LLNL-TR-789326

LAWRENCE LIVERMORE NATIONAL LABORATORY

Solution of the Feynman Equation for Time Evolving Fission Chains

N. J. Snyderman, H. Chandrasekaran

September 10, 2019

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

Solution of the Feynman Equation for Time Evolving Fission Chains

Neal Snyderman and Hema Chandrasekaran
Lawrence Livermore National Laboratory

Abstract

The partial differential equation proposed by Feynman describing the time evolution of a fission chain in metal is solved by a method also proposed by Feynman. The partial differential equation is reduced to a transcendental algebraic equation. The solution involves roots of a 7th order polynomial. Both the roots and the transcendental equation are then solved numerically. Two populations are computed from the probability generating function, the internal number of neutrons in time, and the accumulated number of leaked neutrons in time. This same equation also describes a limit of a thermal reactor, where induced fission is totally driven by neutrons that have thermalized and are diffusing in a moderator.

Feynman equation for time evolving fission chains

The generating function for the internal population of a time evolving fission chain is,

$$f(t,x) = \sum_{\nu=0}^{\infty} P_\nu(t)\, x^\nu \tag{1}$$

where $P_\nu(t)$ is the probability that, starting from a single neutron at $t = 0$, there are ν neutrons in the system at time t. The generating function can be determined from the solution to the Feynman equation [1], (see also[2], and [5] for details),

$$\frac{\partial f(t,x)}{\partial t} = \frac{1}{\tau}[-x + q + pC(x)]\frac{\partial f(t,x)}{\partial x} \tag{2}$$

where $C(x)$ is the induced fission neutron number generating function,

$$C(x) = C_0 + C_1 x + C_2 x^2 + \cdots + C_7 x^7 \tag{3}$$

where C_i is the probability that i neutrons are created in an induced fission. The parameter p is the probability a neutron induces fission, and is related to multiplication through,

$$M = \frac{1}{1 - p\bar{\nu}} \tag{4}$$

where $\bar{\nu} = \sum_{\nu=1}^{7} \nu C_\nu$ is the average number of neutrons created by induced fission. Also, $q = 1 - p$ is the probability that an internal neutron leaks from the system. Feynman showed [1] that this equation has solution of the form,

$$G[f(t,x)] = \frac{t}{\tau} + G(x) \tag{5}$$

where

$$G(x) = \int \frac{dx}{-x + q + pC(x)} \tag{6}$$

That is,

$$f(t,x) = G^{-1}\left[\frac{t}{\tau} + G(x)\right] \tag{7}$$

This form shows the realization of the initial conditions, $f(t = 0, x) = G^{-1}[G(x)] = x$, that there is a single neutron in the system at $t = 0$. The integral for $G(x)$ in equation (6) can be solved by factoring the polynomial denominator and then by using partial fractions. (Feynman solved this equation in the quadratic approximation to the polynomial $C(x)$.) The induced fission neutron number generating function $C(x)$ from equation (3) can be re-expressed as,

$$C(x) = 1 - \bar{\nu}(1 - x) + \nu_2(1 - x)^2 - \nu_3(1 - x)^3 + \cdots \tag{8}$$

where generally,

$$\nu_i = \sum_{\nu=i}^{7} \binom{\nu}{i} C_\nu \tag{9}$$

To keep track of the neutrons that leak from the system, the external population, another generating function variable y can be introduced through $q \to qy$. The generating function variable y acts as a parameter (and will usually be notationally suppressed in $f(t, x, y)$). In terms of these variables the denominator of the integral in equation (6) can be re-written

$$\begin{aligned} &-x + qy + pC(x) \\ &= 1 - x - 1 + qy + p[1 - \bar{\nu}u + \nu_2 u^2 - \cdots - \nu_7 u^7] \\ &= -M^{-1}(M - 1)\Big[D_7 u^7 - D_6 u^6 + D_5 u^5 - D_4 u^4 + D_3 u^3 - D_2 u^2 - \frac{1}{M-1} u \\ &\qquad + \frac{qM}{M-1}(1 - y)\Big] \end{aligned} \tag{10}$$

where $1 - x = u$, $M^{-1} = (1 - p\bar{\nu})$, $p(1 - p\bar{\nu})^{-1} = pM = (M - 1)/\bar{\nu}$, and the D_i are defined,

$$D_i = \frac{\nu_i}{\bar{\nu}} \tag{11}$$

This polynomial expression in equation (10) can be factorized into the form,

$$-M^{-1}(M-1)D_7[(u-u_1)(u-u_2)(u-u_3)(u-u_4)(u-u_5)(u-u_6)(u-u_7)] \quad (12)$$

where $u_1, \cdots, u_7$ are the roots of the equation,

$$u^7 - \frac{D_6}{D_7}u^6 + \frac{D_5}{D_7}u^5 - \frac{D_4}{D_7}u^4 + \frac{D_3}{D_7}u^3 - \frac{D_2}{D_7}u^2 - \frac{1}{(M-1)D_7}u + \frac{qM}{(M-1)D_7}(1-y) = 0 \quad (13)$$

are functions of y, the generating function variable for the external population, but constants for the internal x distribution, with $y = 1$. (The quadratic approximation, including y, and a generating function variable z to keep track of fissions, and therefore fission gamma-rays in time, was solved analytically in [2].)

After factorization of the polynomial denominator, the integral over the variable u,

$$G(u = 1-x) = \frac{M}{(M-1)D_7}\int \frac{du}{[(u-u_1)(u-u_2)(u-u_3)(u-u_4)(u-u_5)(u-u_6)(u-u_7)]} \quad (14)$$

can be performed by partial fractions, which isolates each factor of $\frac{1}{u-u_i}$, where u_i is one of the roots of the polynomial above. The partial fraction decomposition is written as,

$$\frac{1}{[(u-u_1)(u-u_2)(u-u_3)(u-u_4)(u-u_5)(u-u_6)(u-u_7)]} = \sum_{i=1}^{7}\frac{A_i}{(u-u_i)}$$

Each A_i can be obtained as [6]

$$A_i = \lim_{(u \to u_i)}\left(\frac{(u-u_i)}{[(u-u_1)(u-u_2)(u-u_3)(u-u_4)(u-u_5)(u-u_6)(u-u_7)]}\right)$$

Thus, we have,

$$A_1^{-1} = (u_1-u_2)(u_1-u_3)(u_1-u_4)(u_1-u_5)(u_1-u_6)(u_1-u_7)$$
$$A_2 = -\frac{1}{(u_1-u_2)(u_2-u_3)(u_2-u_4)(u_2-u_5)(u_2-u_6)(u_2-u_7)} \quad (15)$$
$$A_3 = \frac{1}{(u_1-u_3)(u_2-u_3)(u_3-u_4)(u_3-u_5)(u_3-u_6)(u_3-u_7)}$$

$$A_4 = -\frac{1}{(u_1 - u_4)(u_2 - u_4)(u_3 - u_4)(u_4 - u_5)(u_4 - u_6)(u_4 - u_7)}$$

$$A_5 = \frac{1}{(u_1 - u_5)(u_2 - u_5)(u_3 - u_5)(u_4 - u_5)(u_5 - u_6)(u_5 - u_7)}$$

$$A_6 = -\frac{1}{(u_1 - u_6)(u_2 - u_6)(u_3 - u_6)(u_4 - u_6)(u_5 - u_6)(u_6 - u_7)}$$

$$A_7 = \frac{1}{(u_1 - u_7)(u_2 - u_7)(u_3 - u_7)(u_4 - u_7)(u_5 - u_7)(u_6 - u_7)}$$

The result of integration is a sum of $Log(u - u_i)$ terms, with root dependent coefficients A_i given in equation (15), which has the form,

$$\begin{aligned} G(u = 1 - x) &= \frac{M}{(M-1)D_7} \sum_{i=1}^{7} -A_i \, Log(u - u_i) \\ &= -\frac{MA_1}{(M-1)D_7} \left[Log(u - u_1) + \sum_{i=2}^{7} Log(u - u_i)^{A_1^{-1} A_i} \right] \\ &= -\frac{MA_1}{(M-1)D_7} Log \left[\prod_{i=1}^{7} (u - u_i)^{A_1^{-1} A_i} \right] \end{aligned} \tag{16}$$

One root u_1 is selected out, with the property that $u_1(y)$ is real and positive for $y = 0$, and less than 1, and vanishes for $y = 1$. As a convention, a minus sign is chosen in the definition of A_i so that the coefficient A_1 accompanying the u_1 term is real and positive. (The reason for this choice will become clear from equation (18) below). Inserting this result for $G(u = 1 - x)$ into the functional Feynman equation, $G[f(t, x)] = \frac{t}{\tau} + G(x)$, dividing by $\frac{M}{(M-1)D_7} A_1$, and exponentiating, where,

$$\begin{aligned} \alpha &= 1/M\tau \\ \alpha' &= \alpha A_1^{-1}(M - 1)D_7 \end{aligned} \tag{17}$$

gives the equation for $f(t, x)$,

$$[1 - f - u_1][1 - f - u_2]^{A_2/A_1}[1 - f - u_3]^{A_3/A_1}[1 - f - u_4]^{A_4/A_1}[1 - f - u_5]^{A_5/A_1}[1 - f - u_6]^{A_6/A_1}[1 - f - u_7]^{A_7/A_1} \tag{18}$$

$$= e^{-\alpha' t}[u-u_1][u-u_2]^{A_2/A_1}[u-u_3]^{A_3/A_1}[u-u_4]^{A_4/A_1}[u-u_5]^{A_5/A_1}[u - u_6]^{A_6/A_1}[u-u_7]^{A_7/A_1}$$

where $u = 1 - x$ in the right-hand side. The partial differential equation for $f(t,x,y)$ has been reduced to a transcendental equation, that must be solved numerically. (In the quadratic approximation to the polynomial $C(x)$, the transcendental equation becomes algebraic, and can therefore be solved explicitly [2].) As $t \to \infty$, the right-hand side $\to 0$, and the vanishing of the left-hand side gives the Böhnel chain as one of the solutions,

$$1 - f(t,y) - u_1 \to 1 - h(y) - u_1(y) = 0 \tag{19}$$

As $t \to \infty$, there are no longer any internal neutrons in the system, so the asymptotic solution is independent of x.

Solution of Transcendental Equation

The first step in solving the transcendental equation for f is to solve for the roots of the polynomial equation (13). The roots of equation (13) must be solved numerically. For the internal neutron population, the roots are constants. There are three real roots and two pairs of complex conjugate roots. For the external neutron population, the roots depend on the generating function variable y; there are two pairs of complex conjugate roots and three real-valued roots for $y\ real$. In order to determine the external probabilities from the generating function, the variable y will be made complex-valued, making all the roots complex.

Fast Fourier Transform (FFT) Inversion of Generating Function $\boldsymbol{f(t,x)}$

The coefficients $P_\nu(t)$ in equation (1) are determined from the generating function by inverse Z transform ([3],[4]),

$$P_\nu(t) = \frac{1}{2\pi i}\oint_C f(t,x)x^{-\nu-1}\,dx \tag{20}$$

where the C is a counterclockwise closed contour in the region of convergence (ROC) of $f(t,x)$. Convergence of $f(t,x)$ in equation (1) is dependent only on $|x|$ for the following reasons:

$$0 \le P_\nu(t) \le 1\ \forall\ \nu$$

$$\sum_{\nu=0}^{\infty} P_\nu(t) = 1$$

$|f(t,x)| < \infty$ if $\sum_{\nu=0}^{\infty} P_\nu(t)\ |x|^\nu < \infty$, the ROC of equation (1) consists of all values of $|x| \le 1$. This ROC includes the unit circle. The inverse Z transform in equation (20) evaluated on the unit circle contour C, defined by $x = e^{i\theta}$, reduces to the Fourier transform. With $x = e^{i\theta}$, equation (20) becomes

$$P_\nu(t) = \frac{1}{2\pi}\int_{-\pi}^{\pi} f\left(t, e^{i\theta}\right)e^{-i\nu\theta}\, d\theta \tag{21}$$

which is a Fourier series expansion. First equation (18) is solved for discrete θ and discrete t_i which when substituted in equation(21) yields the result:

$$P_{t_i}(\nu) = \sum_{n=0}^{N-1} f\left(t_i, e^{i2\pi n/N}\right) e^{-i\nu 2\pi n/N} \tag{22}$$

Equation (22) can be efficiently computed using FFT algorithms. $P_{t_i}(\nu)$ are the internal neutron probabilities at time t_i and the computation is repeated for $0 \le t_i \le t_s$ till $\left|P_{t_{i-1}}(\nu) - P_{t_i}(\nu)\right| < \epsilon$.

Internal Neutron Population Distributions

For the internal distribution, equation (13) reduces to,

$$u^7 - \frac{D_6}{D_7}u^6 + \frac{D_5}{D_7}u^5 - \frac{D_4}{D_7}u^4 + \frac{D_3}{D_7}u^3 - \frac{D_2}{D_7}u^2 - \frac{1}{(M-1)D_7}u = 0 \tag{23}$$

One root is $u = 0$. The numerical solution for the roots $u_1, \cdots, u_7$ of Eq. (18), for an $M = 10$ HEU system, are listed below:

Real Part	Imaginary Part
0.000000000000	0.000000000000
-0.104093723091	0.000000000000
2.183256334534	7.038915914296
2.183256334534	-7.038915914296
2.480453594573	3.783574671759
2.480453594573	-3.783574671759
5.330347424049	0.000000000000

The numerical solution of the transcendental equation (18) for the internal distribution uses as a starting solution, the analytic solution from the quadratic approximation, which corresponds to the high M limit,

$$f(t,x) = 1 - \frac{(1-x)e^{-\alpha t}}{1 + \frac{M-1}{\bar{\nu}}\nu_2(1-x)(1-e^{-\alpha t})} \tag{24}$$

The pseudo-code for calculating the numerical solution $f(t,x)$ to the transcendental equation (18) is given below:

Algorithm 1: Computing the solution $f(t,x)$ to the transcendental equation (18) for internal neutron population

---

Internal-Neutron-Population-Generating-Function (*Multiplier, M*)

in: HEU or Pu nuclear data, multiplication of the system

out: (n by m) array of internal neutron generating function $f\left(t, x = e^{i\theta}\right)$, n values of t, m values of θ

1: Get nuclear data and compute C_i, ν_i, and D_i for $i = 1, 2, .., 7$ in equation (9) and equation (11)

2: Compute the roots of the polynomial in equation (23)

3: Choose $0 \leq \frac{t}{\tau} \leq 50$ (τ = 1 shake) array and $0 \leq \theta \leq \pi$ array (non-linearly spaced arrays) and $\theta_{critical} = \frac{\pi}{6}$

4: **for** each $t_i \in t$ **do**

5: **for** each $\theta_j \in \theta$ **do**

6: Compute approximate solution to internal population generating function using equation (21)

$$f\left(t_i, \theta_j\right)_{approximate} = 1 - \frac{\left(1 - e^{i\theta_j}\right)e^{-\alpha t_i}}{1 + \frac{M-1}{\bar{\nu}}\nu_2\left(1 - e^{i\theta_j}\right)\left(1 - e^{-\alpha t_i}\right)}$$

7: **if** $\theta_j \leq \theta_{critical})$ **then**

initial solution = $f\left(t_i, \theta_j\right)_{approximate}$

else

initial solution = $f\left(t_i, \theta_{j-1}\right) + \frac{f\left(t_i, \theta_{j-3}\right) - f\left(t_i, \theta_{j-2}\right)}{2}$

end

8: compute exact solution $f\left(t_i, \theta_j\right)$ to the transcendental equation (18) using the initial solution from step 7

9: **end for**

10: **end for**

11: **return** $f(t, \theta)$ ⊳ complex-valued

Figure 1 shows the plot of the solution to the transcendental equation $f(t, x = e^{i\theta})$ with M=10. Note that as $t \to \infty$, $f(t,x) \to 1$, that is $P_0^{int}(t) \to 1$ as $t \to \infty$, and all other $P_\nu^{int}(t) \to 0$.

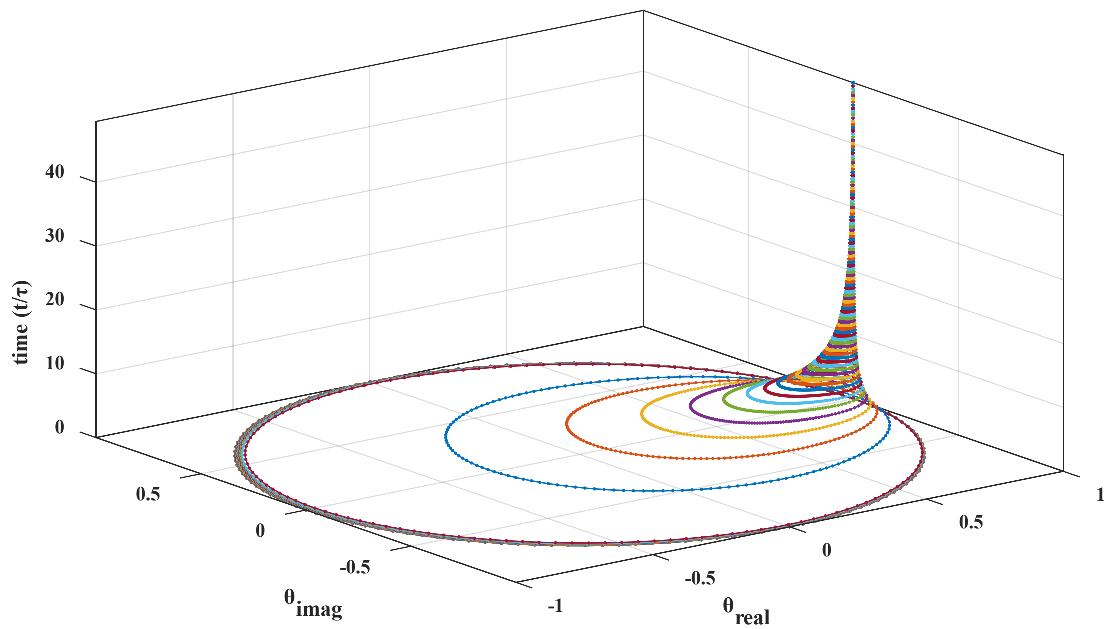

Figure 1: Solution $\boldsymbol{f(t,x)}$ to the transcendental equation (15) for internal neutron population

Figure 2 shows the evolution of $\boldsymbol{P_\nu(t)}$ in time. At each time step, the solution $\boldsymbol{f(t_i, x)}$ is computed which is then inverted using FFT to give $\boldsymbol{P_{t_i}(\nu)}$. Thus, all the internal neutron distributions evolve simultaneously one time-step at a time.

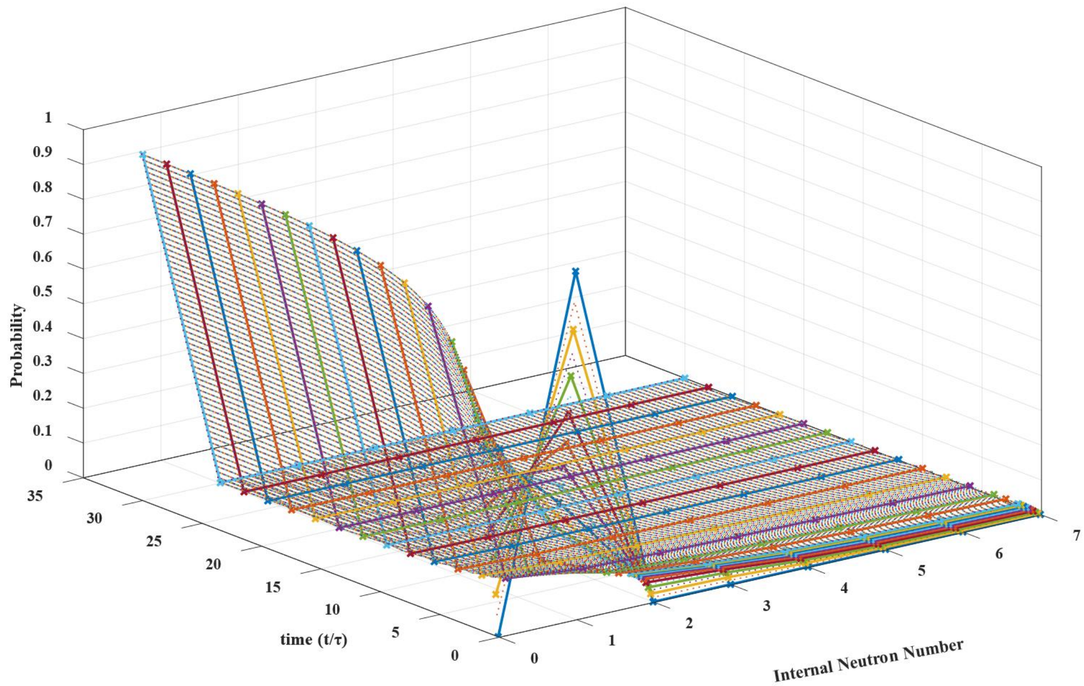

Figure 2: Simultaneous Evolution of Internal Neutron Probabilities $P_\nu(t)$ in Time

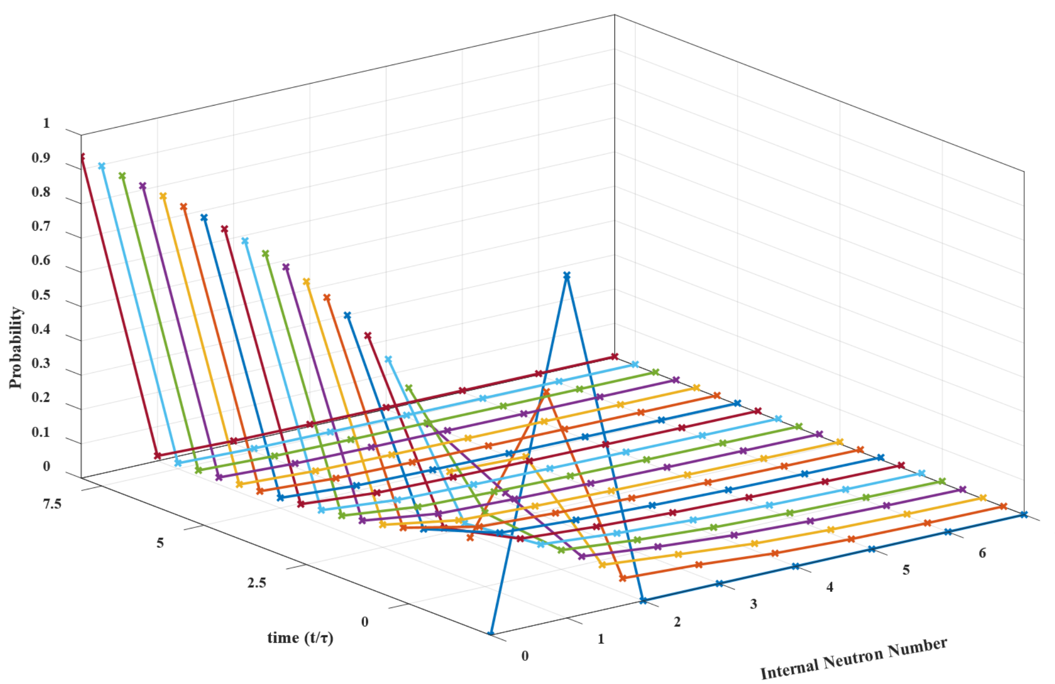

Figure 3: Zoomed-in version of Figure 2

Figure 4 shows the individual internal neutron number distributions obtained after completing the building up of $P_\nu(t)$**.** This plot can be viewed as taking slices out of Figure 2 along the time axis.

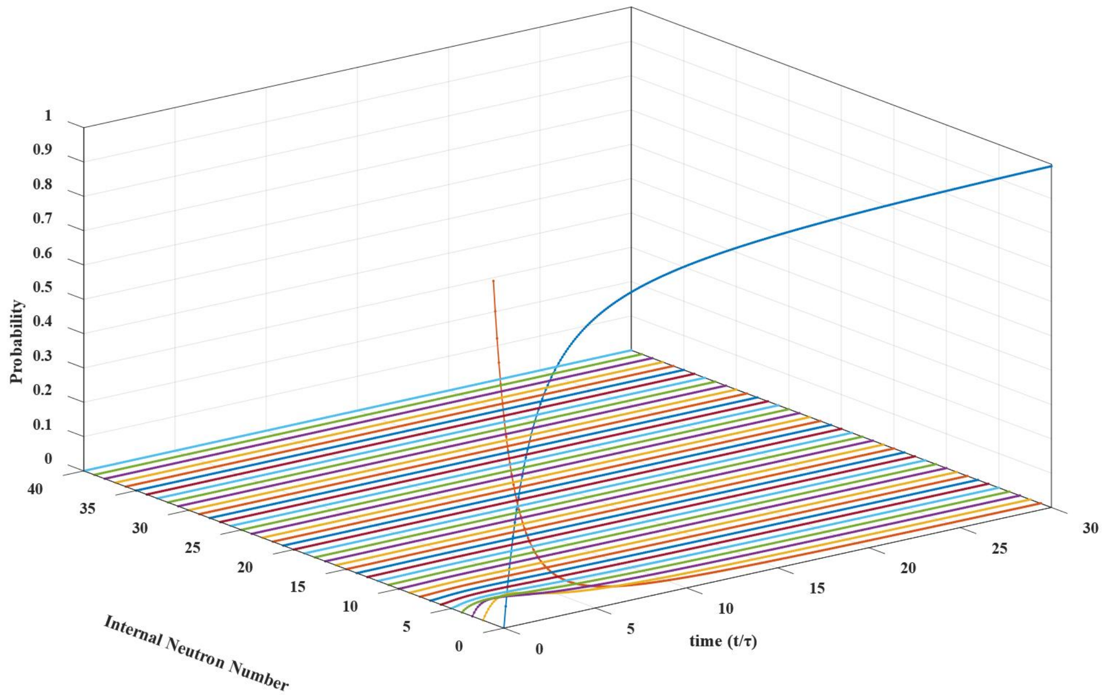

Figure 4: Individual Internal Neutron Number Distributions

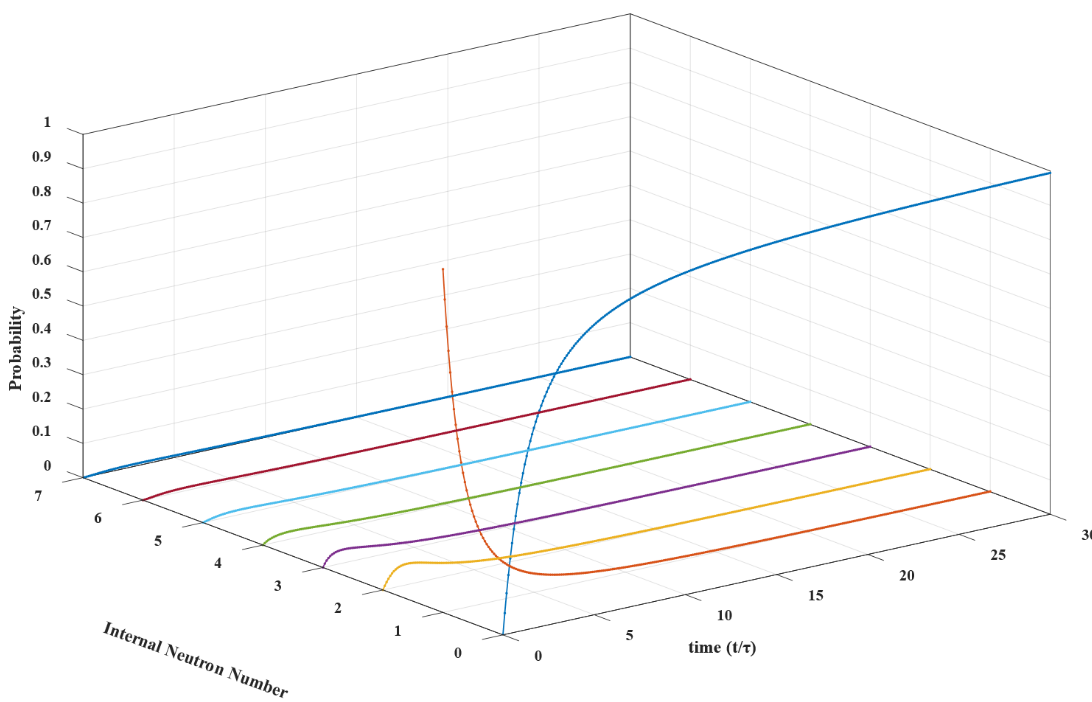

Figure 5: Zoomed-in version of Figure 4

The pseudo-code for computing the probability distribution of internal neutron population is given below:

Algorithm 2: Computing the probability distribution of internal neutron population from the generating function $f(t, x)$

Internal-Neutron-Probability-Distribution-Function $f(t, x)$

in: $n \times m$ array of internal neutron generating function $f\left(t, x = e^{i\theta}\right)$, n values of t, m values of θ

out: $n \times m$ array of internal neutron probability distribution function $P(t, \nu)$, n values of t, m values of ν

1: Generate a uniformly spaced $0 \leq \theta_{uniform} \ll \pi$ array of length N_ν

Generate a uniformly spaced $0 \leq \frac{t}{\tau} \ll 50$ array $t_{uniform}$

2: **for** each $t_i \in \boldsymbol{t}$ **do**

3: Interpolate real part of $f(t_i, \theta)$ over θ and get the interpolant F1

Interpolate imaginary part of $f(t_i, \theta)$ over θ and get the interpolant F2

4: $f\left(t_i, \theta_{uniform}\right) = Complex\left(F1(\theta_{uniform}), F2(\theta_{uniform})\right)$

⊳ $f\left(t_i, \theta_{uniform}\right)$ is over 0 to π and complex-valued

$f\left(t_i, \theta^{2\pi}_{uniform}\right) = [f\left(t_i, \theta_{uniform}\right); flip(f^*\left(t_i, \theta_{uniform}\right))]$

⊳ assemble solution over entire 2π by stacking the solution over π and its reflection along the real-axis; remove coincident points at 0 and π and concatenate

5: $P(t_i, \boldsymbol{v}) = fft(f\left(t_i, \theta^{2\pi}_{uniform}\right)$

Normalize the 2D probability $P(t, v)$ for all t

⊳ Probability for all internal neutrons at time t_i, real-valued

6: **end for**

7: **for** each $k \epsilon N_\nu$ **do**

8: Interpolate $P(t, v_k)$ over $t_{uniform}$ and get the interpolant F

$P\left(t_{uniform}, v_k\right) = F\left(t_{uniform}\right)$

9: **end for**

In Figure 6 through Figure 11, a few internal neutron probability distributions are shown for internal neutrons = {0,1, 5, 10, 20, 40}.

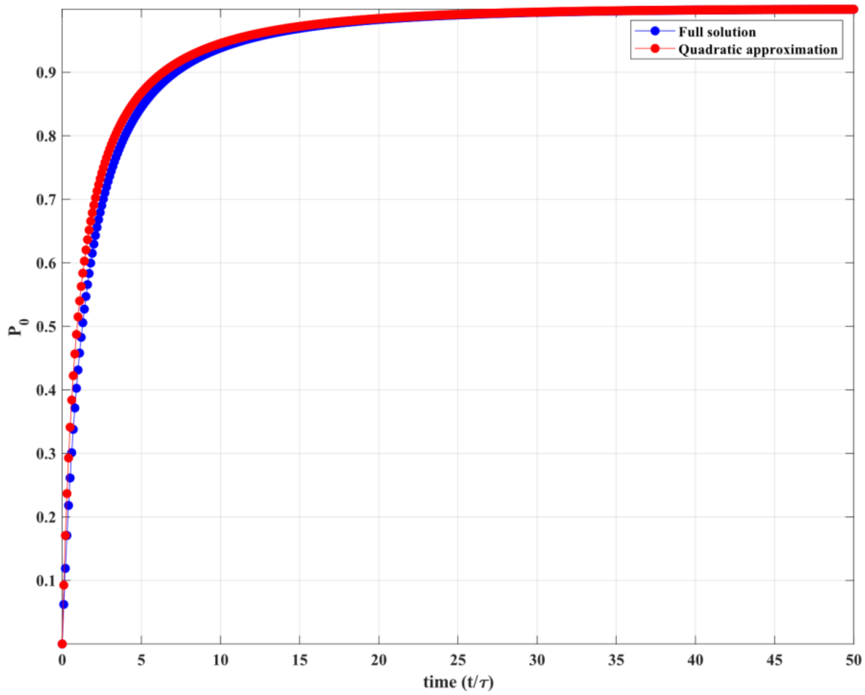

Figure 6: Probability Distribution $P_0(t)$ of 0 internal neutron

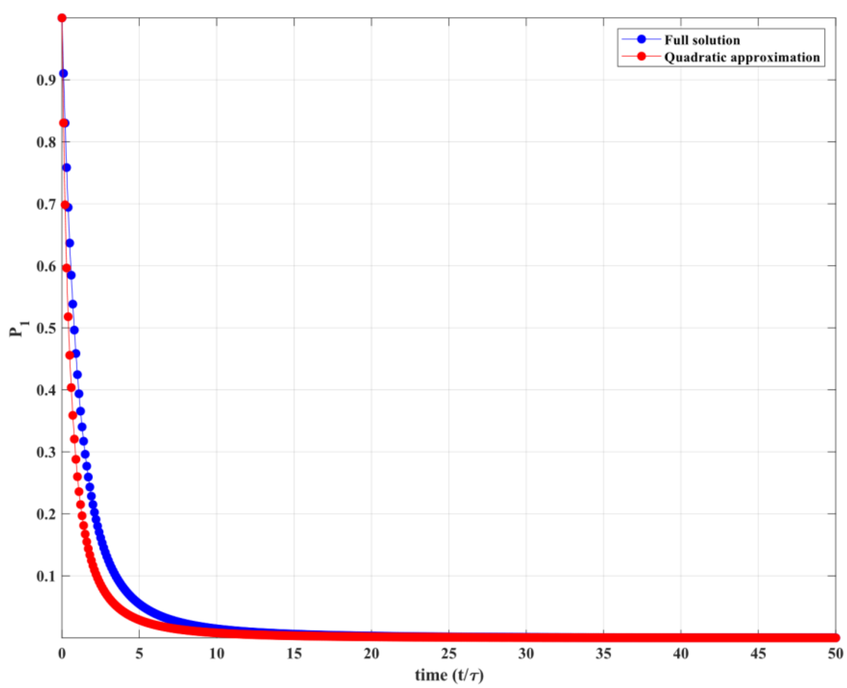

Figure 7: Probability Distribution $P_1(t)$ of 1 internal neutron

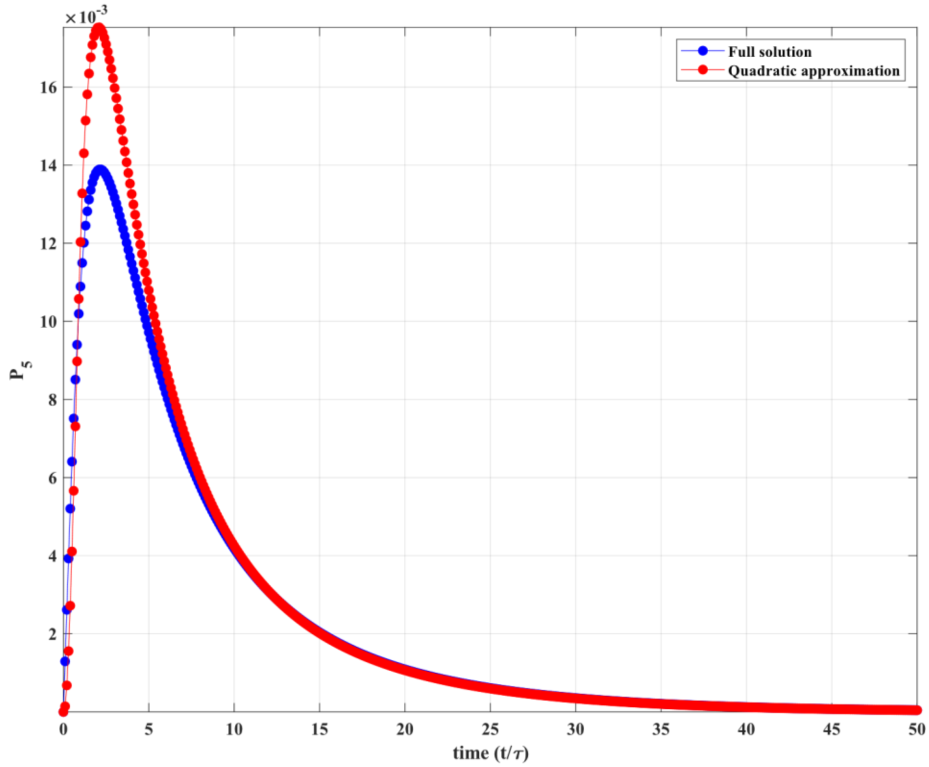

Figure 8: Probability Distribution $P_5(t)$ of 5 internal neutrons

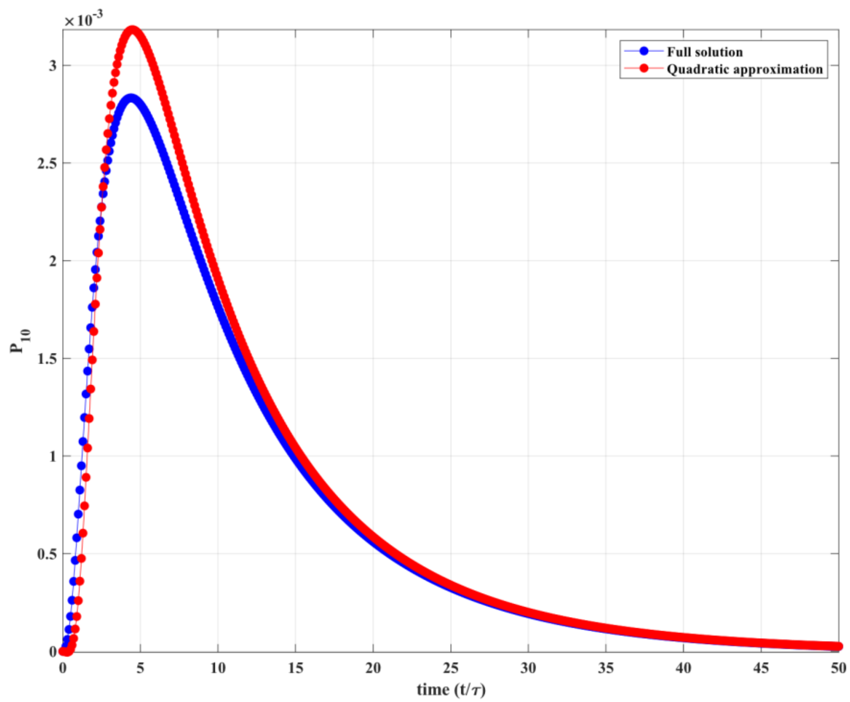

Figure 9: Probability Distribution $P_{10}(t)$ of 10 internal neutrons

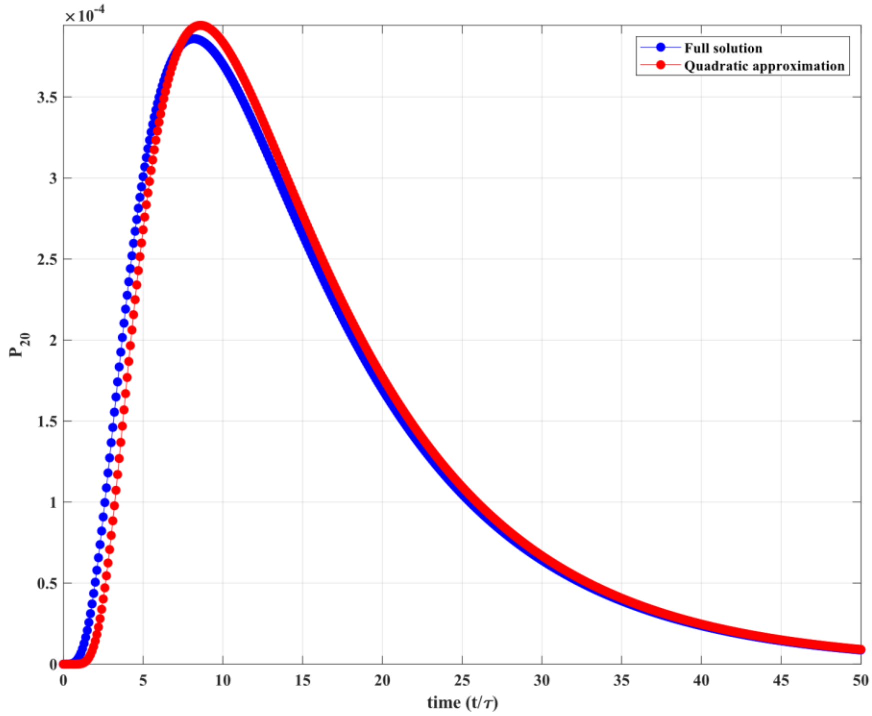

Figure 10: Probability Distribution $P_{20}(t)$ of 20 internal neutrons

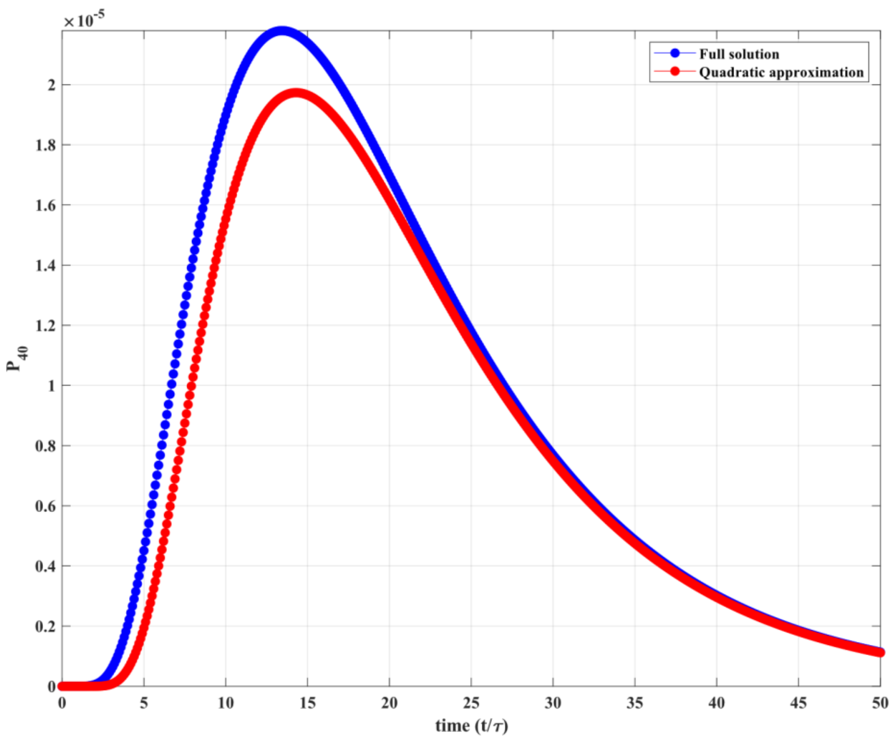

Figure 11: Probability Distribution $P_{40}(t)$ of 40 internal neutrons

External Neutron Population Distributions

The numerical solution for the external distribution starts with the analytic approximation,

$$f(t,y) = 1 - 2qM(1-y)\frac{1-e^{-\alpha' t}}{1-e^{-\alpha' t}+\sqrt{1+4qM(M-1)D_2(1-y)}\left(1+e^{-\alpha' t}\right)} \quad (25)$$

where

$$\alpha' = \alpha\sqrt{1+4qM(M-1)D_2(1-y)}$$

The roots $u_1, \cdots, u_7$ of the equation (13) are functions of y, the generating function variable for the external population and they trace the contours in the complex plane as shown in Figure 12.

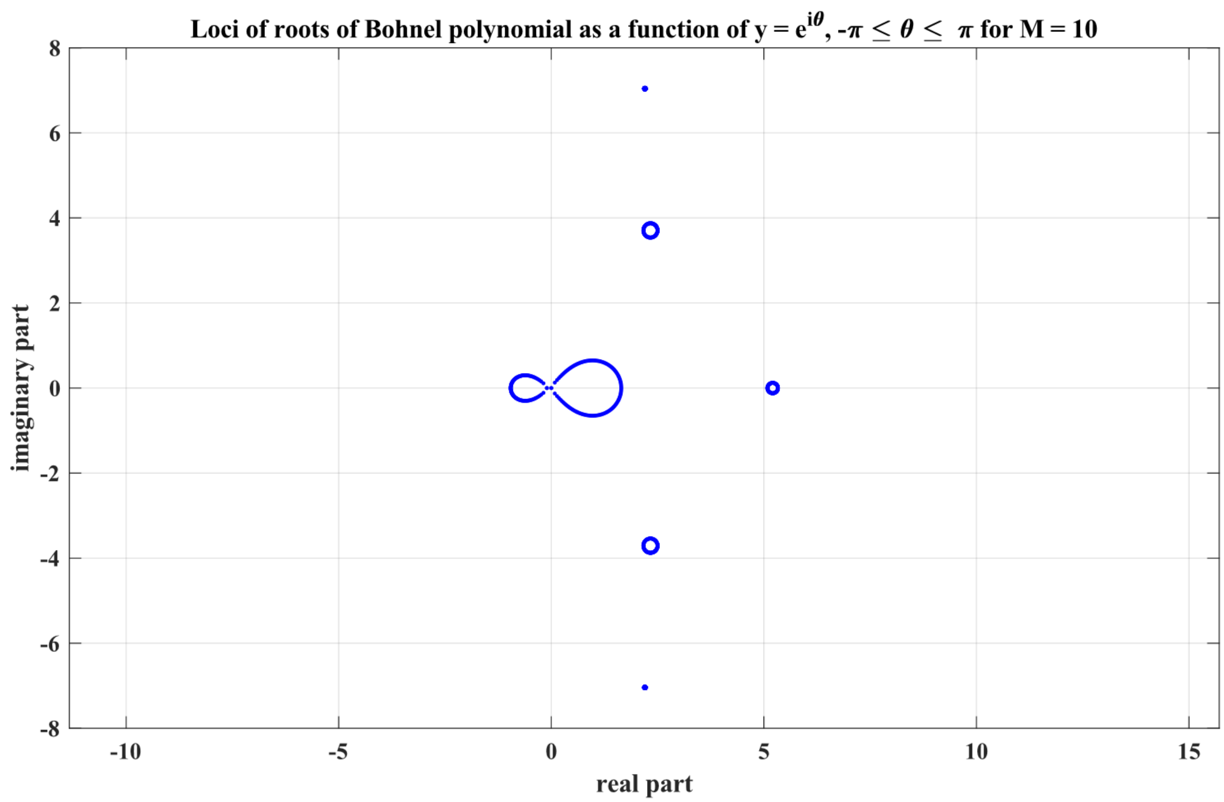

Figure 12: Loci of roots of the polynomial equation (13)

The root corresponding to the $t \to \infty$ Böhnel fission chain is u_1, $h(y) = 1 - u_1(y)$. It is the root in Figure 12 that is real and positive for $y = 0$, and goes to zero for $y = 1$.

Figure 13 shows the plot of the solution to the transcendental equation $f\left(t, x=1, y= e^{i\theta}\right)$.

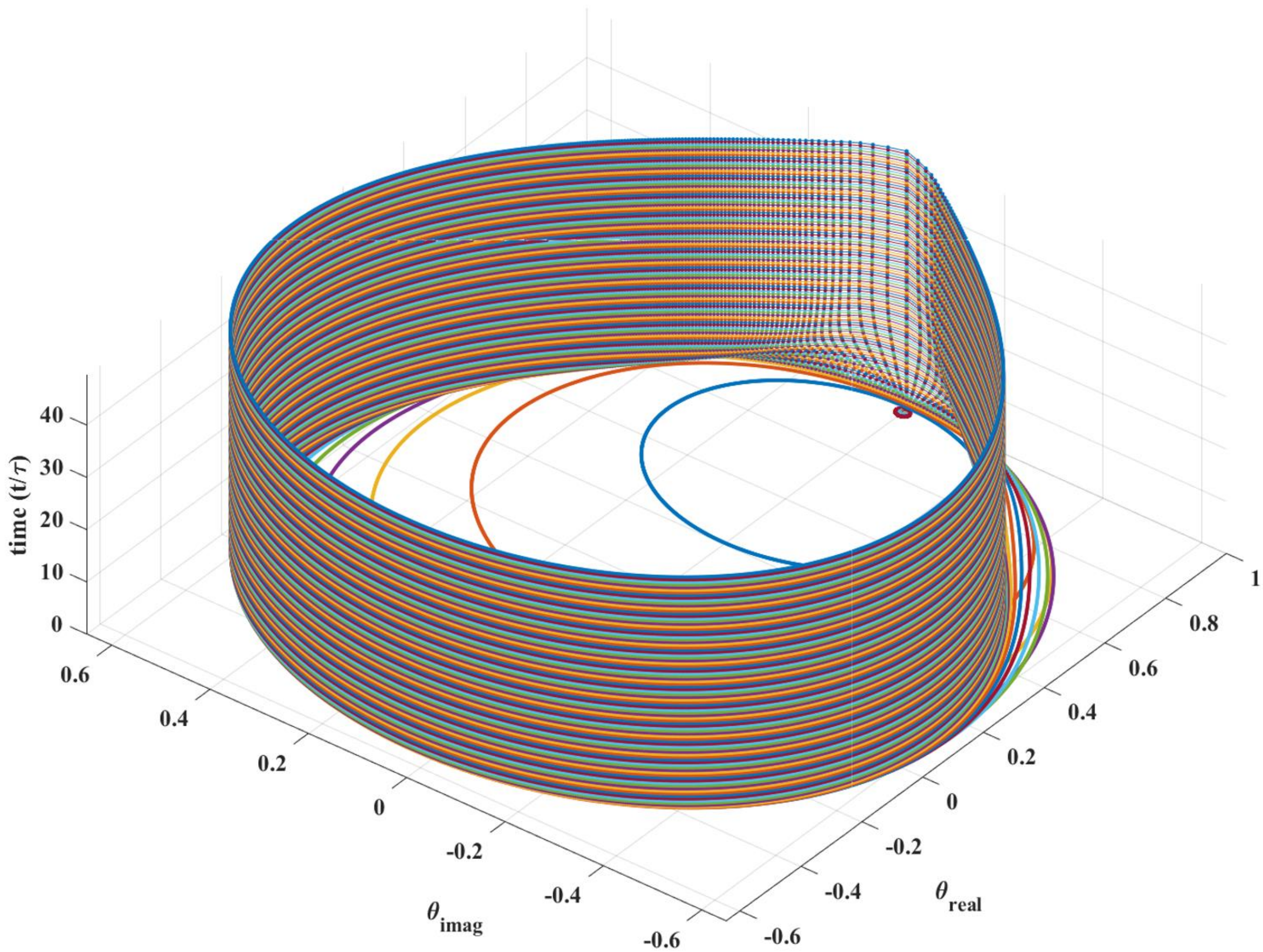

Figure 13: Solution to the transcendental equation $f\left(t, x = 1, y = e^{i\theta}\right)$.

As $t \to \infty$, $f(t, y) \to h(y)$.

Algorithm 3: Computing the solution $\boldsymbol{f\left(t, x = 1, y = e^{i\theta}\right)}$ to the transcendental equation (18) for external neutron population

---

External-Neutron-Population-Generating-Function (*Multiplier, M*)
in: HEU or Pu nuclear data, multiplication of the system
out: $n \times m$ array of external neutron generating function $f\left(t, x = 1, y = e^{i\theta}\right)$, n values of t, m values of θ

1: Get nuclear data and compute C_i, ν_i, and D_i for $i = 1, 2, .., 7$ in equation (9) and equation (11)

2: Choose $0 \leq \frac{\boldsymbol{t}}{\boldsymbol{\tau}} \leq 50$ (τ = 1 shake) array and $0 \leq \theta \leq \pi$ array (non-linearly spaced arrays) and $\theta_{critical} = \frac{\pi}{6}$

3: **for** each $t_i \in t$ **do**

4: **for** each $\theta_j \in \theta$ **do**

5: **if** j =1 **then** Compute the roots of the polynomial in equation (13) **end if**

▻ roots of the polynomial are functions of $y = e^{i\theta_j}$ but not of t

6: Compute approximate solution to external population generating function using equation (22)

$$f(t_i, \theta_j)_{approximate} = 1 - \frac{2qM\left(1 - e^{i\theta_j}\right)\left(1 - e^{-\alpha' t_i}\right)}{1 - e^{-\alpha' t_i} + \sqrt{1 + 4R_{2F}\left(1 - e^{i\theta_j}\right)}\left(1 + e^{-\alpha' t_i}\right)}$$

where $\alpha' = \alpha\sqrt{1 + 4R_{2F}(1 - y)}$, and where $R_{2F} = qM\frac{M-1}{\bar{\nu}}\nu_2$.

7: **if** $\theta_j \leq \theta_{critical})$ **then**

initial solution = $f(t_i, \theta_j)_{approximate}$

else

initial solution = $f(t_i, \theta_{j-1}) + \frac{f(t_i, \theta_{j-3}) - f(t_i, \theta_{j-2})}{2}$

end

8: compute exact solution $f(t_i, \theta_j)$ to the transcendental equation (18) using the initial solution from step 7

9: **end for**

10: **end for**

11: **return** $f(t, \theta)$ ▻ complex-valued

The algorithm for computing the probability distribution for each leaked neutron (external neutron) from the generating function $f(t, x = 1, y = e^{i\theta})$ is identical to computing internal neutron probability distribution. In Figure 14 through Figure 19, a few external neutron probability distributions are shown for external neutrons = {0,1, 5, 10, 20, 40}.

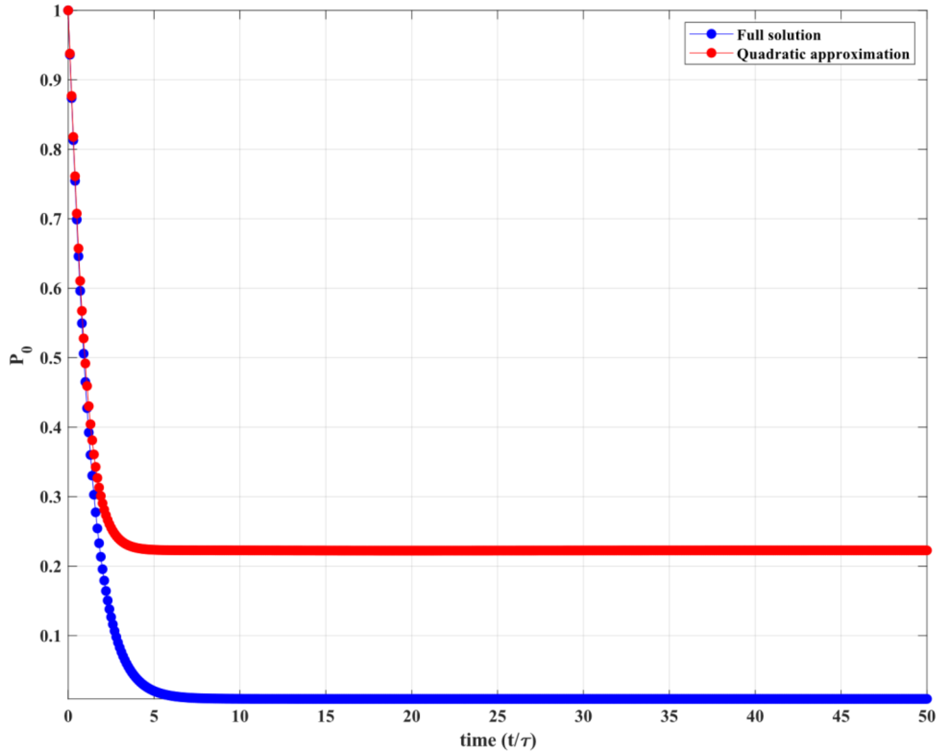

Figure 14: Probability Distribution $P_0(t)$ of 0 external neutron

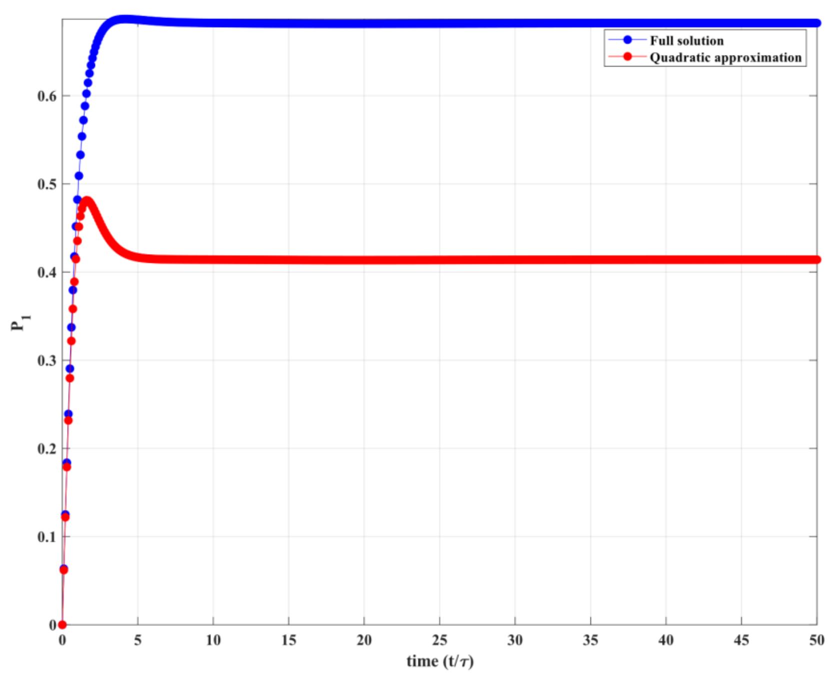

Figure 15: Probability Distribution $P_1(t)$ of 1 external neutron

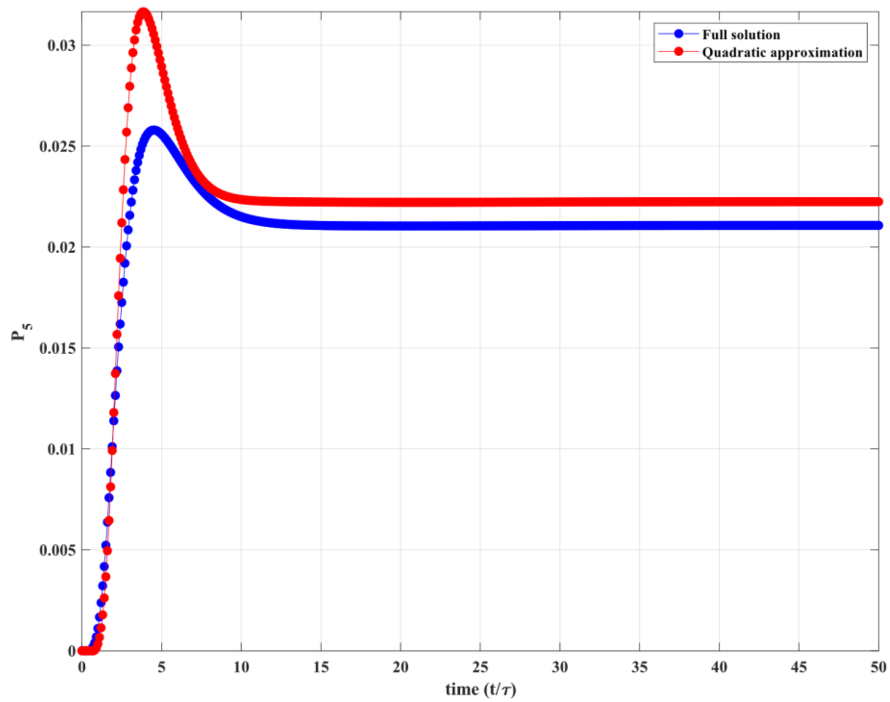

Figure 16: Probability Distribution $P_5(t)$ of 5 external neutrons

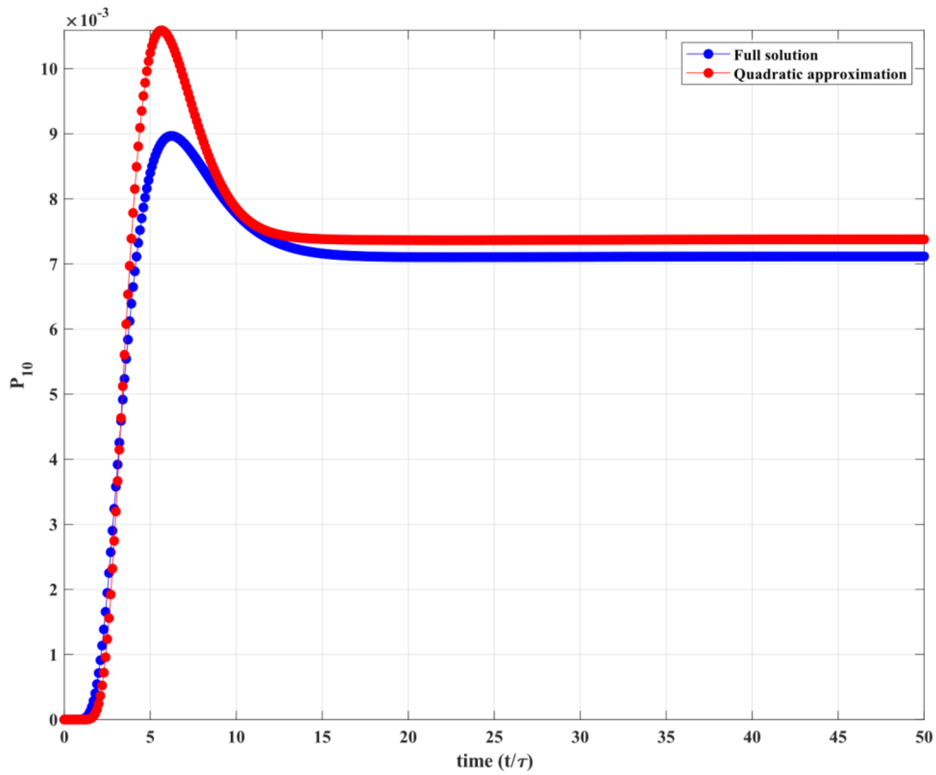

Figure 17: Probability Distribution $P_{10}(t)$ of 10 external neutrons

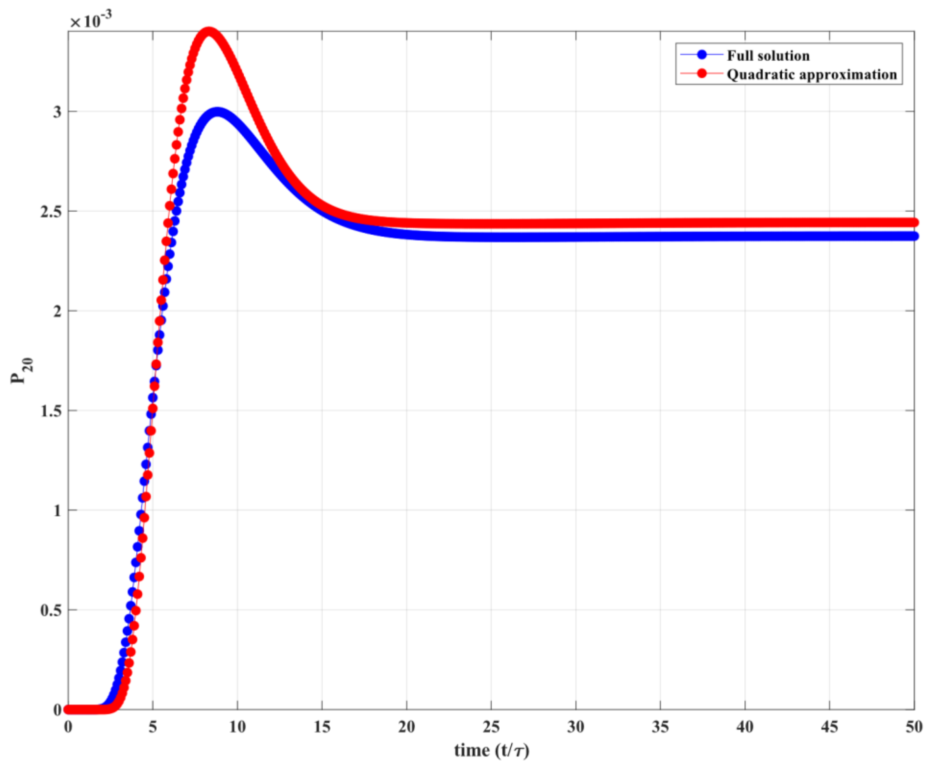

Figure 18: Probability Distribution $P_{20}(t)$ of 20 external neutrons

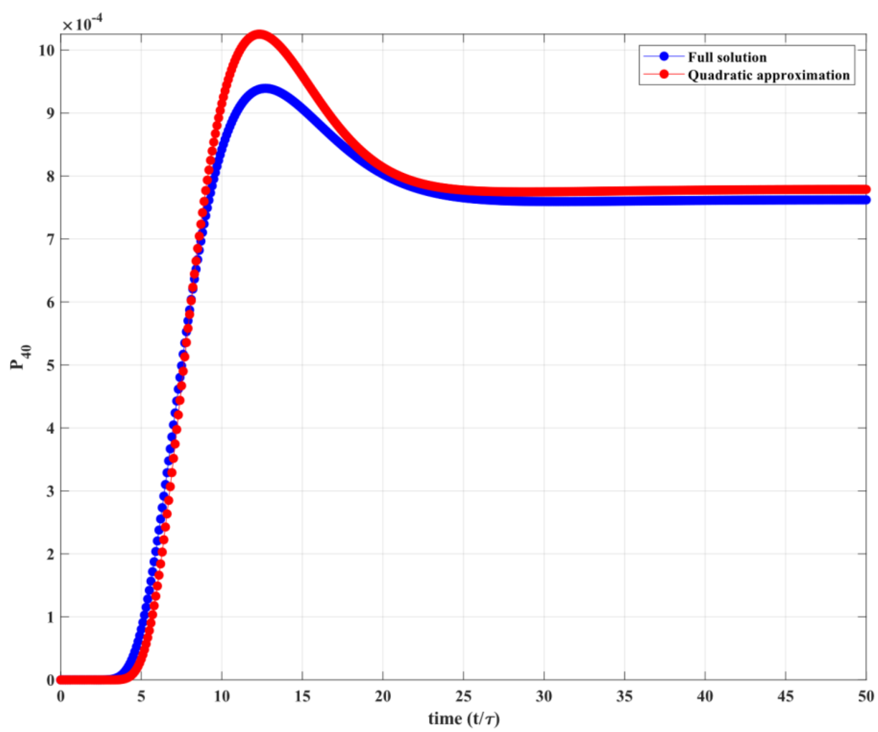

Figure 19: Probability Distribution $P_{40}(t)$ of 40 external neutrons

Conclusions

The populations computed here are foundational for different neutron counting distributions, both for fast neutron counting, and for thermal neutron counting from a limit of a reactor theory (see below). One motivation for pursuing new ways to solve the Feynman equation is that fast computing methods are needed to compare to counting data in real time. Solving differential equations [5] is time expensive. This method can be parallelized, numerically solving the generating function transcendental equation for each time on a separate processor. For computing time interval distributions, besides the internal distribution, only the external population $P_0(t)$ and it's first two time-derivatives are needed, so there is no need for the numerical FFT. It is possible that this method can be fast enough for real time analysis of HEU problems (where the rate of initiation of fission chains is low).

Another motivation for this form of solution is historical, completing a path pursued by Feynman in his original work on fission chains, and implemented by him only in quadratic polynomial approximation for the internal population, and here solved completely.

The same equation (1) can also describe a limit of a thermal reactor [7], where all induced fission is from neutrons that have thermalized and diffuse in a moderator. The translation of the parameters in the equation can be read off from,

$$\frac{\partial f(t,w,v)}{\partial t} = \lambda[-w + q_{th}v + p_{th}C^{th}(w)]\frac{\partial f(t,w,v)}{\partial w}$$

where the generating function variable w tracks internal thermal neutrons, v tracks thermal neutrons lost from the multiplying medium, $C^{th}(w)$ is the thermal neutron induced fission neutron number generating function, λ is the moderator diffusion constant, p_{th} is the probability a thermal neutron induces fission, and $q_{th} = 1 - p_{th}$ is the probability a thermal neutron is lost from the multiplying system. This reactor is driven completely by thermal neutrons, the multiplication of the HEU is approximately $M_0 = 1$. Fast neutrons are created by induced fission but thermalize with probability $s = 1$. A slight generalization of this reactor problem, for $s \neq 1$ and including an additional fast leaked fast neutron population, can be also be solved from this work by a dictionary.

Acknowledgements

This work was performed under the auspices of the U.S. Department of Energy (DOE) by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344. This work was developed under DOE funding. We thank Bart Ebbinghaus and Sean Walston for their support and encouragement.

References

[1] R. P. Feynman, "Statistical Behavior of Neutron Chains," LA-591 Los Alamos (1946).

[2] M. K. Prasad and N. J. Snyderman, "Statistical Theory of Fission Chains and Generalized Poisson Counting Distributions," Nuclear Science and Engineering: **172**, 300 (2012).

[3] D. H. Chambers, H. Chandrasekaran, and S. E. Walston, “Fourier Method for Calculating Fission Chain Neutron Multiplicity Distributions,” Nuclear Science and Engineering: **184** 244 (2016).

[4] J. K. Cavers, "On the Fast Fourier Transform Inversion of Probability Generating Functions," Journal of the Institute of Mathematics and its Applications, **22**, 275 (1978).

[5] K. S. Kim, L. F. Nakae, M. K. Prasad, N. J. Snyderman, and J. M. Verbeke, "Time Evolving Fission Chain Theory and Fast Neutron and Gamma-Ray Counting Distributions,” Nuclear Science and Engineering: **181**, 1 (2015).

[6] E.B. Saff and A.D. Snider, *Fundamentals of Complex Analysis with Applications to Engineering and Science*, Pearson Education Inc. (2003).

[7] K. S. Kim, L. F. Nakae, M. K. Prasad, N. J. Snyderman, and J. M. Verbeke, “Fission Chain Restart Theory,” Nuclear Science and Engineering: **188**, 57 (2017).