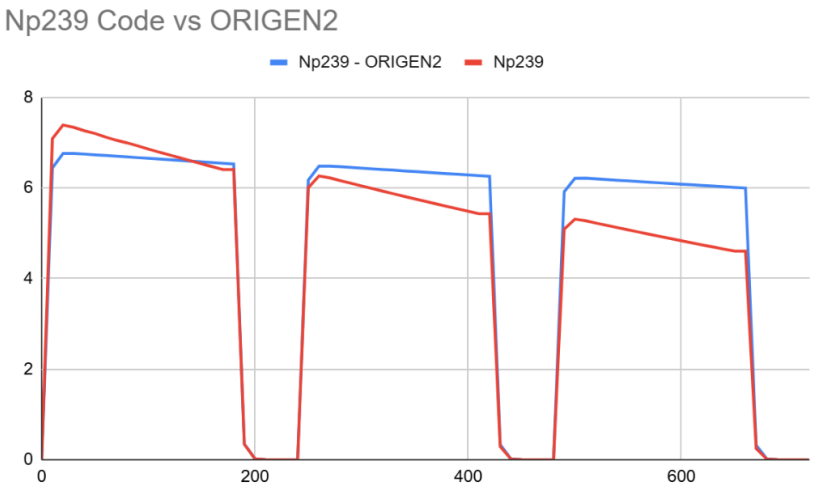
Flux fractions : 1.0E-00, 1.0E-10, 1.0E-00, 1.0E-10, 1.0E-00, 1.0E-10

Total flux : 8.0E+15

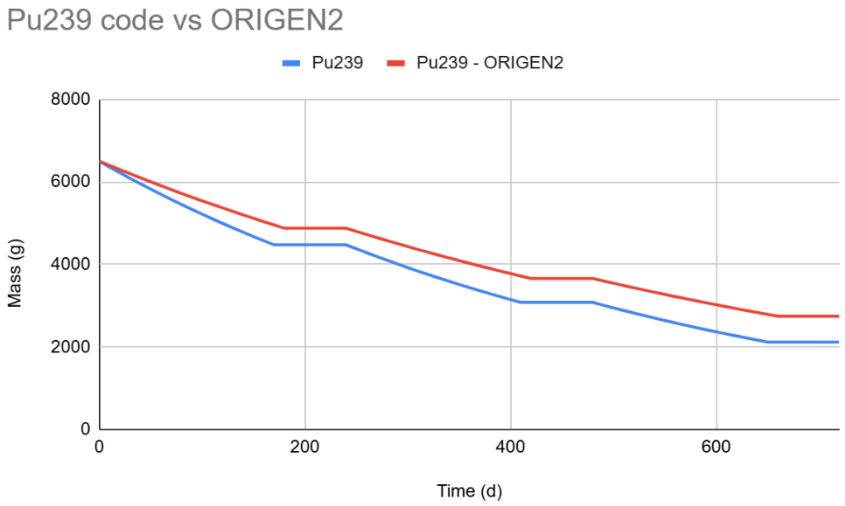
The respective outputs for these input files show that my code has a similar trend to the verification software ORIGEN2 which is used. Although there is a difference is the amount of concentration of masses at the end of the cycle the trend is the same therefore showing a validation of my code. Below is the graphs for each input scenario:

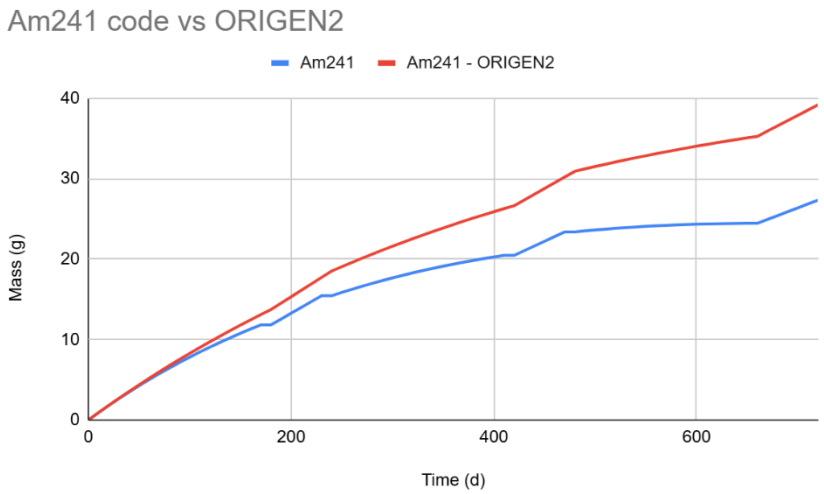
Input1:



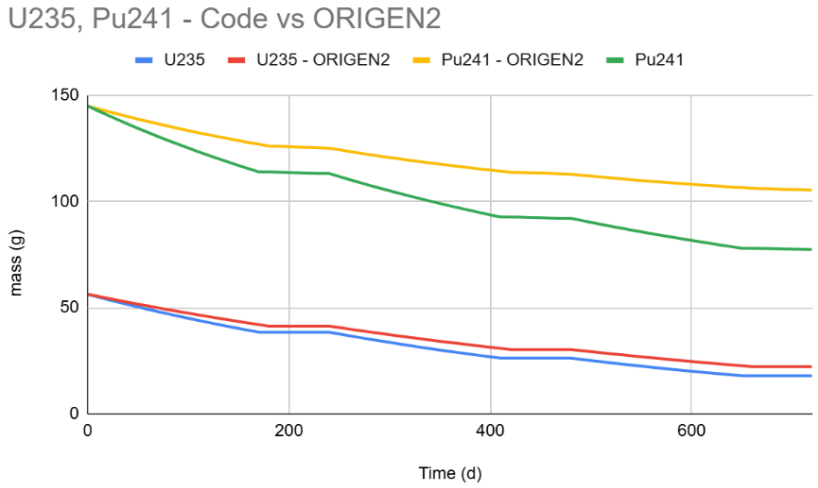


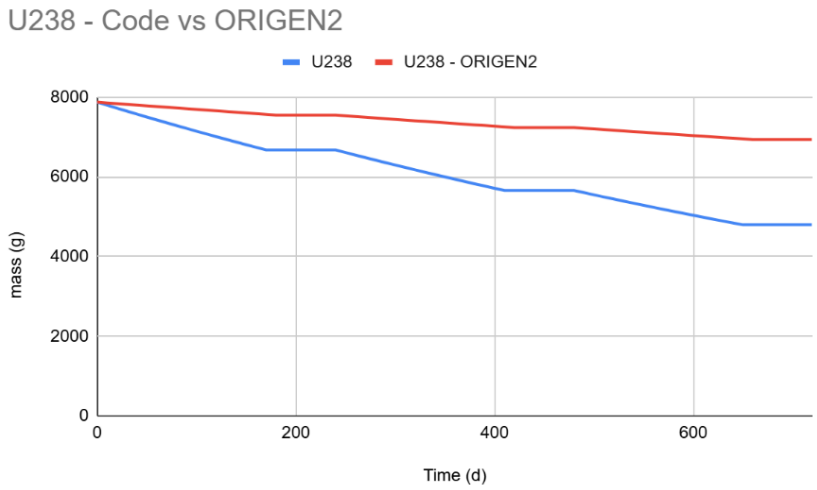
Input2:





Input3:





Below is the final concentration values for each input scenario:

Final Isotope Concentration

Isotope	Input 1 (g)		Input 2 (g)		Input 3 (g)	
	Code	ORIGEN2	Code	ORIGEN2	Code	ORIGEN2
U234	0.00E+00	2.41E-02	0.00E+00	2.76E-02	0.00E+00	2.48E-02
U235	2.26E+01	2.80E+01	0.00E+00	1.49E-01	1.79E+01	2.23E+01
U236	5.96E+00	8.83E+00	0.00E+00	4.94E-01	4.73E+00	7.10E+00
U237	1.45E-04	2.48E-04	0.00E+00	1.77E-05	1.15E-04	2.01E-04
U238	6.05E+03	8.75E+03	0.00E+00	1.15E-03	4.80E+03	6.94E+03
U239	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
Np236	6.93E-05	2.54E-04	0.00E+00	3.30E-06	5.50E-05	2.02E-04
Np237	2.56E+00	4.59E+00	0.00E+00	8.89E-02	2.03E+00	3.66E+00
Np238	0.00E+00	2.88E-10	0.00E+00	2.07E-07	0.00E+00	4.31E-08
Np239	9.98E-08	1.36E-07	0.00E+00	3.58E-05	7.92E-08	7.52E-06
Pu238	6.37E-01	1.20E+00	0.00E+00	5.40E+00	5.05E-01	2.07E+00
Pu239	5.16E+02	6.80E+02	2.12E+03	2.75E+03	8.48E+02	1.11E+03
Pu240	5.00E+01	7.06E+01	1.84E+03	2.48E+03	4.20E+02	5.70E+02
Pu241	2.55E+00	3.81E+00	3.64E+02	4.94E+02	7.73E+01	1.05E+02
Pu242	1.07E-01	1.84E-01	2.29E+02	3.06E+02	4.75E+01	6.35E+01
Pu243	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
Am241	6.89E-02	1.08E-01	2.73E+01	3.92E+01	5.72E+00	8.20E+00
Am242	6.53E-03	1.72E-08	6.71E+00	1.47E-05	1.40E+00	3.06E-06
Am243	3.35E-03	7.48E-03	2.49E+01	4.16E+01	5.17E+00	8.62E+00
Total mass(g)	**6.65E+03**	**9.54E+03**	**4.61E+03**	**6.12E+03**	**6.23E+03**	**8.83E+03**

Post-Simulation Analysis

After solving the ODE system, the resulting isotopic masses were analyzed across each defined time interval. This temporal evolution reveals how uranium isotopes are gradually transmuted into neptunium, plutonium, and americium through successive neutron capture and decay events. For instance, U-238, which has a high capture cross-section for fast neutrons, begins producing U-239, which subsequently decays into Np-239 and further into Pu-239 — a key fissile isotope. The model clearly shows how irradiation cycles lead to actinide buildup and fuel evolution, which are important factors in nuclear waste management, reactor core design, and reprocessing strategies.

To ensure numerical accuracy and stability, adaptive time-stepping and stiff solvers were used. Further validation could be performed by comparing these outputs against ORIGEN2 or SCALE system simulations using the same inputs.

Conclusion

This simulation framework successfully demonstrates the numerical modeling of uranium burnup through coupled ODEs and multigroup neutron flux interactions. By integrating energy-dependent reaction data with time-varying flux profiles, the system captures both the dynamic behavior and complexity of nuclear fuel transmutation. This tool allows users to predict isotopic inventory over time, making it valuable for reactor operation analysis, waste forecasting, and nuclear forensics.

Future work can expand this model to include spatial dependence, delayed neutron precursors, and radiotoxicity indices. Additionally, incorporating experimental or validated benchmark data would further enhance its applicability in research and regulatory settings.