Fuel depletion sequence - *Mathematics & Implementation*

Lorenzo Vergari

NE255

Berkeley
UNIVERSITY OF CALIFORNIA

Outline

- The depletion sequence
- The Bateman equation
- The MATREX algorithm
- ORIGEN: Scale Tool for Depletion
- Case Study

Burnup simulation in advanced reactors result on significantly expensive problems

Goal

Study the evolution of fuel composition under irradiation in reactor and its decay after discharge

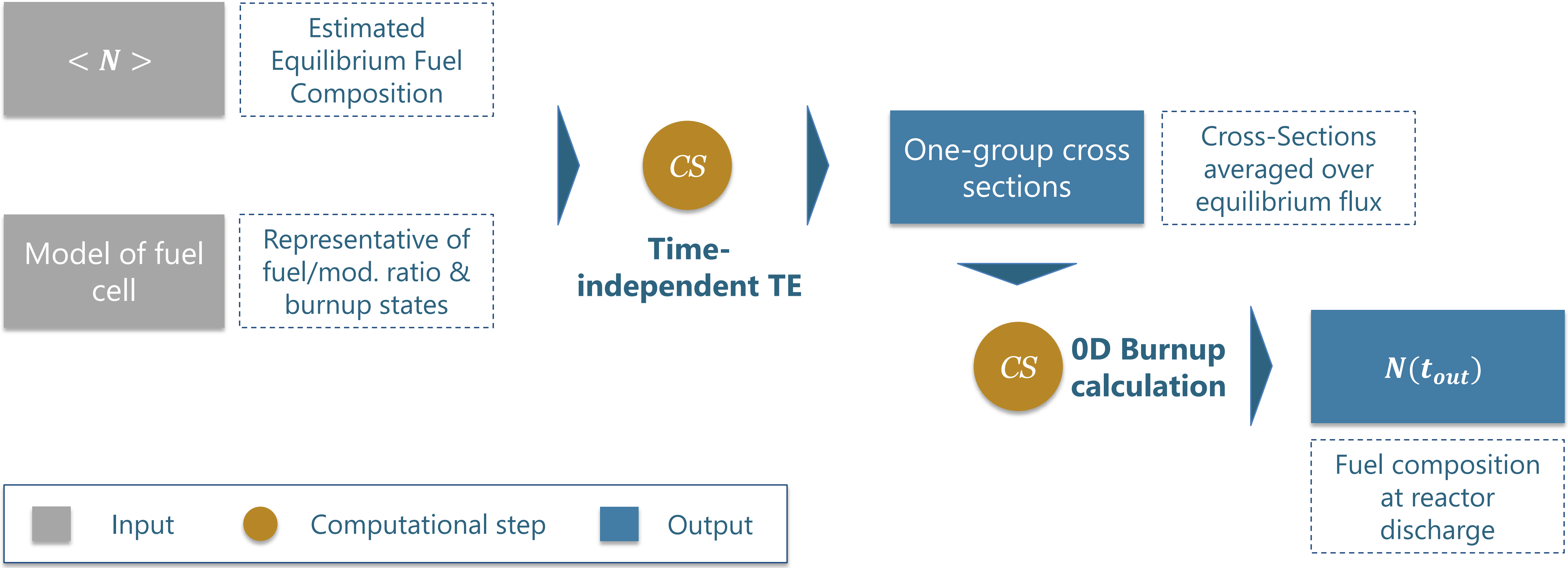
But

- Multi-Group cross sections are space and time dependent
- Flux seen by pebbles dependent on surrounding pebbles

Detailed modeling of reactor and time-dependent transport simulations, coupled with burnup calculation

Unless…

Computation of equilibrium one-group cross sections allows to decouple transport & burnup



The Bateman equation describes the space-averaged evolution of fuel composition

$$\frac{dN_i}{dt} = \sum_{j \neq i}^{N_{isotopes}} \underbrace{\overset{\text{Source from reaction}}{I_{ij}\lambda_j N_j(t)} + \overset{\text{Source from decay}}{f_{ij}\sigma_j \Phi N_j(t)} - \overset{\text{Loss term}}{(\lambda_i + \sigma_i \Phi) N_j(t)}}_{\boldsymbol{A_{ij}} N_j(t)} + S_i(t)$$

$I_{ij} = fractional\ yield\ of\ i\ from\ decay\ of\ j$
$f_{ij} = fractional\ yield\ of\ i\ from\ reaction\ on\ j$
$\lambda = decay\ constant; \sigma = total\ removal\ cross\ section$

Algorithms for Bateman Equation - MATREX

$$\frac{dN}{dt} = \boldsymbol{A}N(t) \rightarrow N(t) = e^{\boldsymbol{A}t}$$

*But **A** is a matrix!* ▶ ***Matr**ix **Ex**ponential**

$$N_i(t) = \underbrace{N_i(0)}_{\boldsymbol{C^0}} + \underbrace{t\sum_j A_{ij}N_j(0)}_{\boldsymbol{C^1}} + \underbrace{\frac{t}{2}\sum_k\left[tA_{ik}\sum_j A_{kj}N_j(0)\right]}_{\boldsymbol{C^2}} + \underbrace{\frac{t}{3}\sum_m\left[\frac{t}{2}A_{im}\sum_k\left[tA_{mk}\sum_j A_{kj}N_j(0)\right]\right]}_{\boldsymbol{C^{n+1}}}$$

$$\boldsymbol{C^0 = N(0)}; \qquad \boldsymbol{C^{\,n+1}} = \frac{\boldsymbol{t}}{\boldsymbol{n+1}}\sum_j A_{ij}\, C_j^{\,n}$$

▶

$$\boxed{\boldsymbol{N_j(t)} = \sum_n C_j^{\,n} + \epsilon_{trunc}}$$

** CRAM is an alternative algorithm*

ORIGEN: SCALE module for depletion studies

Bounds for neutron and photon spectra

CRAM or MATREX

Print, save and run options

Case details for xs, compositions, time and power

```
'SCALE comment
=origen

     % ORIGEN comment %

bounds{ … }
solver{ … }
options{ … }

case(A){
     time=[31 365] % days
     …
}

case(B){
     …
}

% more cases?

end
```

Inputs

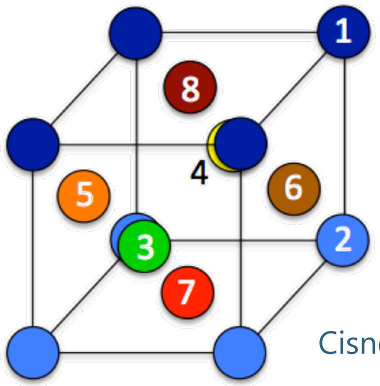
- Composition (entries or *.f71* file)
- Irradiation detail (time and power)
- Cross sections (*.f33* file)

Outputs

- Composition (*.f71* file)
- Neutron and photon spectra (*.f71* file)
- OPUS plots

Case study and results

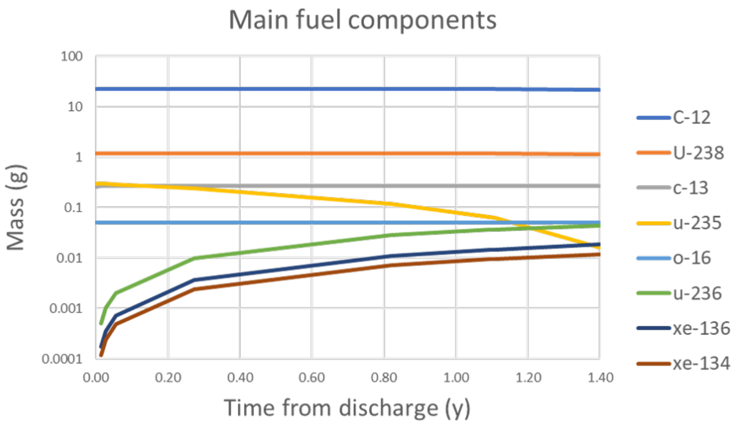
Step 1 – XS Generation



Cisneros, 2013

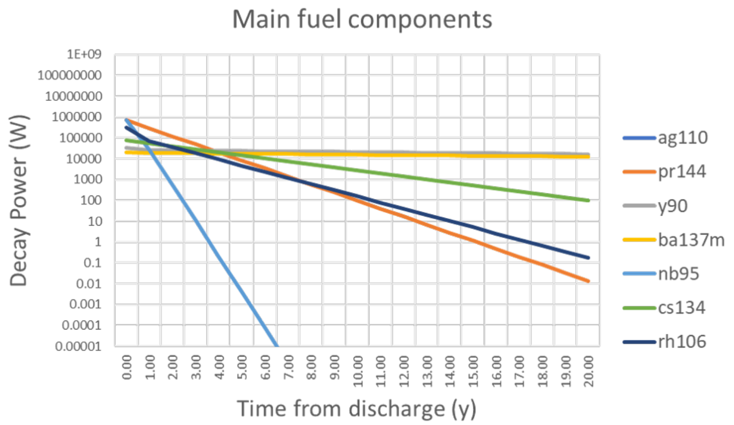
- 8 Burnup states
- Equilibrium cell & reflecting BC

Step 2 – Irradiation



- Production of FP & Activation
- Consumption of 95% U-235
- Simulation time= 51 s

Step 3 – Decay



- P'''@discharge=4.5 W/cm^3
- P'''@1year=6.3 mW/cm^3
- Simulation time= 32 s

Step 4 – Criticality Safety

- Spent pebbles stacked in FLiBe pool
- k_{eff} = *0.35* by stacking all reactor pebbles
- Critical mass is *never* reached (k_{eff} = *0.51* with 147M pebbles)

Summary

- Coupled, time-dependent transport-depletion sequence is computationally expensive. Steady state calculation of equilibrium XS and 0D depletion has a lower cost

- Fuel depletion described by the Bateman Equation. ORIGEN solves the Bateman equation with two algorithms (CRAM & MATREX)

- A fuel depletion (irradiation + decay) sequence was simulated with the described algorithms. Similar outcomes obtained with CRAM and MATREX. CRAM required twice as much time as MATREX in the decay step.

Q&A