Calculation of Reactivity using the Simpson's 3/8 Rule

[1]Daniel Suescún-Díaz, [1]Jesus A. Chala-Casanova and [2]D. Peña Lara
[1]Departamento de Ciencias Naturales, Universidad Surcolombiana, AV. Pastrana, Neiva,
Huila, Colombia
[2]Departamento de Física, Universidad del Valle, A.A 25360 Cali, Colombia
daniel.suescun@usco.edu.co, (57) (8) 8754753

Abstract: In this study, we present a new method of calculating the reactivity using the inverse point kinetic equation. We discretize the history of the neutron population density with the integration method known as Simpson's 3/8. We resolve it numerically with the sum of a convolution. For this study we develop diverse numerical experiments, in which we consider different expressions for the neutron population density and different time steps. When we compare the results acquired with some methods reported in the literature, we find that the proposed method gives correct results and therefore can be used to calculate reactivity correctly.

Key words: Inverse point kinetic equation, reactivity, Simpson's three eights method, convolution, integration method, diverse numerical experiments

INTRODUCTION

Nuclear fission happens inside the core of a nuclear reactor. This fission process is based on the division of heavy atoms, like $^{235}_{92}U$ and $^{239}_{94}P$ into two lighter atoms called "product of fision". Several neutrons can be produced after the fission, freeing up to 200 MeV (Stacey, 2018).

One of the most important parameters, that guarantees the safe operation of a nuclear reactor is the reactivity. Reactivity is obtained from the inverse point kinetic equation. This equation is an integral-differential equation, that contains the history of the neutron population density in the integral side. The differential side is related with the period of the reactor.

Monitoring of the reactivity inside a nuclear reactor guarantees safety and operability of the reactor. By moving the control rods inside the reactor, the density flux in the neutron population produced in the fission events can be controlled (Lamarsh, 2002).

There are different methods in the literature that give solutions to the inverse point kinetic equation to calculate reactivity (Shimazu *et al.*, 1987; Binney and Bakir, 1989; Ansari, 1991; Hoogenboom and Sluijs, 1988; Tamura, 2003). There is also, a reported work that requires the history of the density in the neutron population, known as the Finite Impulse Response (FIR) filter (Diaz *et al.*, 2008). Another method uses the Lagrange polynomial in order to disapear the history of the density of the neutron

population (Malmir and Vosoughi, 2013). Another very precise method known as the Hamming method was presented by Diaz *et al.* (2012). This method discretizes the history.

Due to the importance of knowing the value of the reactivity, we propose solving the inverse point kinetic equation using the integration Simpson's 3/8 method (Mathews and Fink, 2004).

MATERIALS AND METHODS

Theoretical considerations: Inside a nuclear reactor, reactivity can be calculated using the inverse point kinetic equation (Stacey, 2018):

$$\rho(t) = \beta + \frac{\Lambda}{P(t)}\frac{dP(t)}{dt} - \frac{1}{P(t)}\sum_{i=1}^{m}\lambda_i\beta_i\left[\frac{P_0}{\lambda_i}e^{-\lambda_i t} + \int_0^t e^{-\lambda_i(t-t')}P(t')dt'\right]$$

Where: (1)
- ρ = The reactivity
- $P(t)$ = The density of the neutron population
- Λ = The prompt neutron generation time
- λ_i = The decay constant of the ith group of delayed neutrons precursors
- β_i = The effective fraction of the ith group of delayed neutrons
- β = The total effective fraction of delayed neutrons
- P_0 = The intial population of neutrons

Corresponding Author: Daniel Suescún-Díaz, Departamento de Ciencias Naturales, Universidad Surcolombiana, Neiva, Colombia, daniel.suescun@usco.edu.co, (57) (8) 8754753

Assuming that P (t) is known, the problem is reduced to the integral contained in Eq. 1, known as the history of the density of the neutron population. We give solution to Eq. 1, solving the history using the third order Newton-Cortes numerical integration method, also known as the Simpson's 3/8 method.

Simpson's 3/8 method: The Simpson's 3/8 method belongs to the family of Newton-Cortes formulas which calculates the area under the curve of a third degree polynomial by interpolating the curve in four points.

In order to show that the Simpson's 3/8 method can be applied to solve the inverse point kinetic equation, we develop several numerical experiments, in which we calculate the value of the reactivity over an interval of time for different functions of the density of the neutron population such as polynomial, exponential, sinusoidal and hyperbolic. In order to solve numerically Eq. 1, we can rewrite it as:

$$\rho\left(t_{n}\right)=\beta+\frac{\Lambda}{P\left(t_{n}\right)}\frac{dP(t)}{dt}-\frac{1}{P\left(t_{n}\right)}\sum_{i=1}^{m}\lambda_{i}\beta_{i}\left[\begin{array}{c}\dfrac{Pt_{0}}{\lambda_{i}}e^{-\lambda_{i}t_{n}}+\\ \displaystyle\int_{0}^{t_{n}}e^{-\lambda_{i}\left(t_{n}-t_{k}\right)}P\left(t_{k}\right)dt_{k}\end{array}\right]$$

where: $\qquad\qquad\qquad\qquad\qquad\qquad\quad$ (2)

$$t_{0}\equiv\left(t=0\right)$$
$$t\equiv t_{n}\equiv t_{0}+ndt_{n};\,n=1,\,2,\,...,\,n_{max} \qquad (3)$$
$$t'\equiv t_{k}\equiv t_{0}+kdt_{k};\qquad k=1,\,2,\,...,\,k_{max}$$

With the notation defined in Eq. 3, we calculate the derivative numerically, given Eq. 2, using the form:

$$\frac{dP(t)}{dt}=\frac{P\left(t_{n}+\Delta t_{n}\right)-P\left(t_{n}\right)}{\Delta t_{n}} \qquad (4)$$

The history of the density of the neutron population is given by:

$$H\left(t_{k}\right)=\int_{t_{0}}^{t_{n}}e^{-\lambda_{i}\left(t_{n}-t_{k}\right)}P\left(t_{k}\right)\Delta t_{k} \qquad (5)$$

In order to solve Eq. 5 we use a Lagrange interpolating polynomial that is aproximated integrated the following way:

$$\phi\left(t_{k}\right)=e^{-\lambda_{i}\left(t_{n}-t_{k}\right)}P\left(t_{k}\right)\approx Pol_{m}\left(t_{k}\right) \qquad (6)$$

where, Pol_{m} (t_{k}) is a polynomial of degree m that interpolates the function $\bullet$ (t_{k}) in m+1 points. This polynomial is of the Lagrange form and a solution is already known (Burden *et al.*, 2016):

$$Pol_{m}\left(t_{k}\right)=\sum_{p=0}^{m}\left(t_{p}\right)\phi L_{m,p}\left(t_{k}\right);$$
$$L_{m,p}\left(t_{k}\right)=\prod_{q=0,\,q\neq p}^{m}\frac{\left(t_{k}-t_{q}\right)}{\left(t_{p}-t_{q}\right)} \qquad (7)$$

Using Eq. 7 and integrating this polynomial we build:

$$\int_{t_{0}}^{t_{m}}\phi\left(t_{k}\right)\Delta t_{k}=\sum_{p=0}^{m}\phi\left(t_{p}\right)\int_{t_{0}}^{t_{0}+m\Delta t_{k}}\prod_{\substack{q=0\\q\neq p}}^{m}\frac{\left(t_{k}-t_{q}\right)}{\left(t_{p}-t_{q}\right)}\Delta t_{k} \qquad (8)$$

From Eq. 3, we obtain an expresion for $t_{k}=t_{0}+k\Delta t_{k},\,t_{q}=t_{0}+q\Delta t_{k}$, therefore, we can get the following expresion:

$$t_{k}-t_{q}=t_{0}+k\Delta t_{k}-\left(t_{0}+q\Delta t_{k}\right)=\left(k-q\right)\Delta t_{k} \qquad (9)$$

Using the same reasoning we find a way to re-write the denominator in Eq. 8:

$$t_{p}-t_{q}=\left(p-q\right)\Delta t_{k} \qquad (10)$$

Substituting q = k-1 in Eq. 9, we find that $\Delta t_{k}=\Delta t_{k}\Delta k$. This result, along with Eq. 9 and 10 allows to rewrite Eq. 8 into the following form:

$$\int_{t_{0}}^{t_{m}}\phi\left(t_{k}\right)\Delta t_{k}=\sum_{p=0}^{m}\phi\left(t_{p}\right)\Delta t_{k}\int_{0}^{m}\prod_{\substack{q=0\\q\neq p}}^{m}\frac{\left(k-q\right)}{\left(p-q\right)}\Delta k \qquad (11)$$

Taking m = 3 in Eq. 11 and expanding the right hand side of Eq. 11, we obtain:

$$\sum_{p=0}^{m=3}\phi\left(t_{p}\right)\Delta t_{k}\int_{0}^{m=3}\prod_{\substack{q=0\\q\neq p}}^{m=3}\frac{k-q}{p-q}\Delta k=$$
$$\phi\left(t_{0}\right)\Delta t_{k}\int_{0}^{3}\frac{\left(k-1\right)\left(k-2\right)\left(k-3\right)}{\left(0-1\right)\left(0-2\right)\left(0-3\right)}\Delta k+$$
$$\phi\left(t_{1}\right)\Delta t_{k}\int_{0}^{3}\frac{\left(k-0\right)\left(k-2\right)\left(k-3\right)}{\left(1-0\right)\left(1-2\right)\left(1-3\right)}\Delta k+ \qquad (12)$$
$$\phi\left(t_{2}\right)\Delta t_{k}\int_{0}^{3}\frac{\left(k-0\right)\left(k-1\right)\left(k-3\right)}{\left(2-0\right)\left(2-1\right)\left(0-3\right)}\Delta k+$$
$$\phi\left(t_{3}\right)\Delta t_{2}\int_{0}^{3}\frac{\left(k-0\right)\left(k-1\right)\left(k-2\right)}{\left(3-0\right)\left(3-1\right)\left(3-2\right)}\Delta k$$

Solving the integrals in Eq. 12 it is possible to obtain:

$$\int_{t_0}^{t_3}\phi(t_k)\Delta t_k = \sum_{p=0}^{3}\phi(t_p)\Delta t_k \int_0^3 \prod_{\substack{q=0\\q\neq p}}^{3}\frac{(k-q)}{(p-q)}\Delta k =$$

$$\frac{3}{8}\left[\phi(t_0)+3\phi(t_1)+3\phi(t_2)+\phi(t_3)\right]\Delta t_k \qquad (13)$$

The right hand side of Eq. 13, happens to be the Simpson's 3/8 rule (Mathews and Fink, 2004). This rule is used to find the area under the curve for a function defined between t_0 and t_3, however, the function • (t_k) is discretized in k parts with k = 0, 1, 2, 3, ..., n. Where, n is a mutiple of 3. Therefore, we can write:

$$\int_{t_0}^{t_n}\phi(t_k)\Delta t_k = \int_{t_0}^{t_3}\phi(t_k)\Delta t_k + \int_{t_3}^{t_6}\phi(t_k)\Delta t_k +,...,+$$

$$\int_{t_{3k-3}}^{t_{3k}}\phi(t_k)\Delta t_k = \sum_{k=1}^{\frac{n}{3}}\int_{t_{3k-3}}^{t_{3k}}\phi(t_k)\Delta t_k \qquad (14)$$

Using Eq. 13 to solve Eq. 14, we get:

$$\int_{t_0}^{t_3}\phi(t_k)\Delta t_k = \sum_{k=1}^{\frac{n}{3}}\int_{t_{3k-3}}^{t_{3k}}\phi(t_k)\Delta t_k =$$

$$\sum_{k=1}^{\frac{n}{3}}\frac{3}{8}\left[\phi(t_{3k-3})+3\phi(t_{3k-2})+3\phi(t_{3k-1})+\phi(t_{3k})\right]\Delta t_k \qquad (15)$$

Equation 15 is known as the composite Simpson's 3/8 rule. Now we have $\phi(t_k)=h(t_k)P(t_k)$ with $h(t_k)=e^{-\lambda_i(t_n-t_k)}$. We can rewrite the history of the neutron population density using the following form:

$$H(t_k)=\int_{t_0}^{t_n}h(t_k)P(t_k)\Delta t_k =$$

$$\frac{3}{8}\Delta t_k \sum_{k=1}^{\frac{n}{3}}\left[\begin{array}{c}h(t_{3k-3})P(t_{3k-3})+3h(t_{3k-2})P(t_{3k-2})+\\3h(t_{3k-1})P(t_{3k-1})+h(t_{3k})P(t_{3k})\end{array}\right] \qquad (16)$$

Equation 16 can be simplified to:

$$H(t_k)=\int_{t_0}^{t_n}h(t_k)P(t_k)\Delta t_k =$$

$$\frac{3}{8}\Delta t_k \sum_{k=1}^{\frac{n}{3}}\sum_{z=0}^{3}C(z)h(t_{3k-z})P(t_{3k-z}) \qquad (17)$$

where, C is a vector that contains the integration of the Lagrange coefficients, such that C = [1, 3, 3, 1] for Simpson's 3/8 as was shown on Eq. 13. Using Eq. 3 and 17 we can rewrite Eq. 2 as:

$$\rho(t)=\beta + \frac{\Lambda}{P(t)}\left[\frac{P(t_{n+1})-P(t_n)}{\Delta t_n}\right]-$$

$$\frac{1}{P(t)}\sum_{i=1}^{m}\lambda_i\beta_i\left[\frac{Pt_0}{\lambda_i}e^{-\lambda_i t_n}+\frac{3}{8}\Delta t_k \sum_{k=1}^{\frac{n}{3}}\sum_{z=0}^{3}C(z)h(t_{3k-z})P(t_{3k-z})\right] \qquad (18)$$

We will use Eq. 18 in the following section to study the efectivity of our method to calculate the reactivity for different densities of the neutron population.

RESULTS AND DISCUSSION

In this study we show numerical experiments simulating the behaviour of a nuclear reactor for different densities of the neutron population. Equation 18 represents the method proposed using the Simpson's 3/8 rule. We obtain the reference method solving Eq. 1 analitically. The densities of the neutron population have histories with analitical solutions given in Table 1. These solutions allow having reference values to compare with the method at hand.

The parameters used in the numerical experiments are presented in Table 2. We use eight different neutron densities and calculate the reactivity with time steps of 0.5, 0.1, 0.05 and 0.01 sec.

Table 1: Analitical solutions of the history of the neutron population density in the inverse kinetic equation for some shapes of the neutron population density

$P(t_k)$	$\int_0^t e^{-\lambda_i t_k}P(t_k)dt_k$
e^{wt_k}	$\dfrac{e^{(\lambda_i+\omega)t}-1}{\lambda_i+\omega}$
$a+\omega t_k$	$\dfrac{a}{\lambda_i}(e^{\lambda_i t}-1)+\dfrac{\omega}{\lambda_i^2}\left[e^{\lambda_i t}(\lambda_i t-1)+1\right]$
$a+bt_k^2$	$\dfrac{a}{\lambda_i}(e^{\lambda_i t}-1)+\dfrac{b}{\lambda_i^3}\left\{e^{\lambda_i t}\left[\lambda_i t(\lambda_i t-2)+2\right]-2\right\}$
$a+bt_k^3$	$\dfrac{a}{\lambda_i}(e^{\lambda_i t}-1)+\dfrac{b}{\lambda_i^4}\left\{e^{\lambda_i t}\left[\lambda_i t(\lambda_i t(\lambda_i t-3)+6)-6\right]+6\right\}$
$a+b\sin(\omega t_k)$	$\dfrac{a}{\lambda_i}(e^{\lambda_i t}-1)+\dfrac{b}{\lambda_i^2+\omega^2}\left\{e^{\lambda_i t}\left[\lambda_i\sin(\omega t)-\omega\cos(\omega t)\right]+\omega\right\}$
$a+b\cos(\omega t_k)$	$\dfrac{a}{\lambda_i}(e^{\lambda_i t}-1)+\dfrac{b}{\lambda_i^2+\omega^2}\left\{e^{\lambda_i t}\left[\omega\sin(\omega t)-\lambda_i\cos(\omega t)\right]-\lambda_i\right\}$
$a+b\sinh(\omega t_k)$	$\dfrac{a}{\lambda_i}(e^{\lambda_i t}-1)+\dfrac{b}{\lambda_i^2+\omega^2}\left\{e^{\lambda_i t}\left[\lambda_i\sinh(\omega t)-\omega\cosh(\omega t)\right]+\omega\right\}$
$a+b\cosh(\omega t_k)$	$\dfrac{a}{\lambda_i}(e^{\lambda_i t}-1)+\dfrac{b}{\lambda_i^2+\omega^2}\left\{e^{\lambda_i t}\left[\lambda_i\cosh(\omega t)-\omega\sinh(\omega t)\right]+\lambda_i\right\}$

Table 2: Parameters for the numerical experiments for all six precursor groups

Parameters	Values
β_1	0.000266
β_2	0.001491
β_3	0.001316
β_4	0.002849
β_5	0.000896
β_6	0.000182
λ_1	0.012700
λ_2	0.031700
λ_3	0.115000
λ_4	0.311000
λ_5	1.400000
λ_6	3.870000

$a = 1$, $b = \bullet/10$, $\bullet = 0.00002$

Table 3: Greatest diferences (in pcm) for the first numerical experiment with $w = 0.12353$ sec^{-1} and a simulation time of $t = 100$ sec

$P(t_k)$	dt = 0.5[sec]	dt = 0.1[sec]	dt = 0.05[sec]	dt = 0.01[sec]
$e^{\bullet t_k}$	2.0706	5.9628×10^{-3}	3.8236×10^{-4}	6.1692×10^{-7}
$a+\bullet t_k$	2.0145	5.7905×10^{-3}	3.7123×10^{-4}	5.9893×10^{-7}
$a+bt^2_k$	2.7413	8.4593×10^{-3}	5.4519×10^{-4}	8.8117×10^{-7}
$a+bt^3_k$	3.8407	1.2645×10^{-2}	8.1930×10^{-4}	1.3274×10^{-6}
$a+b\sin(\bullet\, t_k)$	1.9531	5.5527×10^{-3}	3.5576×10^{-4}	5.7384×10^{-7}
$a+b\cos(\bullet\, t_k)$	1.9529	5.5527×10^{-3}	3.5576×10^{-4}	5.7384×10^{-7}
$a+b\sinh(\bullet\, t_k)$	2.0706	5.9627×10^{-3}	3.8236×10^{-4}	6.1691×10^{-7}
$a+b\cosh(\bullet\, t_k)$	2.0706	5.9627×10^{-3}	3.8236×10^{-4}	6.1691×10^{-7}

Table 4: Greatest diferences (in pcm) for the second numerical experiment with $w = 1.008477$ sec^{-1} and a simulation time of $t = 100$ sec

$P(t_k)$	dt = 0.5[sec]	dt = 0.1[sec]	dt = 0.05[sec]	dt = 0.01[sec]
$e^{\bullet t_k}$	3.7235	1.2121×10^{-2}	7.8440×10^{-4}	1.2695×10^{-6}
$a+\bullet t_k$	2.3967	7.4770×10^{-3}	4.8177×10^{-4}	7.7852×10^{-7}
$a+bt^2_k$	2.7413	8.4593×10^{-3}	5.4519×10^{-4}	8.8117×10^{-7}
$a+bt^3_k$	3.8407	1.2645×10^{-2}	8.1930×10^{-4}	1.3274×10^{-6}
$a+b\sin(\bullet\, t_k)$	2.3934	7.1079×10^{-3}	4.5721×10^{-4}	7.3814×10^{-7}
$a+b\cos(\bullet\, t_k)$	2.3938	7.1070×10^{-3}	4.5705×10^{-4}	7.3813×10^{-7}
$a+b\sinh(\bullet\, t_k)$	3.7235	1.2121×10^{-2}	7.8440×10^{-4}	1.2695×10^{-6}
$a+b\cosh(\bullet\, t_k)$	3.7235	1.2121×10^{-2}	7.8440×10^{-4}	1.2695×10^{-6}

Table 3 and 4 have information about the maximum differences between the results in the experiments, obtained by Eq. 18 and the analitical solution given by Eq. 2 with the registered solutions on Table 1. For the first experiment we used $w = 0.12353$ sec^{-1} and for the sec we used $w = 1.00847$ sec^{-1}. The results show that the method converges to the solution for small time steps. The error obtained for each time step does not show any significant dependency in the neutron population density. Also can be seen for a neutron population density with no dependency on w such as $P(t_k) = 1 + bt^3_k$ and $P(t_k) = 1 + bt^2_k$ that the results show no changes in the maximum difference.

On Table 5 we register the maximun difference for the third numerical experiment. For this experiment we use an exponential function to simulate the neutron population density and some of the w values frecuently used in the literature. On Table 5 we observe that the value on the maximum diference on the method proposed does not

Table 5: Greatest diferences (in pcm) for the third experiment with differentwand a neutron population density $P(t) = e^{wt_k}$

$\bullet$ [sec^{-1}]	t[sec]	dt = 0.5 [sec]	dt = 0.1 [sec]	dt = 0.05 [sec]	dt = 0.01 [sec]
0.00243	100	1.9004	5.3700×10^{-3}	3.4393×10^{-4}	5.5468×10^{-7}
0.01046	100	1.9113	5.4078×10^{-3}	3.4637×10^{-4}	5.5864×10^{-7}
0.02817	100	1.9356	5.4918×10^{-3}	3.5182×10^{-4}	5.6745×10^{-7}
0.12353	100	2.0706	5.9628×10^{-3}	3.8236×10^{-4}	6.1692×10^{-7}
1.00847	100	3.7235	1.2121×10^{-2}	7.8440×10^{-4}	1.2695×10^{-6}
2.345	100	7.5608	2.9720×10^{-2}	1.9537×10^{-3}	3.1790×10^{-6}
11.6442	60	3.4118	6.2469×10^{-1}	5.0956×10^{-2}	9.0281×10^{-5}
52.80352	10	5.1520×10^{1}	5.9695	1.3177	5.7269×10^{-3}

Table 6: Comparison with other methods with $w = 0.12353$ sec^{-1}, time of simulation $t = 100$ sec and $P(t) = e^{wt_k}$

dt [sec]	FIR	Trapezoidal rule	Hammnig	Simpson 3/8
0.5	2.4400×10^{2}	7.0562×10^{1}	4.727×10^{1}	5.1520×10^{1}
0.1	8.9341×10^{1}	9.7611	5.13	5.9695
0.05	2.3719	3.0888	1.00	1.3177
0.01	1.5390×10^{-1}	1.3737×10^{-1}	2.33×10^{-3}	5.7269×10^{-3}

Table 7: Comparison with other methods with $w = 52.80352$ sec^{-1}, time of simulation $t = 100$ sec and $P(t) = e^{wt_k}$

dt [sec]	FIR	Trapezoidal rule	5-Points Lagrange	Simpson 3/8
0.5	8.6509×10^{1}	1.0341×10^{1}	2.78×10^{2}	2.0706
0.1	1.5662×10^{1}	4.2872×10^{1}	4.50×10^{2}	5.9628×10^{-3}
0.05	3.2054	1.0731×10^{1}	1.24×10^{3}	3.8236×10^{-4}
0.01	1.5277	4.2940×10^{1}	1.24×10^{3}	6.1692×10^{-7}

show dependence on the value of w when this parameter is smaller than 1. When w is >1, we reduced the time step to compensate the loss on precision.

Next, we compare the integration method using Simpon's 3/8 rule with other methods reported in the literature. On Table 6 we compare the method we show in this document with the FIR filter method, the trapezoidal rule and the Hamming method, we clearly observe that the method we choose is superior at converging to the solution compared to: FIR filter and the trapezoidal rule. With respect to the Hamming method, ours mantains the same precision. On Table 7, we include the comparison with the 5 points lagrange method presented on Malmir and Vosoughi (2013). That method consists on derivating a forth degree polynomial while the Simpson's 3/8 method consists on integrating a third degreepolynomial which results on better precision (for small time steps), although, both methods are based on the Lagrange interpolation polynomial.

On Fig. 1 and 2, we observe that the behaviour in the third degree polynomial and the convergency of the method to the solution. Simpson's 3/8 is a method that aproximates to the solution each three points. The oscilations observed on Fig. 1 and 2 are due to the values that that takes the aproximation when it is evaluated on a point that is not a multiple of 3. Taking only the values
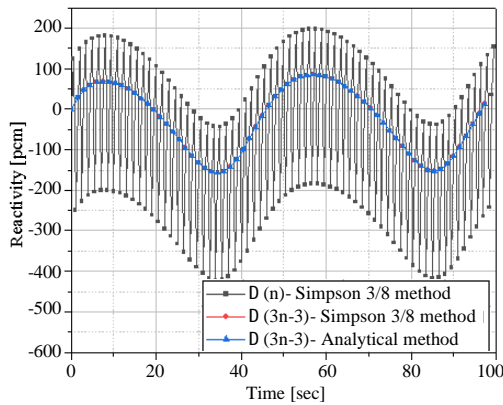
Fig. 1: Oscilation in the reactivity for the neutron population density $p(t_k) = a + b \sin(wt_k)$ with $w = 0.12353 \ sec^{-1}$ and $dt = 0.5 \ sec$

Fig. 2: Oscilation in the reactivity for the neutron population density $p(t_k) = a + b \sin(wt_k)$ with $w = 0.12353 \ sec^{-1}$ and $dt = 0.05 \ sec$

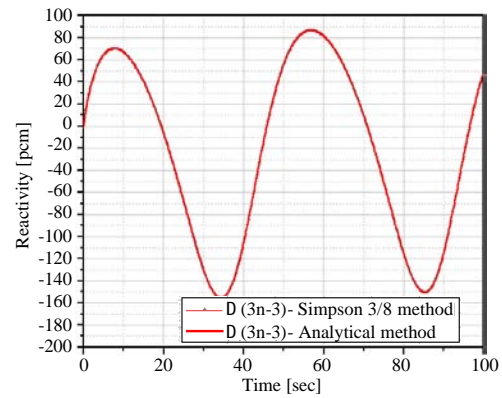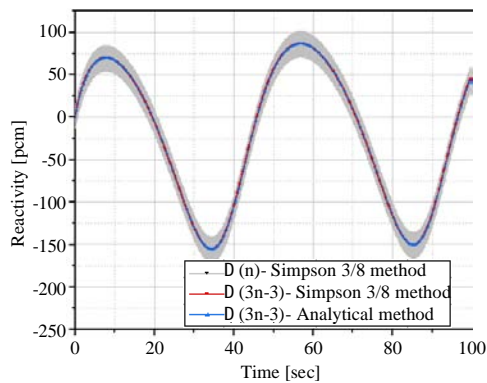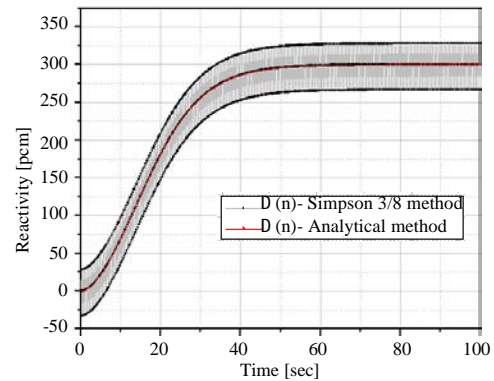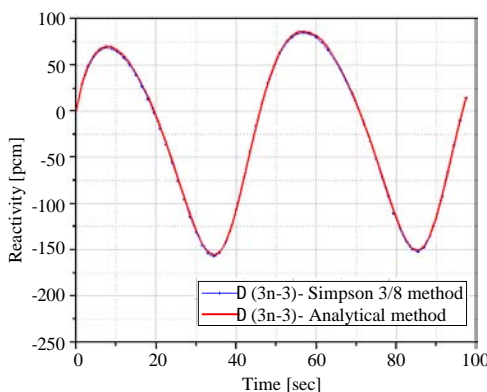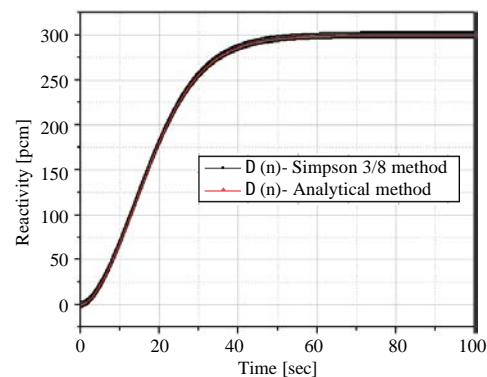Fig. 3: Calculation of the reactivity for the neutron population density $p(t_k) = a + b \sin(wt_k)$ with $w = 0.12353 \ sec^{-1}$ and $dt = 0.5 \ sec$

Fig. 4: Calculation of the reactivity for the neutron population density $p(t_k) = a + b \sin(wt_k)$ with $w = 0.12353 \ sec^{-1}$ and $dt = 0.05 \ sec$

Fig. 5: Oscilation in the reactivity for the neutron population density $p(t_k) = a + b \cosh(wt_k)$ with $w = 0.12353 \ sec^{-1}$ and $dt = 0.1 \ sec$

Fig. 6: Oscilation in the reactivity for the neutron population density $p(t_k) = a + b \cosh(wt_k)$ with $w = 0.12353 \ sec^{-1}$ and $dt = 0.01 \ sec$

that correspond to 3n with n = 0, 1, 2, 3, ..., n_{max} we obtain Fig. 3 and 4 which coresponds to the natural behaviour of the method (Fig. 5 and 6). It is worth noting that the curves • (n) for Simpsons 3/8 on Fig. 1 and 2 are only
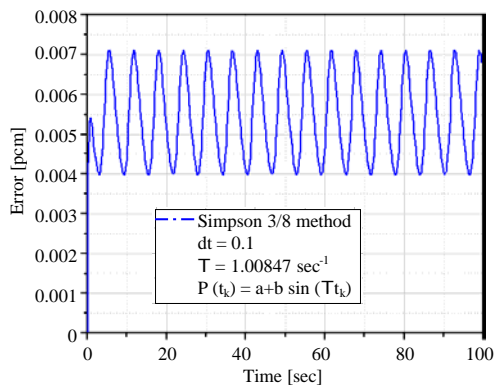
Fig. 7: Error in the Simpson's 3/8 method (in pcm) for the neutron population density p (t_k) = a+b sin (wt_k): with w = 1.00847 sec^{-1} and dt = 0.1 sec
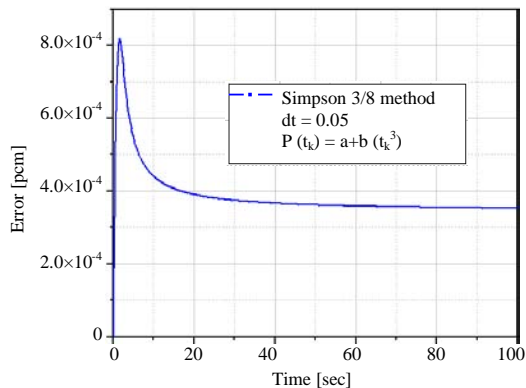

Fig. 8: Error in the Simpson's 3/8 method (in pcm) for the neutron population density: p (t_k) = a+bt_k^3 with dt = 0.05 sec

shown to examing the behaviour of the third degree polynomial which is integrated and then used to approximate the history of the neutron population density. The oscilations generated by the polynomial are reduced when the the time steps are smaller. On Fig. 5 and 6 we present the results of the other numerical experiment which show the change on the oscilations when we change the time steps for a neutron population density of the form P (t_k) = a+b cosh (w t_k). Figure 5 corresponds to the oscilations for a time step of 0.1 sec and in Fig. 6 the oscilations are due to a time step of 0.01 sec.

In this research we use a maximum time of t = 100 sec for most of the numerical experiments shown. Other methods run the simulation for times of t =1000 sec and even t = 10000 sec. This decision is justified because for numerical experiments realized after a time t, the error of

the aproximation in the metod becomes constant or it oscilates around a constant value because the reactivity has the same behaviour. To show this tendency, we include Fig. 7 and 8.

CONCLUSION

We presented a new numerical way to calculate reactivity using a third degree Lagrange Interpolation polynomial. We analytically showed how our method is related to the Simpson's 3/8 method. We showed a way to solve the history of the neutron population density given different functional shapes for it and also using different time steps. Although, this method is very simple in its numerical implementation, it is a powerfull tool given the precision we achieved. This is reinforced by the maximum errors shown in the numerical experiments. Due to its simplicity and precision, it is recommended this method to be implemented on digital reactivity meter.

ACKNOWLEDGEMENT

REFERENCES

Ansari, S.A., 1991. Development of on-line reactivity meter for nuclear reactors. IEEE. Trans. Nucl. Sci., 38: 946-952.

Binney, S.E. and A.J.M. Bakir, 1989. Design and development of a personal-computer-based reactivity meter for a research reactor. Nucl. Technol., 85: 12-21.

Burden, R.L., J.D. Faires and A.M. Burden, 2016. Numerical Analysis. 10th Edn., Cengage Learning, Boston, Massachusetts, USA., ISBN:9789814834285, Pages: 896.

Diaz, D.S., A.S. Martinez and F.C. Da Silva, 2008. Calculation of reactivity using a finite impulse response filter. Ann. Nucl. Energy, 35: 472-477.

Diaz, D.S., J.F.F. Ospina and J.A.R. Sarasty, 2012. Hamming method for solving the delayed neutron precursor concentration for reactivity calculation. Ann. Nucl. Energy, 42: 47-49.

Hoogenboom, J.E. and A.R.V.D. Sluijs, 1988. Neutron source strength determination for on-line reactivity measurements. Ann. Nucl. Energy, 15: 553-559.

Lamarsh, J.R., 2002. Introduction to Nuclear Reactor Theory. American Nuclear Society, Illinois, USA., ISBN-13:9780894480409.

Malmir, H. and N. Vosoughi, 2013. On-line reactivity calculation using Lagrange method. Ann. Nucl. Energy, 62: 463-467.

Mathews, J.H. and K.D. Fink, 2004. Numerical Methods using MATLAB. 4th Edn., Prentice Hall, Upper Saddle River, New Jersy, USA., ISBN: 9780131911789, Pages: 696.

Shimazu, Y., Y. Nakano, Y. Tahara and T. Okayama, 1987. Development of a compact digital reactivity meter and a reactor physics data processor. Nucl. Technol., 77: 247-254.

Stacey, W., 2018. Nuclear Reactor Physics. 3rd Edn., Wiley-VCH Verlag, Weinheim, Germany, ISBN:978-3-527-41366-9, Pages: 766.

Tamura, S., 2003. Signal fluctuation and neutron source in inverse Kinetics method for reactivity measurement in the sub-critical domain. J. Nucl. Sci. Technol., 40: 153-157.