Neutron flux simulation of a one-dimensional reactor zone taking into account xenon generation

Zsófia Tóth,

Dániel Péter Kis

Institute of Nuclear Techniques,
Budapest University
of Technology and Economics,
Műegyetem rakpart 3,
1111 Budapest, Hungary
Email: zsofiatoth412@gmail.com

The energy in nuclear power plants is produced by thermal fission. It is extremely important to be able to monitor the processes in the reactor to ensure the safety and reliability of the power plant. One of the main traits of the reactor core is neutron flux. It changes in time and space therefore it is crucial to be able to simulate its changes with computer codes. In the research work a program code was established in the Matlab software with which the neutron flux of a one-dimensional zone can be simulated with homogenous and heterogenic zone parameters as well. The code is written using the one-group one-dimensional time- and space-dependent diffusion equation. The equation of an average delayed neutron group and xenon and iodine distributions was also included in the system to give a more precise look on the problem. The main innovation in the code is that numerical methods were used to solve the problem: the finite difference approach was applied for the place-dependent and for the time-dependent solution. The advantage of this code compared to other ones is that one-dimensional zones can be simulated in a really short time and it still gives a precise solution because of the complex numerical methods used.

Keywords: nuclear power plant, neutron flux, simulation, finite differences

INTRODUCTION AND MOTIVATION

Nowadays, nuclear power plants are being used and planned to build in many countries around the world to meet the growing energy needs. Most of the currently planned and under construction nuclear power plants have an installed capacity exceeding 1000 MW. One of the most important parameters of these types of nuclear reactors is the xenon flux. The essence of the process is that the xenon isotope being non-uniformly distributed in space during operation of the reactor can result in strong instabilities during the power control. Neglecting this process endangers the maintenance of a critical state of the reactor during its normal operation [1]. A good example of this is the Paks II project. Two new VVER-1200 units [2] are planned in the framework of the project where the increased reactor size requires high performance.

The axially large zone of the reactor cores is typically sensitive to changes in high non-uniformly generated xenon concentrations. On this basis, monitoring and determination of changes in xenon concentration, and compensation of its effect on performance are essential to guarantee nuclear safety.

The aim of this research is to simulate the neutron flux distribution and the spatial and temporal changes of xenon concentration during reactor operation. The main goal is to get to know how xenon oscillation affects the reactor power and neutron flux distribution. To investigate this phenomenon, we firstly developed a code for a one-dimensional, time-dependent model to solve the diffusion equation, and then a program was written to solve the non-linear equation system consisting of equations which describe the concentration variations of the xenon. With this one can numerically solve the differential equation system describing the phenomenon and examine the properties of xenon oscillation and the change of neutron flux and xenon concentration.

SOLUTION OF THE STATIC DIFFUSION EQUATION

During the research the diffusion equation was applied to describe the problem. In this chapter, the solution of the static diffusion equation with the finite difference method is presented.
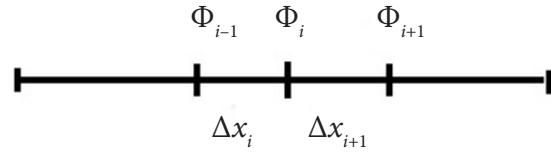
The finite differences method

In the first step, the method of finite differences is applied to the one-group diffusion equation, the form of which in a homogeneous reactor is as follows:

$$0 = D\Delta\phi(x) + \left[\frac{v\Sigma_f}{k_{eff}} - \Sigma_a\right]\phi(x). \qquad (1)$$

Here D is the diffusion constant, ϕ is the space-dependent group flux, v is the average neutron yield, Σ_f is the macroscopic fission cross-section, Σ_a is the macroscopic absorption cross-section, and k_{eff} is the effective multiplication factor. In the one-group approximation, the energy of the neutrons is considered equal.

When applying the finite difference method, the examined domain is divided into arbitrary small parts according to the variable x within which the material parameters are considered to be homogeneous and the flux between the dividing points is also considered to be constant. The point of the x coordinate is indicated with index 1, the last with index n and in between with the index of the endpoint of the domain. An example of this division is shown in Fig. 1 in the range $(i - 1)$ and $(i + 1)$. The boundary conditions of the problem are the vacuum conditions that are implemented at the $i = 0$ and $i = n + 1$ points.

Fig. 1. An example of division in the $(i - 1) - (i + 1)$ range by the finite difference method

Integrating the first equation in the $[x_{i-1/2}, x_{i+1/2}]$ range with the finite differences the equations can be described as follows:

$$\int_{x_{i-1/2}}^{x_{i+1/2}} \Sigma_a \phi(x) dx$$

$$\cong \left(\frac{\Delta x_{i+1}}{2}\Sigma_{a_{i+1}} + \frac{\Delta x_i}{2}\Sigma_{a_i}\right)\Phi_i, \qquad (2)$$

$$\int_{x_{i-1/2}}^{x_{i+1/2}} \Sigma_a \phi(x) dx$$

$$\cong \left(\frac{\Delta x_{i+1}}{2}\Sigma_{a_{i+1}} + \frac{\Delta x_i}{2}\Sigma_{a_i}\right)\Phi_i, \qquad (3)$$

$$\int_{x_{i-1/2}}^{x_{i+1/2}} \Sigma_a \phi(x) dx$$

$$\cong \left(\frac{\Delta x_{i+1}}{2}\Sigma_{a_{i+1}} + \frac{\Delta x_i}{2}\Sigma_{a_i}\right)\Phi_i. \qquad (4)$$

The average neutron yield is considered to be constant in all ranges because its change can be neglected. Thus, the stationary diffusion equation is a finite differential approximation to the i-th dividing point:

$$0 = \frac{D_{i+1}}{\Delta x_{i+1}}\phi_{i+1} + \frac{D_i}{\Delta x_i}\phi_{i-1} - \left(\frac{D_{i+1}}{\Delta x_{i+1}} + \frac{D_i}{\Delta x_i}\right)\phi_i$$

$$-\left(\frac{\Delta x_{i+1}}{2}\Sigma_{a,i+1} + \frac{\Delta x_i}{2}\Sigma_{a,i}\right)\phi_i$$

$$+\frac{1}{k_{eff}}\left(\frac{\Delta x_{i+1}}{2}\Sigma_{f,i+1} + \frac{\Delta x_i}{2}\Sigma_{f,i}\right)v\phi_i. \tag{5}$$

In order to simplify the solution, it is worth writing the equations in a vector or a matrix form, therefore, Equation 5 is simplified to the following form:

$$0 = a_i\phi_{i+1} + b_i\phi_{i+1} - e_i\phi_i + f_i. \tag{6}$$

Here

$$a = \begin{bmatrix} 0 & \frac{D_2}{\Delta x_2} & 0 & 0 & 0 & 0 \\ 0 & 0 & \ddots & 0 & 0 & 0 \\ 0 & 0 & \ddots & \ddots & 0 & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & \ddots & \ddots & \frac{D_n}{\Delta x} \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$b = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{D_2}{\Delta x_2} & 0 & 0 & 0 & 0 & 0 \\ 0 & \ddots & \ddots & \ddots & 0 & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & \ddots & \ddots & 0 \\ 0 & 0 & 0 & 0 & \frac{D_n}{\Delta x_n} & 0 \end{bmatrix},$$

$$c = \begin{bmatrix} \frac{\Delta x_1}{2}\Sigma_{a,1} + \frac{\Delta x_2}{2}\Sigma_{a,2} & 0 & 0 & 0 & 0 & 0 \\ 0 & \ddots & 0 & 0 & 0 & 0 \\ 0 & \ddots & \ddots & \ddots & 0 & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & \ddots & \ddots & 0 \\ 0 & 0 & 0 & 0 & \frac{\Delta x_n}{2}\Sigma_{a,n} + \frac{\Delta x_{n+1i}}{2}\Sigma_{a,n+1} \end{bmatrix}, \tag{7}$$

$$e = a + b + c, \tag{8}$$

and

$$f = P\phi = \frac{1}{k_{eff}} \tag{9}$$

$$\begin{bmatrix} \frac{\Delta x_1}{2}v\Sigma_{f,1} + \frac{\Delta x_2}{2}v\Sigma_{f,2} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \frac{\Delta x_n}{2}v\Sigma_{f,n} \\ +\frac{\Delta x_{n+1}}{2}v\Sigma_{f,n+1} & & \end{bmatrix}\begin{bmatrix}\phi_1 \\ \vdots \\ \phi_n\end{bmatrix}.$$

Arranging Equation 6 into a matrix, the flux vector can be calculated as follows:

$$\phi = -A^{-1}f. \tag{10}$$

Here matrix A is

$$A = \begin{bmatrix} -e_1 & a_1 & 0 & 0 & 0 & 0 \\ b_2 & -e_2 & a_2 & 0 & 0 & 0 \\ 0 & b_3 & -e_3 & a_3 & 0 & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & \ddots & \ddots & a_{n-1} \\ 0 & 0 & 0 & 0 & b_n & -e_n \end{bmatrix}. \tag{11}$$

To solve this equation, iterations shall be used. Two types of iterations were utilized during the calculations which are presented below.

SOURCE ITERATION WITH DIRECT INVERSE CALCULATION

In this method, firstly the neutron flux for the source is calculated which is then normalized

by a special norm (which equals the effective multiplication factor during iteration) to ensure the convergence of iteration [4]. Thus, the effective multiplication factor is calculated during the whole process. Iteration ends when the relative difference between the effective multiplication factors in two consecutive iteration steps falls below a specified error. The flowchart of iteration is shown in Fig. 2.
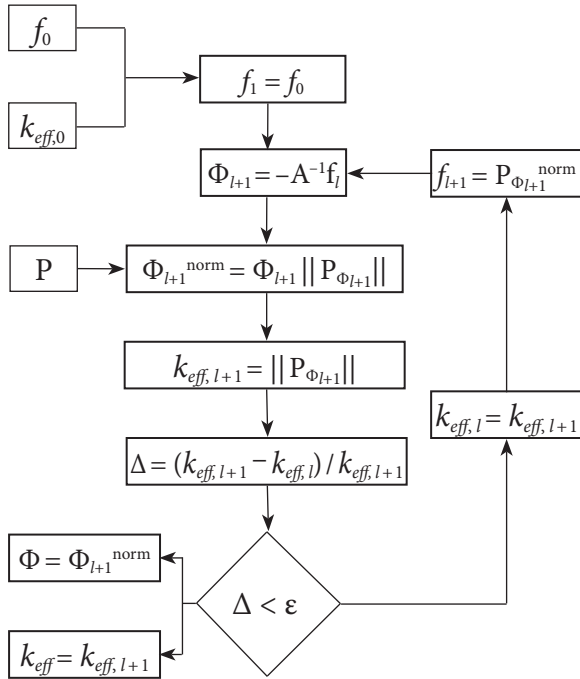

Fig. 2. The flowchart of source iteration

The method can easily solve the equation; however, if larger matrices are used, the value of the numerical error increases due to the inaccuracy of the matrix inversion. Although Matlab solves the matrix version by iterating, we tried to solve the equation with a new iteration procedure instead of a matrix version to increase the accuracy of the calculations. This method is called internal iteration and is described in the following section.

SOURCE ITERATION WITH INTERNAL ITERATION

To apply the internal iteration matrix B is defined [4]:

$$B = \begin{bmatrix} 0 & \dfrac{a_1}{e_1} & 0 & 0 & 0 & 0 \\ \dfrac{b_2}{e_2} & 0 & \dfrac{a_2}{e_2} & 0 & 0 & 0 \\ 0 & \dfrac{b_3}{e_3} & 0 & \dfrac{a_3}{e_3} & 0 & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & \ddots & \ddots & \dfrac{a_{n-1}}{e_{n-1}} \\ 0 & 0 & 0 & 0 & \dfrac{b_n}{e_n} & 0 \end{bmatrix}. \quad (12)$$

Using Equations 10 and 12 and finite differences, the following equation can be obtained:

$$\Phi^{k+1} = B\phi^k + f'. \quad (13)$$

Here

$$f' = P'\phi = \dfrac{1}{k_{eff}} \quad (14)$$

$$\begin{bmatrix} \dfrac{\dfrac{\Delta x_1}{2} v\Sigma_{f,1} + \dfrac{\Delta x_2}{2} v\Sigma_{f,2}}{e_1} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \dfrac{\dfrac{\Delta x_n}{2} v\Sigma_{f,n} + \dfrac{\Delta x_{n+1}}{2} v\Sigma_{f,n+1}}{e_n} \end{bmatrix} \begin{bmatrix} \phi_1 \\ \vdots \\ \phi_n \end{bmatrix}.$$

The flowchart of internal iteration is shown in Fig. 3.
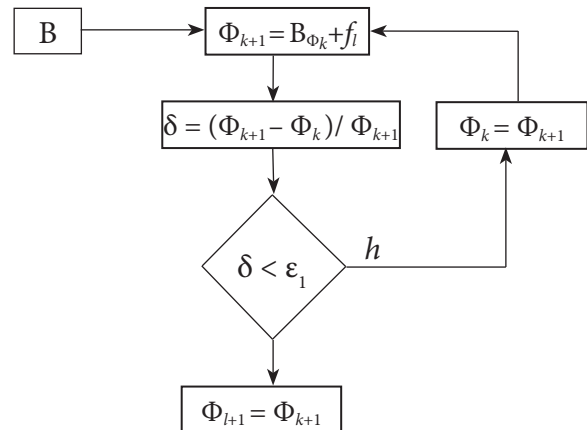

Fig. 3. The flowchart of internal iteration

NUMERICAL TESTING

The accuracy of the code was tested with calculations on a critical one-dimensional reactor based on the example of Zoltán Szatmári's Reactor Physics Calculations [3]. During these calculations the values of the effective multiplication factor are obtained by two methods of the solution for a different number of the dividing points given in Table 1.

Table 1. **Values and relative deviations of the effective multiplication factor using two methods**

Number of dividing points	Source iteration	Internal iteration	Relative difference, pcm
10	1.0308	1.0308	9.11E-03
20	1.0161	1.0160	2.77E-02
30	1.0108	1.0108	5.51E-02
50	1.0066	1.0066	1.22E-01
75	1.0044	1.0044	2.26E-01
100	1.0034	1.0033	3.50E-01

The test was done up to 100 dividing points because the effective multiplication factor above this number changes only marginally within one division. In both cases, the deviation of the effective multiplier for 100 dividing points differs less than by 0.34% from the analytically calculated, therefore, the program can be considered to be accurate enough for further calculations.

SOLUTION OF THE TIME-DEPENDANT DIFFUSION EQUATION

In the next step, a program was established for solving the time-dependant diffusion equation.

Presentation of the time-dependent solution [1]

In this case the examined equation is

$$\frac{1}{v}\frac{\partial \phi(x,t)}{\partial t} = D\Delta\phi(x,t) + [v\Sigma_f - \Sigma_a]\phi(x,t), \quad (15)$$

and the method of solving is also the finite differences method as in the stationary case. The equation with this can be described as follows:

$$\frac{1}{v}\frac{\partial}{\partial t}\int_{x_{i-1/2}}^{x_{i+1/2}}\phi(x,t)dx = \frac{1}{v}\frac{\partial}{\partial t}\left(\frac{\Delta x_{i+1} + \Delta x_i}{2}\phi_i\right)$$

$$= \frac{1}{v}\frac{\partial \phi_i(t)}{\partial t}\frac{\Delta x_{i+1} + \Delta x_i}{2}). \quad (16)$$

Applying Equation (5) to the right side, Equation (13) with finite differences is as follows:

$$\frac{\partial \phi_i}{\partial t} = \frac{2v}{\Delta x_{i+1} + \Delta x_i}\frac{D_{i+1}}{\Delta x_{i+1}}\phi_{i+1}$$

$$+ \frac{2v}{\Delta x_{i+1} + \Delta x_1}\frac{D_i}{\Delta x_i}\phi_{i-1} - \frac{2v}{\Delta x_{i+1} + \Delta x_1}\left(\frac{D_{i+1}}{\Delta x_{i+1}} + \frac{D_i}{\Delta x_i}\right)\phi_i$$

$$- \frac{2v}{\Delta x_{i+1} + \Delta x_1}\left(\frac{\Delta x_{i+1}}{2}\Sigma_{a,i+1} + \frac{\Delta x_i}{2}\Sigma_{a,i}\right)\phi_i$$

$$\frac{2v}{\Delta x_{i+1} + \Delta x_1}\left(\frac{\Delta x_{i+1}}{2}v\Sigma_{f,i+1} + \frac{\Delta x_i}{2}v\Sigma_{f,i}\right)v\phi_i. \quad (17)$$

By space dependence the components of the flux can be sorted into a vector therefore the term is simplified to the matrix equation

$$\frac{d\mathbf{\Phi}(t)}{dt} = \mathbf{K}\mathbf{\Phi}(t), \quad (18)$$

where

$$\mathrm{K} = \frac{2v}{\Delta x_{i+1} + \Delta x_i} \quad (19)$$

$$\begin{bmatrix} -e_1 & a_1 & 0 & 0 & 0 & 0 \\ b_2 & -e_2 & a_2 & 0 & 0 & 0 \\ 0 & b_3 & -e_3 & a_3 & 0 & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & \ddots & \ddots & a_{n-1} \\ 0 & 0 & 0 & 0 & b_n & -e_n \end{bmatrix} + \frac{2v}{\Delta x_{i+1} + \Delta x_i}$$

$$\begin{bmatrix} \frac{\Delta x_1}{2}v\Sigma_{f,1} + \frac{\Delta x_2}{2}v\Sigma_{f,2} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \frac{\Delta x_n}{2}v\Sigma_{f,n} + \frac{\Delta x_{n+1}}{2}v\Sigma_{f,n+1} \end{bmatrix}.$$

After that time discretization can be performed and the entire time range can be divided into small intervals. The dividing points are denoted by index p. The solution using an implicit scheme is as follows:

$$\frac{\mathbf{\Phi}^{p+1} - \mathbf{\Phi}^p}{\Delta t} = \mathbf{K}\mathbf{\Phi}^{p+1}, \quad (20)$$

$$\Phi^{p+1} = (1-\mathbf{K}\Delta t)^{-1}\Phi^p. \tag{21}$$

EXAMINATION OF THE SYSTEM WITH DELAYED NEUTRONS

The systems described in the previous chapters describe only the prompt neutrons; however, late neutrons play a particularly important role in reactor operation [1]. These neutrons do not occur at the moment of fission but they are generated in the negative beta decay of the nuclei. This process makes it possible to control the reactor.

Presentation of the equations applied for delayed neutrons
A new source appears in the diffusion equation when introducing the delayed neutrons to the system [3]

$$\frac{1}{v}\frac{\partial\phi(x,t)}{\partial t} = D\Delta\phi(x,t) - \Sigma_a\phi(x,t)$$

$$+v(1-\beta)\Sigma_f + \sum_{i=1}^{6}\lambda_i C_i(x,t). \tag{22}$$

where β is the delayed neutron ratio, λ_i is the decay constant of the parent nucleus i, and $C_i(x, t)$ is the concentration of the parent nucleus i. The latter can be described by the following equation:

$$\frac{\partial C_i(x,t)}{\partial t} = -\lambda_i C_i(x,t) + \beta_i v\Sigma_f\phi(x,t). \tag{23}$$

To speed up the calculation an average delayed neutron group was used instead of 6 groups, the parameters of which were defined as follows:

$$\beta = \sum_{i=1}^{6}\beta_i, \tag{24}$$

$$\Phi^{p+1} = (1-\mathbf{K}\Delta t)^{-1}\Phi^p. \tag{25}$$

The effect of delayed neutrons to the flux
To study the effect of delayed neutrons on the time dependence of the flux, a slightly supercritical system was examined with neglecting and taking into account the delayed neutrons. The results are presented in Fig. 4.

It can be seen that with the delayed neutrons the change of the neutron flux is much slower. The reason for this is that the delayed neutrons are generated later in time (not in the moment of fission), therefore they decelerate the increase of the neutron flux. Without them the controlled chain reaction could not be achieved.

XENON

Presentation of the process of xenon poisoning and xenon oscillation
The xenon-135 isotope has a capture cross-section of $2.65 \cdot 10^6$ barn which is 4000 times more than the same value of the uranium-235 isotope [5]. The incidence of xenon-135 is also very high with 6.4% [5]. These two properties result
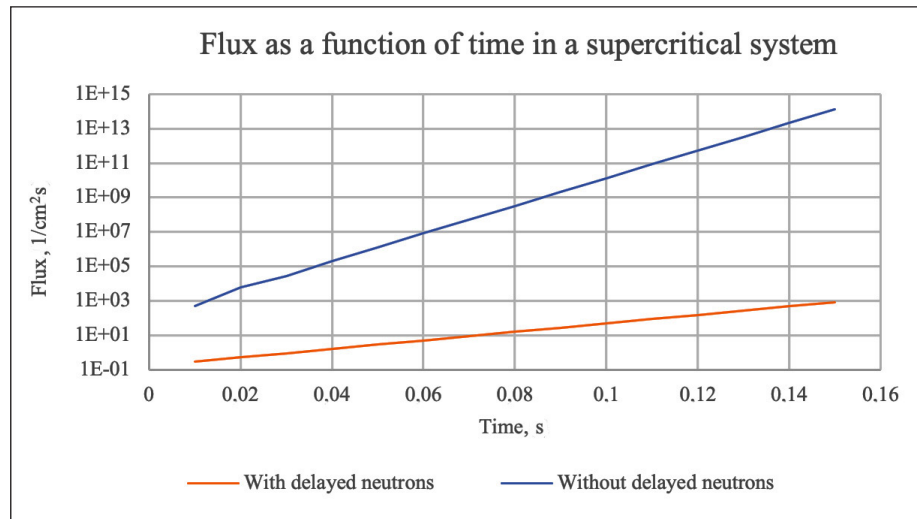

Fig. 4. Flux change with neglecting late neutrons and taking them into account

in a significant effect regarding the reduction of reactivity in the zone, therefore its effect should be taken into account throughout the lifetime of the reactor.

The xenon-135 isotope can also be formed directly during the cleavage or from tellurium-135 (cleavage product) via iodine-135 with beta decomposition. The half-life of tellurium is very short, therefore it can be assumed that iodine is directly formed during cleavage. The concentration of the xenon-135 isotope can decrease by negative beta decomposition and neutron capture.

Figure 5 shows a simplified scheme for the generation and decomposition of xenon-135 [1].

For reactors with a small built-in capacity the concentration of xenon can be considered independent of space with only small spatial inequalities in the flux. The same cannot be claimed for high-performance reactors where the spatial distribution of neutron flux can vary widely.
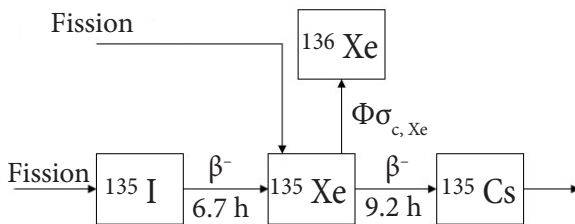
The concentration of xenon during flux changes is presented in Fig. 6 as a function of time:

The cause of xenon oscillation is the inhomogeneity of power distribution within the reactor. As an example, when a control rod is displaced, the flux increases in one part of the reactor and decreases in the other part because of the constancy of performance. Thus, the concentration of xenon differs in the two different parts of the reactor. As a result, reactivity will increase in one half of the reactor and decrease in the other. However, the process reverses xenon poisoning, while the direction of power change remains for a while. As the power change direction changes, the reactor continues to return to its original state, causing a period of oscillation to occur. The process period is about 30 h. After the end of the period, the swing starts again and the performance and flux will swing in time and space. Thus, testing and simulating xenon poisoning is particularly important for high-performance reactors.

Presentation of the applied model
To describe xenon poisoning equations describing the concentration of iodine and xenon isotopes were attached to the code that solves the diffusion equation [6]
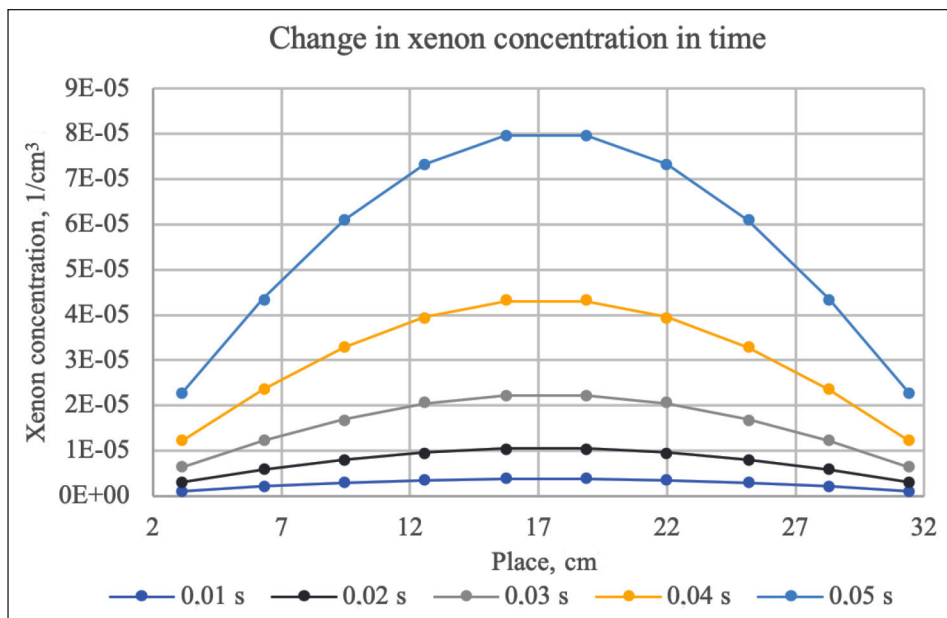
Fig. 5. Xenon-135 – A scheme of generation and decomposition

Fig. 6. Changes in xenon-135 concentration as a function of location at different time points

$$\frac{\partial N_I(x,t)}{\partial t} = \lambda_I N_I(x,t) + y_I \Sigma_f \phi(x,t), \qquad (26)$$

$$\frac{\partial N_{Xe}(x,t)}{\partial t} = \lambda_I N_I(x,t) + y_{Xe} \Sigma_f \phi(x,t)$$

$$-[\lambda_{Xe} + \sigma_{Xe}^a \phi(x,t)] N_{Xe}(x,t), \qquad (27)$$

where N denotes the core density, y is the generation frequency, λ is the decomposition constant while the Xe and I indices refer to the xenon-135 and iodine-135 nuclei.

These equations were also built into the code using the finite difference method. Due to the appearance of xenon, the absorption part changes in the original diffusion equation therefore it was modified as follows:

$$\frac{1}{v}\frac{\partial \phi(x,t)}{\partial t} = D\Delta\phi(x,t) + [v\Sigma_f$$

$$-\Sigma_a]\phi(x,t) - \sigma_{Xe}^a N_{Xe}(x,t)\phi(x,t). \qquad (28)$$

Here σ_{cXe} is the microscopic capture cross-section of the xenon-135 nucleus.

In the solution, the xenon concentration is continuously overwritten and so is the absorption part of the equation. The scattering part also changes slightly due to the appearance of xenon, but this vanishes in one group model.

The spatial and temporal discretization of the equations was performed as described in Equations 2 and 3. An iteration solver code was established to solve these numerical equations.

Investigation of a homogeneous symmetric reactor

With the code one-dimensional zones can be examined in a really short time as a run with it only takes some seconds. When looking at the xenon concentration, a cosine distribution was expected regarding the center of the zone. The reason for this is that the cosine function is an appropriate solution when solving the static xenon balance equation. Running the code resulted in the distribution shown in Fig. 6 which meets our expectations.

It can be clearly seen that the xenon concentration increases over time, which is also in line with our expectations.

From these first runs, the code will be further developed in the future, not only for xenon concentration but also for xenon flux in homogeneous and heterogenous reactors.

CONCLUSIONS

During this work a diffusion code was built that can calculate flux in the reactor in stationary and transient cases using the finite difference method. For this differential equations describing the changes in the concentration of xenon and iodine nuclei were attached to the code which allowed us to investigate the spatial and temporal distribution of flux and nuclei using the finite difference method.

Although this is only the beginning of a great research project, there are still some advantages regarding the code. One-dimensional zones can be simulated in a really short time with the code and it still gives a precise solution because of the complex numerical methods used. It is even coupled with the equations regarding the xenon-135 concentration, therefore information about the xenon concentration can be obtained as well.

There are many possibilities for future developments. The main goal is to increase the accuracy of numerical solutions using the Runge–Kutta method. In addition, a more detailed examination of the effects of xenon toxicity and xenon flux on reactor power needs to be done as well, investigating the spatial and temporal characteristics and the influencing parameters which will require further development of the code. We are still working on comparing our results with the literature and the work of other authors as it is still the beginning of a great research project.

As a further development option, code extension from one group to several groups arises, which would allow one to examine the problem more accurately. In addition, there is a great potential for testing heterogeneous systems with the code which would also open up another area of research. The code will also be verified with the existing published data to ensure its accuracy.

References

1. Csom G. *Atomerömüvek üzemtana* (in Hungarian). Budapest, 2005.

2. Internet source: https://www.paks2.hu/hu/PaksII/AJovo/az_uj_blokkok

3. Szathmáry Z. *Reaktorfizikai számítások* (in Hungarian). Budapest, 2012.

4. Csengeri E. *A xenonlengés elméleti vizsgálata* (in Hungarian). BSc Thesis. Budapest, 2016.

5. *Experimental Nuclear Reaction Data (EXFOR)*. NRDC. https://www-nds.iaea.org/exfor/exfor.htm

6. Szathmáry Z. *Reaktorfizika mérnököknek* (in Hungarian). Budapest, 2007.

Zsófia Tóth, Dániel Péter Kis

NEUTRONŲ SRAUTO MODELIAVIMAS VIENMATĖJE REAKTORIAUS ZONOJE ATSIŽVELGIANT Į KSENONO GENERAVIMĄ

Santrauka

Branduolinėse jėgainėse energija generuojama šiluminiams neutronams dalijant branduolius. Siekiant užtikrinti jėgainės saugą ir patikimumą, nepaprastai svarbu mokėti kontroliuoti šiuos procesus reaktoriuje. Vienas pagrindinių reaktoriaus aktyviosios zonos parametrų yra neutronų srautas. Jis keičiasi bėgant laikui ir erdvėje, todėl svarbu mokėti iš anksto prognozuoti jo pokyčius kompiuterinėmis programomis. Šiame straipsnyje pristatoma autorių „Matlab" programinėje aplinkoje sukurta programa, kurią naudojant galima modeliuoti vienmatį homogeninį ir heterogeninį neutronų srautą. Programa parašyta naudojant vienos grupės 1D laiko ir erdvės difuzijos lygtį. Siekiant tikslesnių rezultatų, į programą buvo įtrauktos tarpinių vėluojančių neutronų grupės, taip pat ksenono ir jodo pasiskirstymo aktyviojoje zonoje lygtys. Pagrindinė naujovė – skaičiavimams naudoti skaitmeniniai metodai, t. y. nuo vietos ir laiko priklausantiems sprendiniams gauti buvo taikomas baigtinių skirtumų metodas. Lyginant su kitomis panašiomis programomis, ši programa taikant kompleksinius skaitinius metodus leidžia greitai atlikti 1D zonos modeliavimą ir gauti pakankamai tikslius sprendinius.

Raktažodžiai: branduolinė jėgainė, neutronų srautas, modeliavimas, baigtinių skirtumų metodas