Rohan Raj Shrivastav

rohraj812@gmail.com

PROJECT REPORT

Numerical solution of ODEs governing nuclear reactor kinetics.

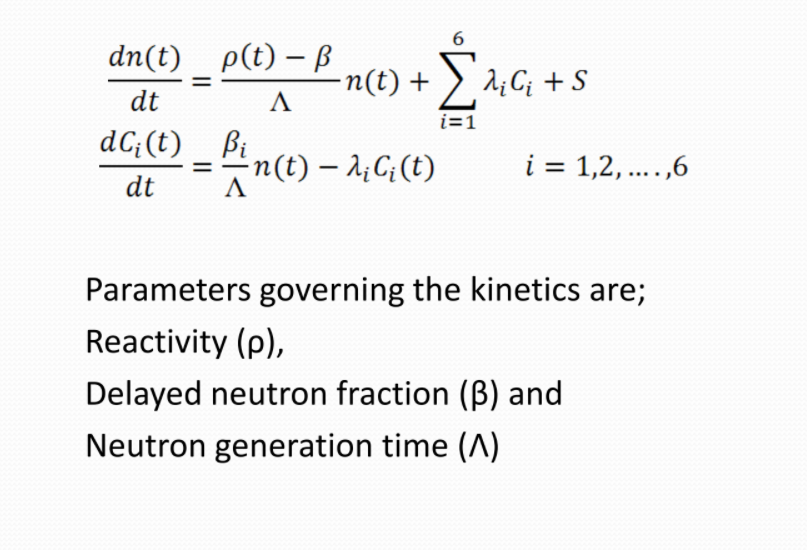
AIM

Numerical solution of ODEs governing nuclear reactor kinetics.

OBJECTIVES

* Mathematical formulation of reactor neutron kinetics phenomenon
* Developing a Matlab programme to solve certain reactor kinetics problems.
* Problem formulation and subsequent computational work.

WORKING WITH KINETIC EQUATIONS (One group Model)



This is more useful form of point reactor kinetics

equations (PRK)

Using the values of parameters like :

1. Reactivity (ρ),
2. Delayed neutron fraction (β)
3. Neutron generation time (Λ)
4. n(t) denote the neutron population
5. λ denotes decay constant

Process which will be carried out :

* n(t) i.e. neutron population value wil be calculated as different time instances. Which will be our Analytical Solution (Exact).
* We will then obtain the results for single delayed group using Numerical solution of ODEs :

1. Runge-Kutta
2. Implicit-Euler
3. Crank-Nicolson

And then compare those results with Analytical one

* Later on we will compare the results of our Analytical solutions with the MATLAB computational work where we will code the required the Kinetic Equation given on the previous page.
* For comparision of data of both Analytical Solution and the MATLAB computational work we will seek the help of log log plot (in the form of graph) which will make it easier for us to compare.
* All these analysis was done considering only a single group of delayed neutrons.

Characteristics of PRK equations

(Why Point Reactor Kinetics Equations are used ? )

* PRK equations constitute a system of linear ordinary

coupled differential equations with variable coefficients

* In general, no exact analytical solutions are possible

and approximate methods have to be used

* An exception is the case with a step-change of

reactivity, for which an exact analytical solution can be

found.

* [In most of the reactor dynamics problems we are

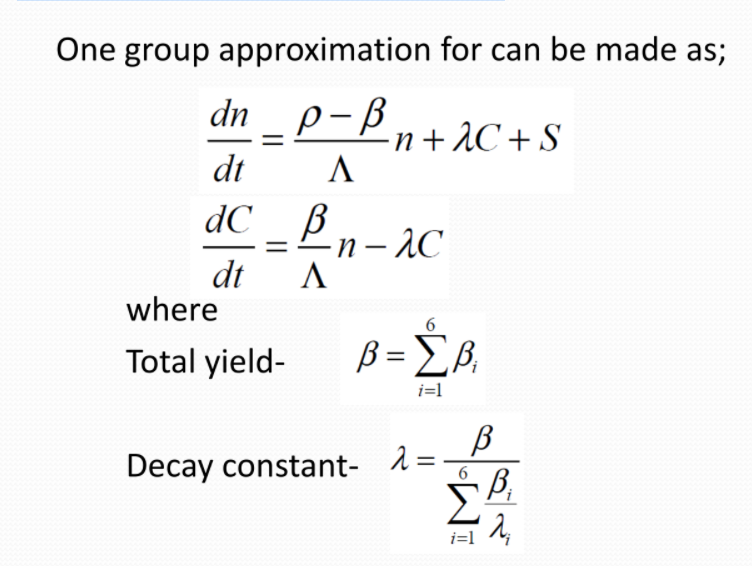
interested in the reactor power P(t) than reactor neutron

population. Since neutron concentration and neutron

power level are proportional, we can choose to work with

P(t) in place of n(t)]

One group model



Kinetics solution implications

Λ, changes with k (can be ignored)

ρ, changes with evolving core state (feedbacks required)

β, changes due to change in contribution of fissioning isotope (space, size and burnup effect)

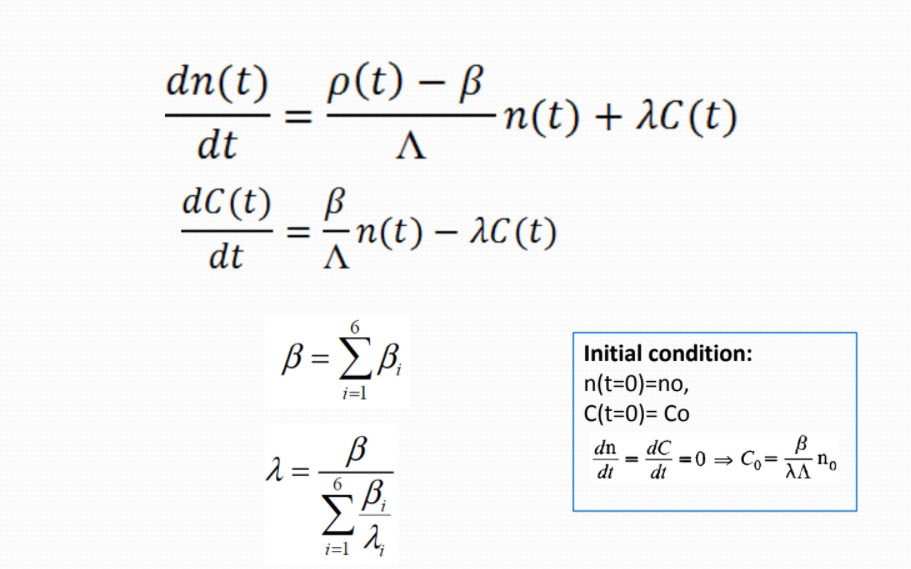
* Considering these aspects, one requires numerical

solution however analytical solution with certain

approximation is possible for few cases.

* Analytical solution will be studied first for one group

One Group Approximation ( Differential equation in two variable format)



* This is the Differential equation form of the Kinetic Equation we will code this equation into the MATLAB program

Kinetics Governing Parameters (DATA)

Reactivity (ρ) = 0.0065

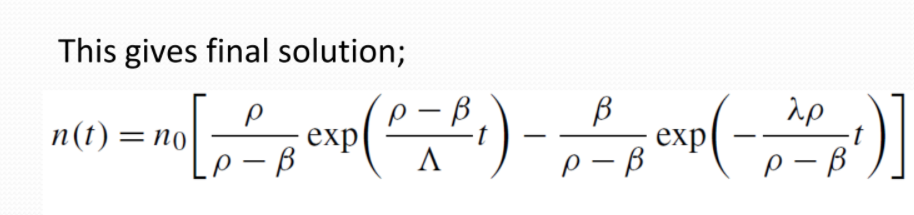
Delayed neutron fraction (β) = 0.0065

Neutron generation time (Λ) = 0.001

Decay constant (λ) = 0.076

On further simplification of the ;

* One Group Approximation ( Differential equation in two variable format) on page 5 mentioned, we finally get the formula to which will be used to generate Analytical Solution (Exact).



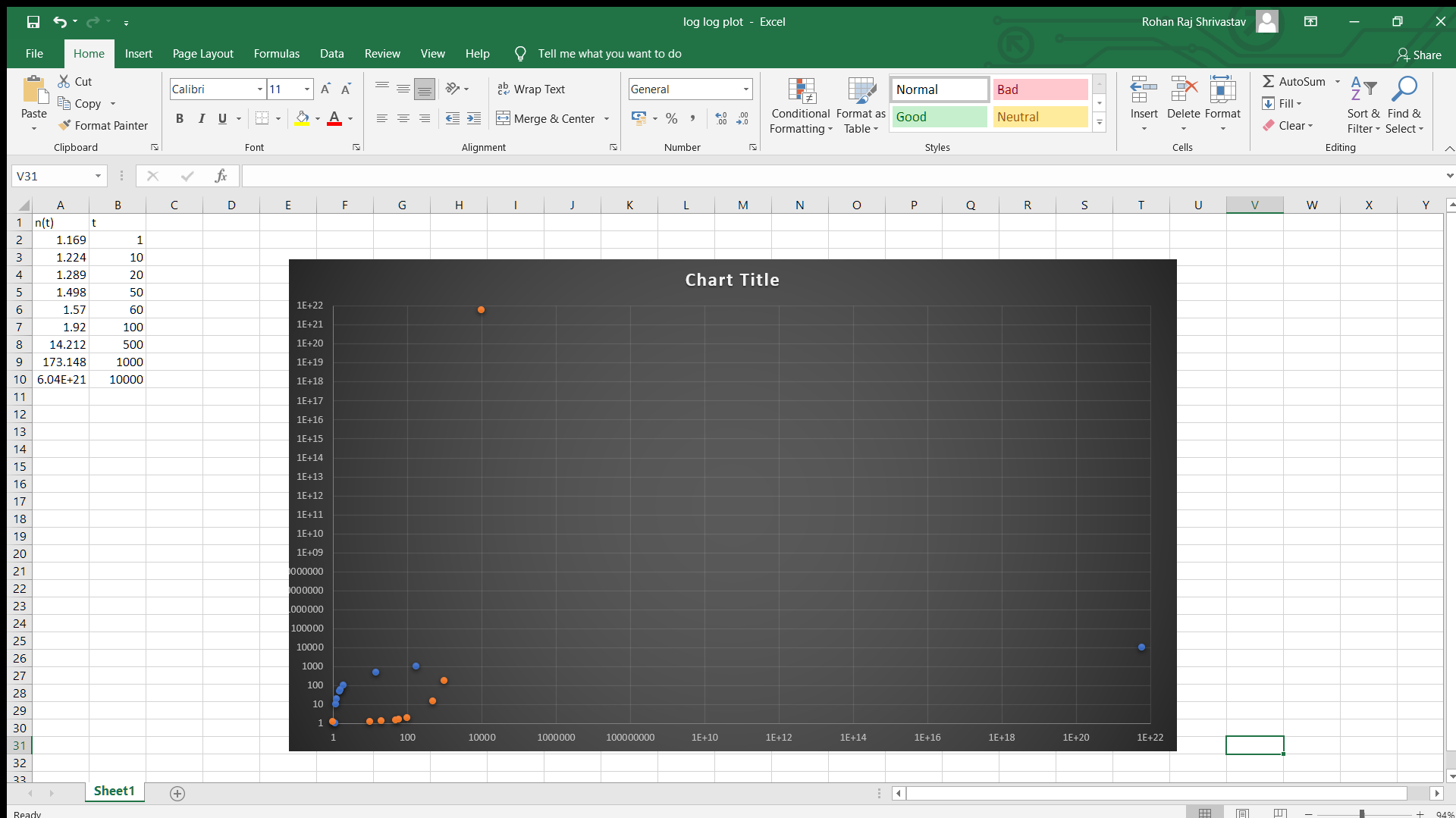
* This will be used to calculate the value of n(t) i.e. the Neutron Population at various time instances.
* All the Kinetics Governing Parameters (DATA) are the same as mentioned above.
* **Analytical Solution (Exact).**

The values of n(t) at different instances of time(t) are as follows :

|  |  |
| --- | --- |
|  |  |



Log log plot of the results :



* Here n(t) i.e. Neutron Population which is on the Y – axis
* Time (t) is on the X – axis

Numerical solution of ODEs

Results obtained for single delayed group :

|  |  |  |  |
| --- | --- | --- | --- |
| Time(s) | Runge-Kutta | Implicit-Euler | Crank-Nicolson |
| 1 | 8.0382 | 8.0399 | 8.0382 |
| 2 | 18.5450 | 18.5532 | 18.5450 |
| 5 | 143.5332 | 143.6924 | 143.5332 |
| 10 | 4.0105×10^3 | 4.0194×10^3 | 4.0105×10^3 |
| 20 | 3.1265×106 | 3.1404×10^6 | 3.1266×10^6 |

* Here the Runge-Kutta method and Crank-Nicolson

method give better results than the Implicit-Euler method as expected since Euler method is a

first order method.

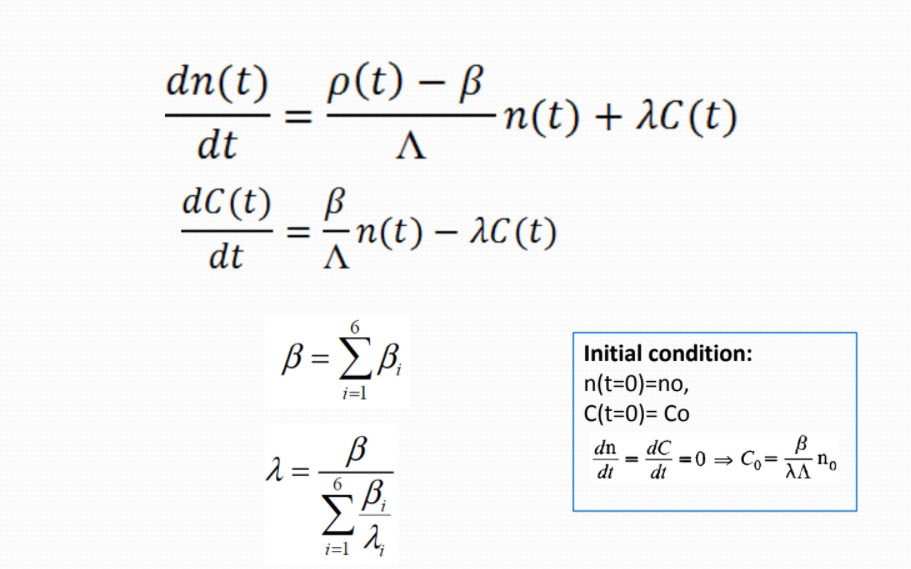
The error in the numerical methods can be attributed to truncation errors/ numerical

approximations which are unavoidable if ODEs are solved via numerical methods. These errors

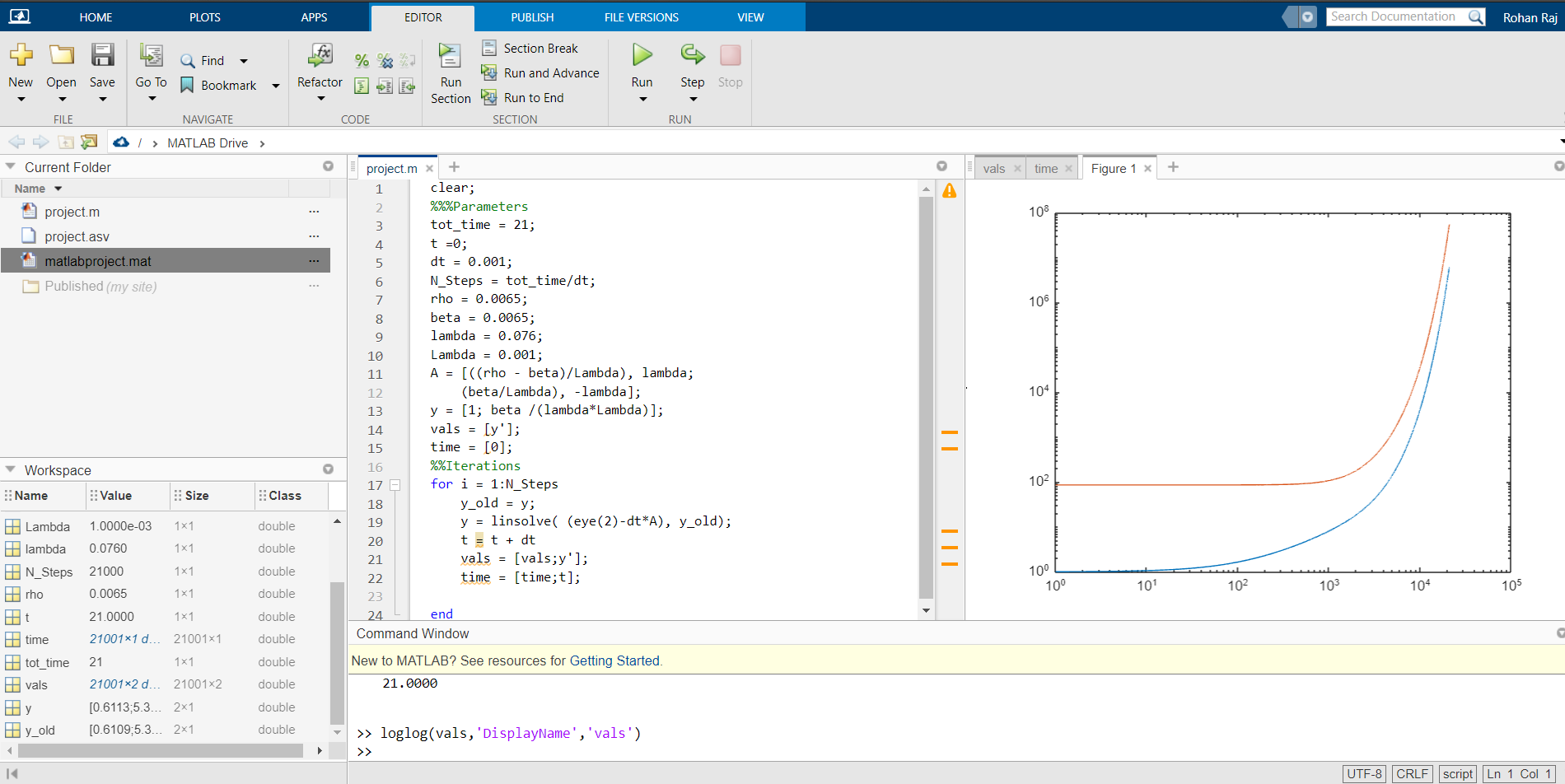
can be reduced slightly by using a method of higher order, but complete removal of these is

not possible.

MATLAB computational work

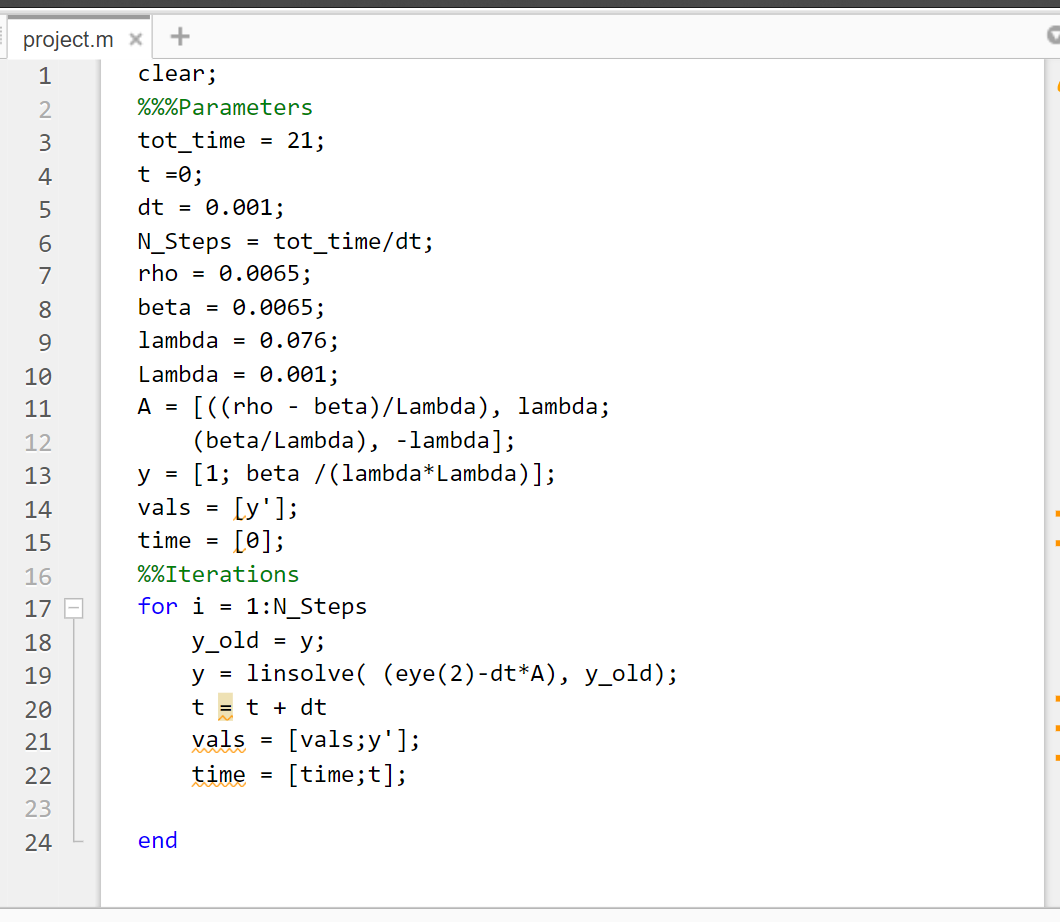


* Programming the above differentia equation into matlab :

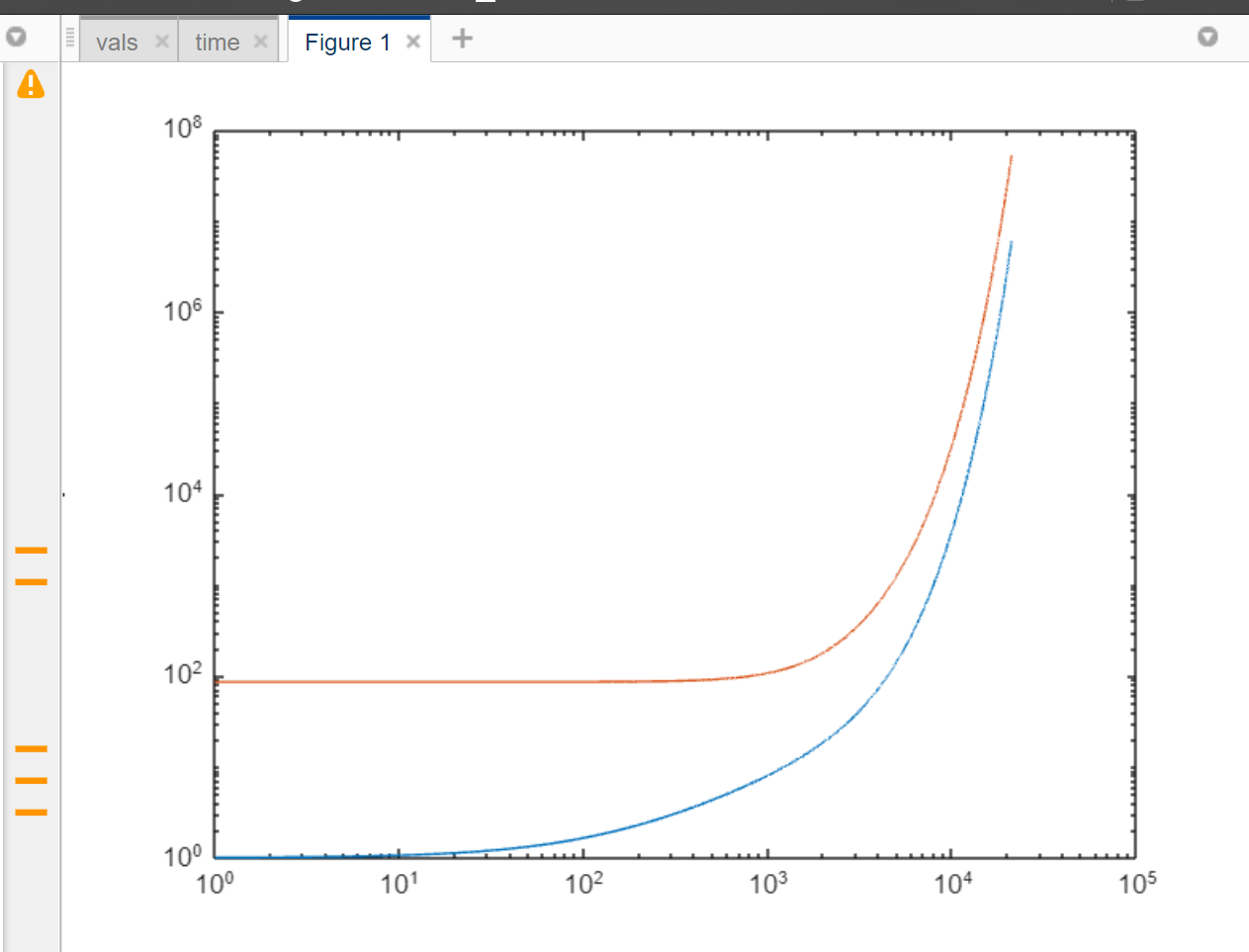


* Here it is a graph of log log plot

MATLAB CODE :



LOG LOG PLOT (GRAPH) :



* From this log log plot in MATLAB we can easily compare it with the log log plot of the analytical solution obtained

So hence by this we can easily compare our results of :

1. **Analytical Solution (Exact).**
2. Numerical solution of ODEs
3. MATLAB computational work

Conclusions :

1. All the objectives were attained
2. Since the system

of differential equations considered here are stiff, implicit methods tend to almost always work

better than explicit methods and hence we arrive at our choice of the Crank-Nicolson method

which has been used for all the further study.

1. From this Project I got the understanding of the basics of nuclear reactor engineering, reactor physics fundamentals, kinetics and ordinary differential  
   equations.
2. I also understood mathematical formulation of reactor neutron kinetics phenomenon, approximate solutions,
3. And problem formulation and subsequent computational work.