NUEN 647 Final Project

Uncertainty quantification of depletion calculations for specific isotopes using ORIGEN2.

I Introduction

Determing composition of irradiated fuel is of importance for a myriad of reasons. Whether for flux calculations, reprocessing, or irradiation history verification, calculating fuel composition requires a Bateman solver, and a means for building a sparse matrix.

Applications using these compositions rarely report the uncertainty associated with results, even when inputs, such as flux shape, fission yield, cross sections, and half-lives have varying degrees of uncertainty. Further sources of error in this calculation are due to the multi-group approximation, and single point approximation, but will not be explored here.

Several isotope concentrations, shown in Table 1, were calculated as a function of burnup with the depletion code ORIGEN2 for a PWR system with 3 Wt% enriched uranium. ORIGEN2 solves the bateman equations with the matrix expoential method and requires a library with decay and cross section information. Cross sections and fission product yields are reduced to single group through flux averaging before execution of the code, with the assumption that the flux has the same shape as a typical PWR.

The uncertainty of concentrations were determined by varying the cross section, fission yeild, and half-life information that was fed into ORIGEN2. The uncertainties on cross sections were determined by calculating the range of the single group cross section, taking the mid point as a mean, the range as a standard deviation, and assuming a Gaussian distribution. A similar calculation was done for the fisison yields, and half-life distributions were taken at face value as normally distributed.

Table 1: Isotope solve list.

^{133}Cs	^{136}Ba	^{153}Eu
^{134}Cs	^{138}Ba	^{154}Eu
^{135}Cs	^{149}Sm	^{239}Pu
^{137}Cs	^{150}Sm	^{242}Pu
^{148}Nd	^{106}Rh	^{125}Sb

II Objectives

- ☑ Build ORIGEN2 model for thermal system which calculates concentrations of isotopes shown in Table 1.

Listing 1: PWR Input Deck

```
-1
-1
-1
  RDA  Irradiation of 1 MT of PWR fuel
```

```
  RDA  Fuel enrichment is 3.0 w/o U-235
  RDA
  LIB  0  1 2 3  601 602 603  9  50 0 1 38
  PHO     101 102 103  10
  INP  1   1  -1  -1   1   1
  BUP
  IRP 100.0 37.5 1 2 4 2 BURNUP=3,750 MWD/MT
  IRP 200.0 37.5 2 2 4 0 BURNUP=7,500 MWD/MT
  IRP 300.0 37.5 2 2 4 0 BURNUP=11,250 MWD/MT
  IRP 400.0 37.5 2 2 4 0 BURNUP=15,000 MWD/MT
  DEC 500.0      2 3 4 0 DECAY FOR 100.0 DAYS
  DEC 4150.0     3 4 4 0 DECAY FOR 10 YEARS
  DEC 73500.0    4 5 4 0 DECAY FOR 200.0 YEARS
  BUP
  OPTL 24*8
  OPTA 4*8 5 19*8
  OPTF 4*8 5 19*8
  OUT     5   1  -1   0
  END
2 922340 270. 922350 30000. 922380 969730. 0 0.0
0
```

The model irradiates 1 metric ton of US PWR fuel for a single cycle (15,000 MWd/Mt). The calculations use a constant power assumption of 37.5 W/g. The model does not include the oxygen because we are not interested in the activation of oxygen. Cross section modifcation throughout the calculation use the changing flux associated with a US PWR.

Initial verification of the model analyzed the end concentration of ^{137}Cs and calculated the burn-up from that value. This calculation does not have an exact value for the yield of ^{137}Cs and is used qualitatively as a sanity check.

$$\frac{552.8 \text{ g } ^{137}\text{Cs}}{Mt} \cdot \frac{6.022E23 \text{ atoms}}{137 \text{ g } ^{137}\text{Cs}} \cdot \frac{\text{Fission}}{0.06 \text{ atoms}} \cdot \frac{200 \text{ MeV}}{\text{Fission}} \cdot \frac{1.602E-19 \text{ MJ}}{1 \text{ MeV}} \cdot \frac{1 \text{ day}}{86400 \text{ s}} = 15,018 \frac{\text{MWd}}{Mt}$$

☑ Determine how to vary cross section and or flux spectrum inputs for calculation

☑ Determine how to vary fission yields for calculation

☑ Determine how to vary half-life information for calculation

ORIGEN2 reads in cross section information through a file named "TAPE9.inp", specified by the 8th input on the LIB card. "TAPE9.inp" needs at least 3 cross section libraries. These are specified by the 5th 6th and 7th inputs on the LIB card as 601, 602, and 603 for the activation products, actinides, and fission products, respectively. The input for ^{137}Cs from library 603 is shown in the listing below, with a corresponding key shown in Table 2 [1].

Listing 2: ^{137}Cs cross section library 603 input

```
603   551370 2.546E-02 0.0        0.0        0.0        0.0        0.0          1.0
603      1.81E-02 6.78E-01 1.25E-01 2.08E-01 7.56E-01 9.45E-02 9.43E-02 9.43E-02
```

Table 2: Key to parameters in cross section library

LIB	NUCLID	(n,γ)	(n,2n)	(n,α)	(n,p)	(n,γ^*)	(n,2n*)	YYN
LIB	Y(^{232}Th)	Y(^{233}U)	Y(^{235}U)	Y(^{238}U)	Y(^{239}Pu)	Y(^{241}Pu)	Y(^{245}Cm)	Y(^{249}Cf)

Both σ_γ and yield will be modified based on the locations in the cross section libraries shown above. Half-life information is contained in the decay libraries 1, 2, and 3 for activation products, actinides, and fission products, respectively. The input for ^{137}Cs from library 3 is shown in the listing below, with a corresponding key shown in Table 3 [1].

Listing 3: ^{137}Cs cross section library 603 input

```
   3   551370   1      9.467E+08 9.460E-01 0.0         0.0         0.0         0.0
   3                   0.0       0.0       1.866E-01 0.0         5.000E-10 2.000E-05
```

Table 3: Key to parameters in decay library

LIB	NUCLID	IU	THALF	FBX	FPEC	FPECX	FA	FIT	FSF
LIB	-	-	FN	QREC	ABUND	ARCG	WRCG	-	-

The two important parameters are IU, the time unit designation of the half-life, and THALF, the half-life of nuclide in units given by IU. Also, a program was writen to modify σ_γ, yield, or half-life based on lines from a text file. This code is shown below.

Listing 4: Script for modifying ORIGEN2 input.

```
#!/usr/bin/env python3

#Please note, might have to run command
# sed -i 's/E /E\+/g' TAPE9_BANK.inp
# on file to make sure there are no spaces after E's

######################################################################
######################### Import Packages ############################
######################################################################

import time
start_time = time.time()
import numpy as np
import Functions as Fun

######################################################################
######################### Class For Change ###########################
######################################################################

class ChangeClass:
    def __init__(self):
        self.XSec      = False #X Section
        self.Y         = False #Yield
```

```
        self.HL         = False #Half Life
        self.LIB        = ""
        self.ID         = ""
        self.Mod        = 0

#####################################################################
######################### Calculations ##############################
#####################################################################

#Open Input File
with open('../Origen2/TAPE9_BANK.inp') as f:
    content=f.readlines()

#Open output file
output=open("../Origen2/TAPE9.inp","w")

#Grab all the changes you want to make from
#the 'ChangesToMake.txt' file C will be a list with
#Each item being a change to make (all items are the same class)
C=[]
with open('ChangesToMake.txt') as f:
    Lines=f.readlines()
for i in Lines:
    i=i.split()
    C.append(ChangeClass())
    C[-1].XSec = eval(i[0])
    C[-1].Y    = eval(i[1])
    C[-1].HL   = eval(i[2])
    C[-1].LIB  = i[3]
    C[-1].ID   = i[4]
    C[-1].Mod  = i[5]
    if C[-1].HL:
        C[-1].Mod2 = i[6] #Change time scale too for half life


#Loop through the TAPE9 file, make changes, and write to output
SecondLine=False
for i in content:
    hold=i.split()
    for M in C: #Look through all the Mods
        if M.LIB in hold[0] and "-" not in hold[0]:
            if M.ID in hold[1]:
                if M.XSec: #Replace the gamma x-section
                    i=i.replace(hold[2],M.Mod)
                if SecondLine: #If on second line,replaceU235yield
                    #Iftherewerea third line, thenthiswouldn't work
                    i=i.replace(hold[3],M.Mod)
                SecondLine=False
                if M.Y: #Look to see if there is a second line
                    if hold[8]>0:
                        SecondLine=True #if so, then change yield
                                        #next tim
                if M.HL: #Change Half-life
                    #Count occurances of integer to replace (before
                    #our occurance), so
                    #we only replace the one we want to
```

```
                    Count=hold[0].count(hold[2])+hold[1].count(hold[2])+1
                    i=Fun.nth_repl(i,hold[2],M.Mod2,Count)
                    i=i.replace(hold[3],M.Mod)
    i=i.replace("\n","")
    print(i,file=output)


#########################################################################
######################## Time to Execute ################################
#########################################################################

print("--- %s seconds ---" % (time.time() - start_time))
```

☐ Create a sampling space for all possible variations of calculations

I might have to reduce the number of elements to look at, maybe I'll start by just looking at ^{154}Eu.

$$^{153}Pm \xrightarrow{\beta^-} {}^{153}Sm \xrightarrow{\beta^-} {}^{153}Eu \xrightarrow{(n,\gamma)} {}^{154}Eu$$

Table 4: Summary of parameters to vary for ^{154}Eu

Isotope	Half-life	σ_γ	Yield
^{153}Pm	5.25 ± 0.02 min	SIGMA	Yield
^{153}Sm	46.284 ± 0.004 hr	SIGMA	Yield
^{153}Eu	Stable	SIGMA	Yield
^{154}Eu	8.60 ± 0.01 y	SIGMA	Yield

In order to determine the sigma and yield variables I am planning on flux averaging both via:

$$\sigma = \frac{\int \sigma(E)\phi(E)dE}{\int \phi(E)dE}$$

and

$$\gamma = \frac{\int \gamma(E)\phi(E)dE}{\int \phi(E)dE}$$

where:

$$\begin{aligned}
\phi(E) =& C_1 \cdot E \cdot exp\left(-\frac{E}{\theta_{th}}\right) && E < E_{max,th} \\
=& \frac{C_2}{E} && E_{max,th} < E < E_{max,epi} \\
=& C_3 \cdot \sqrt{E} \cdot exp\left(-\frac{E}{\theta_{fis}}\right) && E > E_{max,epi}
\end{aligned}$$

and:

$$C_1 = \frac{1}{E_{max,th}^2 e^{-E_{max,th}/\theta_{th}}}$$
$$C_2 = 1$$
$$C_3 = \frac{1}{E_{max,epi}^{3/2} e^{-E_{max,epi}/\theta_{fis}}} = 1.64872e-09$$

Where: $E_{max,th} = 0.108$ eV, $E_{max,epi} = 2.1E6$ eV, $\theta_{th} = 0.054$ eV, and $\theta_{fis} = 1.4E6$ eV.

I was going to calculate σ_γ with nominal cross section values, then with error subtracted, and then added. These values would constitue a mean with error. A similiar calculation would be done for the yields.

- ☐ Determine importance of various uncertain parameters by running the code a number of times randomly sampling the sample space (still not 100% sure how to do this - not even 50% sure how to do this)

III Quantities of Interest and Uncertain Parameters

Quantities of interest are shown in Table 1 above. Uncertain parameters are listed below:

- Fission yield
- Cross sections
- Half-lives

IV Prediction

The first major prediction for this project is that half-lives will not have a large impact on results because they are relatively well known. Secondly, ^{125}Sb is notorious for being difficult to calculate correctly, I would would predict that there would be large uncertainties due to uncertainties in the cross section data.

References

[1] Allen G Croff. User's manual for the origen2 computer code. Technical report, Oak Ridge National Lab., 1980.