Nuclear 0D modelling MECA2600

Power, neutron flow and atoms quantities

Context

- A nuclear reactor is a place where hundreds of different atoms can react.
  There are millions of different reaction per seconds.
  The kinetic reactor behavior can be hard to predict.
- As part of MECA2600 classes, students have to analyze a reactor, simplify it and model the reactor kinetic behavior. This work is based on kinetic analysis for a 0D model of reactor.

Goals

- Create a Matlab® model forecasting concentrations in a reactor based on initial fuel. OUTPUT required :
  - Quantity of atoms and neutrons in the reactor at each iteration
  - Power produced by the reactor at each iteration
  - Write a report (information will be provided):
    - How the problem has been solved (equations)
    - Physical analysis of parameters.
- Understand reactor kinetics (oral exam) :
  - Prompt neutron/ delayed neutron
  - fast neutron/thermal neutron
  - Poisons (specific fission products)
  - Control rods
  - Oral exam during final exam (~15 minutes)

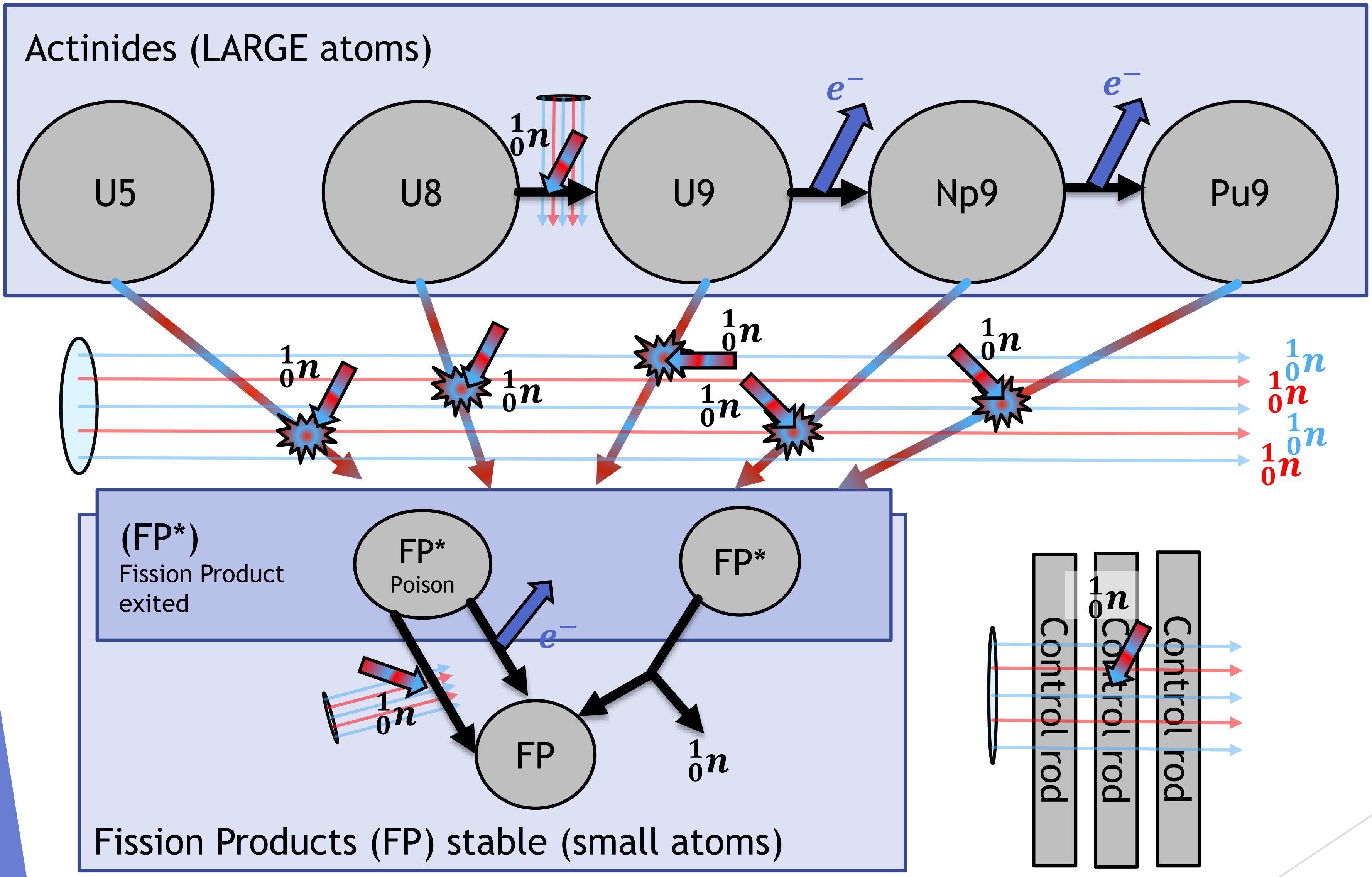
Input

- Geometry and neutronic data:
  - Vessel characteristics (volume, geometry has no impact)
  - neutronic data (cross sections, fission energy, delayed neutron fraction, half-life, fission yields…)
- Boundary conditions :
  - m_U5 : mass of U235 before starting reaction (oxide neglected)
  - m_U8 : mass of U238 before starting reaction (oxide neglected)
  - n_thermal initial : initial number of thermal neutrons
  - t_final : end time of modelisation
- Target :
  - Power production target

Process

- Hypothesis
  - Reaction
  - Kinetics
  - Power
- Teaching team is helping :
  - Matlab® structure code
  - Proposition of intermediate steps
  - Example of results
  - Time line

Process – Reaction

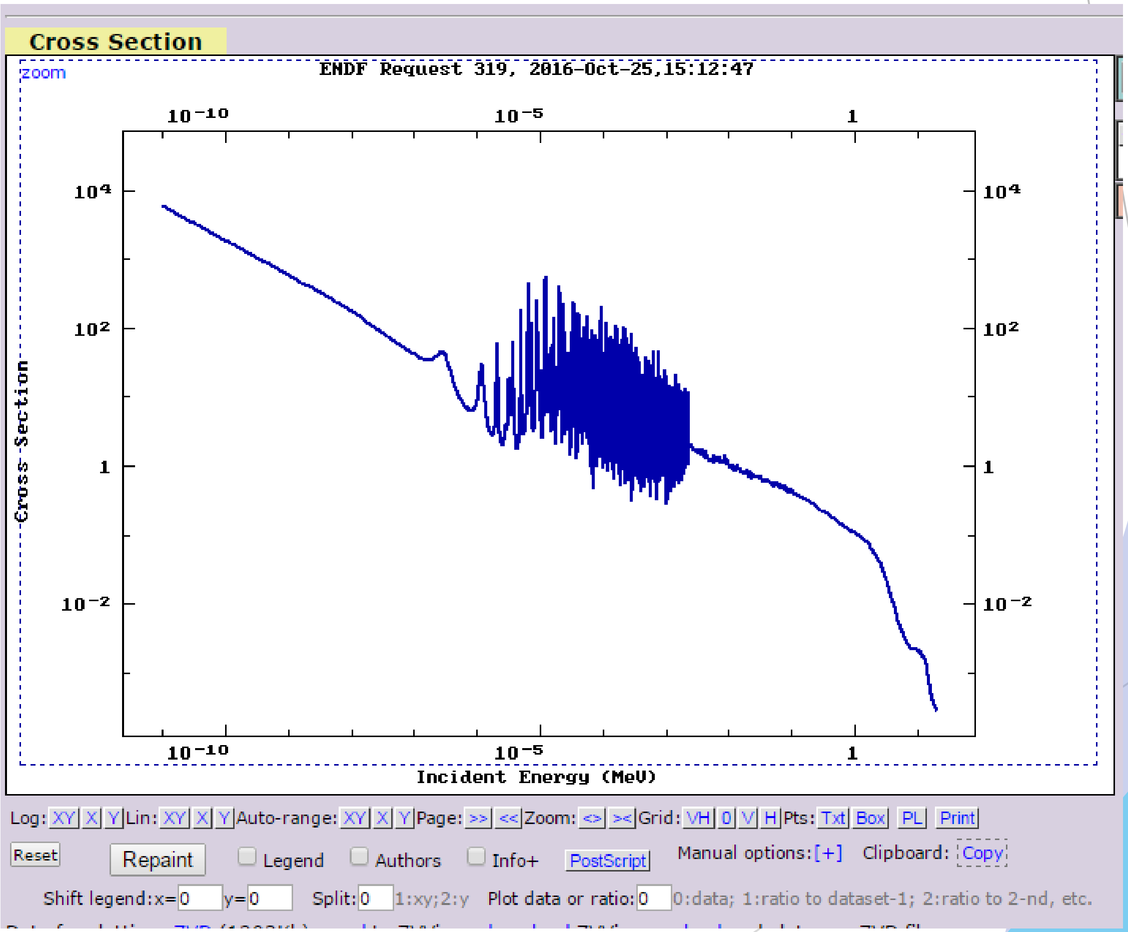


Legend	
	Neutron flux
^1_0n ^1_0n	Fast thermal neutron
e^-	Beta - ray
^1_0n	Fast or thermal capture
^1_0n	Fast or thermal fission

	Assumption
1	Only these atoms
2	Only these reactions
3	No retarded neutrons from LARGE atoms
4	Infinite reactor (no leaks, or leaks included in Control rods)

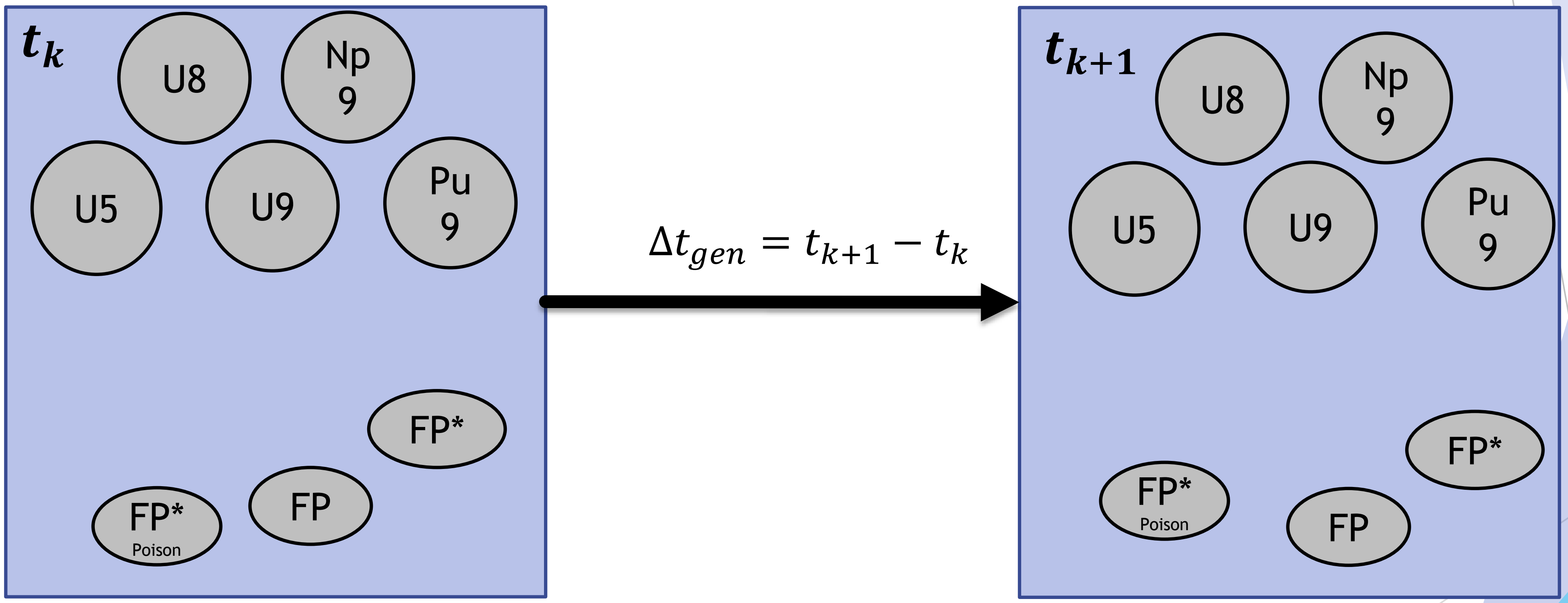
Process – Kinetics

- Reaction (given before)
  - Cross section (example neutron fission of U235) : https://www-nds.iaea.org/exfor/endf.htm
    - 2 kind of cross section used here :
      - neutron fission
      - neutron capture



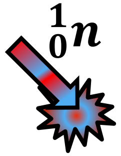
Processus – Cinetic

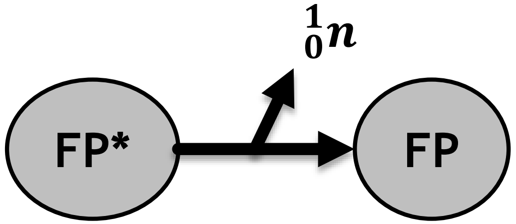
- Reaction (given before)
  - Time generation assumption : $\Delta t_{gen} = 10^{-4}[s]$



Process – Power

- Assumption : 3 sources of energy :
  - power released by fission : 200MeV
  - power released by stabilization (FP*->FP +1n +E). E=5MeV
  - power released by neutron fast to thermal





Teaching staff is helping :

- Matlab structure code :

Main function

Reactor_model

```matlab
function m=Reactor_model(t_final,dt_plot,P_stable,PF_retarded)
% REACTOR_MODEL modelise a nuclear reactor behavior in varying time.
%      [m] Structure containing all the atoms used in the model. This
%      structure contain the time varying number of atoms for each species.
%      Moreover, it has power, total energy, fast neutrons, thermal
%      neutrons ... See function GenerateStrucutre in this code.
%
%      [t_finale] model start at t=0[s] and ends at t_finale
%
%      [dt_plot] time descretisation where data have to be saved. This time will
%      indicate the size of m
%
%      [P_stable,PF_retarded] portion of atoms PF* -> PF + PF_retarded n
%      where, PF_retarded represent the avaerage number of neutrons
%      released during this reaction. P_stable represent the number of
%      atoms decreasing per second. This value is representative of
%      retarded neutrons.
%
%      The code is made in 3 parts :
%      1°) Variable definition
%      2°) Time iteration with non linear matrix to solve
%      3°) Graphic generation
%
%      2 auxillary function are inside the code :
%           barDeControle(t,Power) : Give the proportion of neutrons
%           absorbed by neutron bar per second.
%           GenerateStructure(SIZE) : generate a structure of size SIZE.
```

Auxiliary functions

Demi_vie :
beta decay

```matlab
function demi_vie = Demi_vie(X,Transfo)
%FROM : http://wwwndc.jaea.go.jp/NuC/
```

molar Mass :
molar masses

```matlab
function M=molarMass(X)
% FROM : http://wwwndc.jaea.go.jp/NuC/
```

Section efficace :
cross section based on ENDF

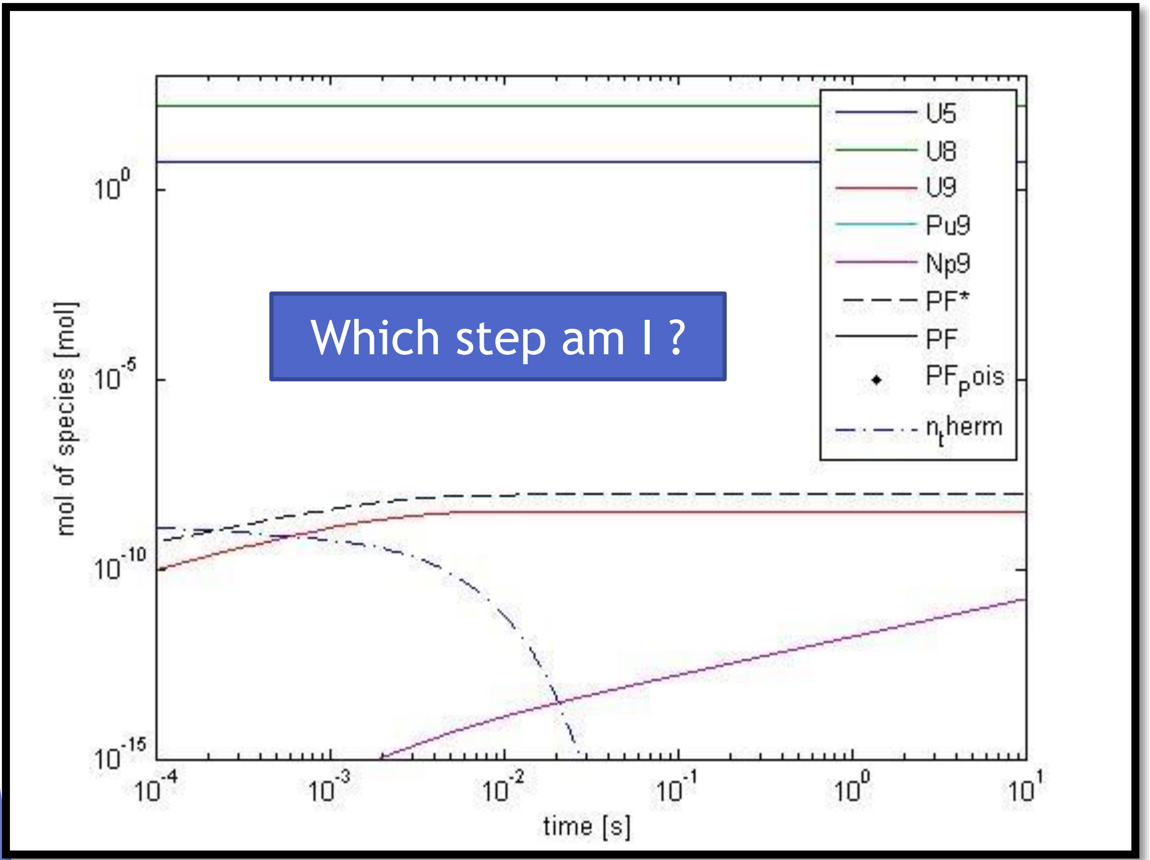
```matlab
function [sigma]=Section_efficace(X,Transfo,n_eV,User_Adress)
% Section_efficace [barn]  Section efficace d'une transformation pour 1 composant
%    [SIGMA] = Section_efficace(X,TRANSFO,N_EV) donne la section
%    efficace SIGMA de la transformation TRANSFO de l'element chimique X
%    lorsque le neutron incident a une énergie de N_EV
%
%    X : Espèce chimique.
%        ATTENTION : ici on travail avec un nombre limité d'espèces
%        chimiques
%
%    TRANSFO : seules les transformations ci dessous sont utilisées :
%        Fission : Probabilité qu'un noyau absorbe un neutron et fissionne
%        Capture : Probabilité qu'un noyau absorbe un neutron et fissionne
%
%    n_eV : energie du neutron incident. Peut etre un vecteur
%        ATTENTION : On suppose une energie comprise entre 1e-5 et 2e7 [eV]
%
%    User_Adress : Adress of the data base
%
%    ETAPES : etapes intermédiaires pour construire la data base :
%        1°) Construction d'une data base pour les elements Ux et Np9 et Pu9
%        où Ux peuvent soit fissioner soit capturer jusqu'à U9. Puis U9 à
%        Np9 peuvent soit fissioner soit beta -. Pu peut juste fissioner
%
%    SOURCES : fichier viennent de
%    https://www-nds.iaea.org/exfor/endf.htm
%    C'est la base ENDF qui a généralement été utilisé
```

Teaching staff is helping

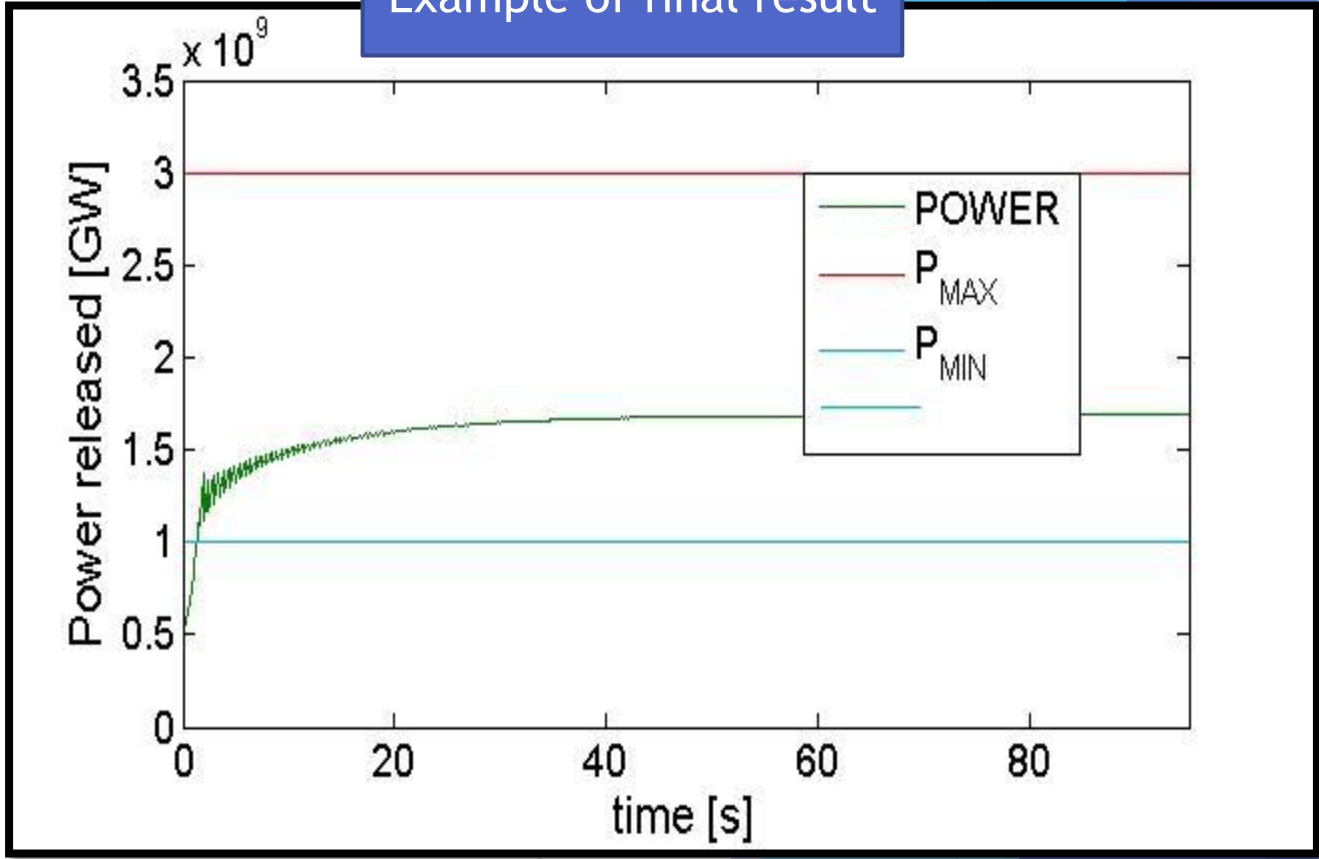
- Proposition of intermediate steps implementation
    1. Create auxiliary functions (Molar mass, half life, cross sections)
    2. Thermal reactions with constant neutron flow. No poison.
    3. Add variable prompt neutron flow
    4. Add variable delayed neutron flow
    5. Add fast neutron which can slow down
    6. Add control rod (at fixed value)
    7. Add fast reactions
    8. Add Poisons
    9. Add control rod (variable value <-> Power)

Teaching staff is helping

- Example of results :
  - Graphs are not linked!

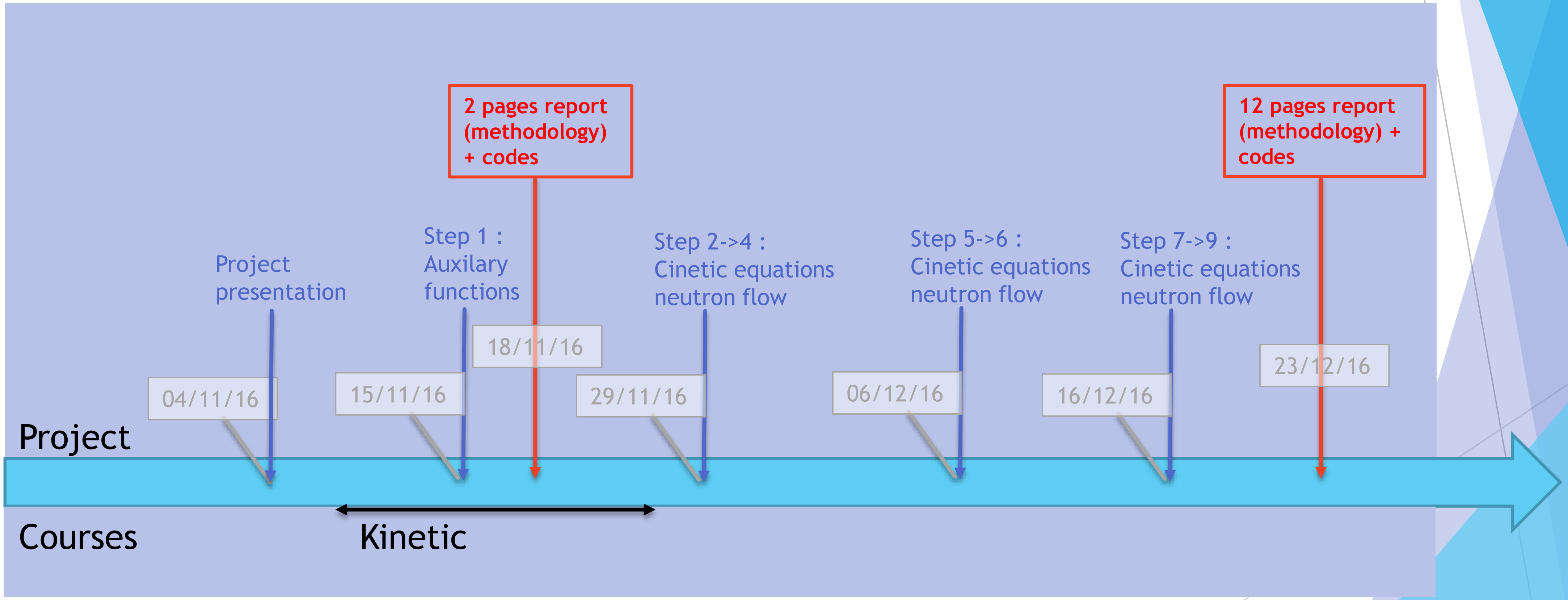


Example of final result



Teaching staff is helping

- Time line:
  - For any question, ask an appointment (Gauthier.limpens@uclouvain.be desk : b.055 STEVIN)



Teaching staff is helping

- Evaluation (to be confirmed):
  - Code : 20%
  - Report : 50%
  - Oral : 30%
- Report informations (10 pages) :
  - 2 pages (18/11/2016) : Methodology how did you get cross sections, half-lifes and molar masses.
  - 10 pages :
    - 5 pages : methodology
    - 4 pages : analyse of the code (behavior, exception…)
    - 1 page : additional work done (free topic)