# ENGR30003: Numerical Programming for Engineers

Semester 2, 2017 / Assignment 2

Justin Bugeja (758397)

## Question 2.1: Analytical Solution

2.1 Analytical Solution for $\theta = 0°$

$$\tan(0) = 0 = 2\cot(\beta)\frac{M^2\sin^2(\beta) - 1}{M^2(\gamma + \cos(2\beta) + 2}$$

(sol. 1) $\Rightarrow$

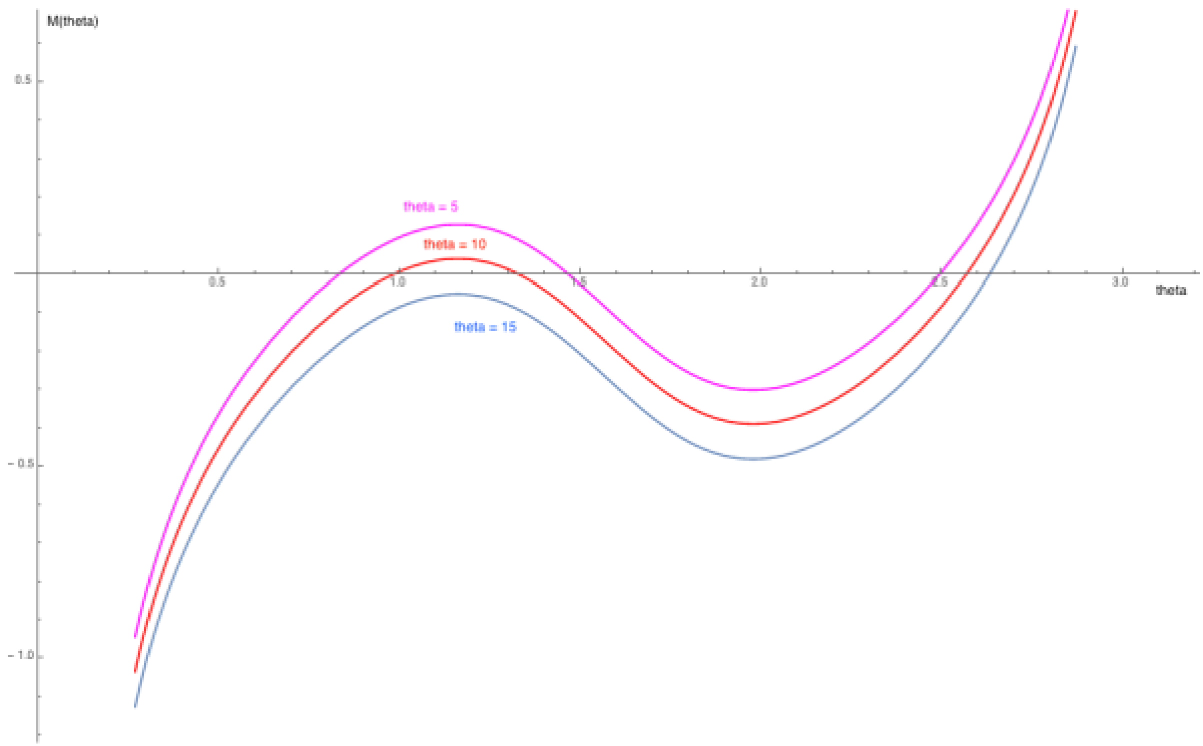
$$0 = M^2\sin^2(\beta) - 1$$
$$1 = M^2\sin^2(\beta)$$
$$1/M^2 = \sin^2(\beta)$$
$$1/M = \sin(\beta)$$
$$\therefore \beta_L = \arcsin\frac{1}{M}$$

(sol. 2) $\Rightarrow$

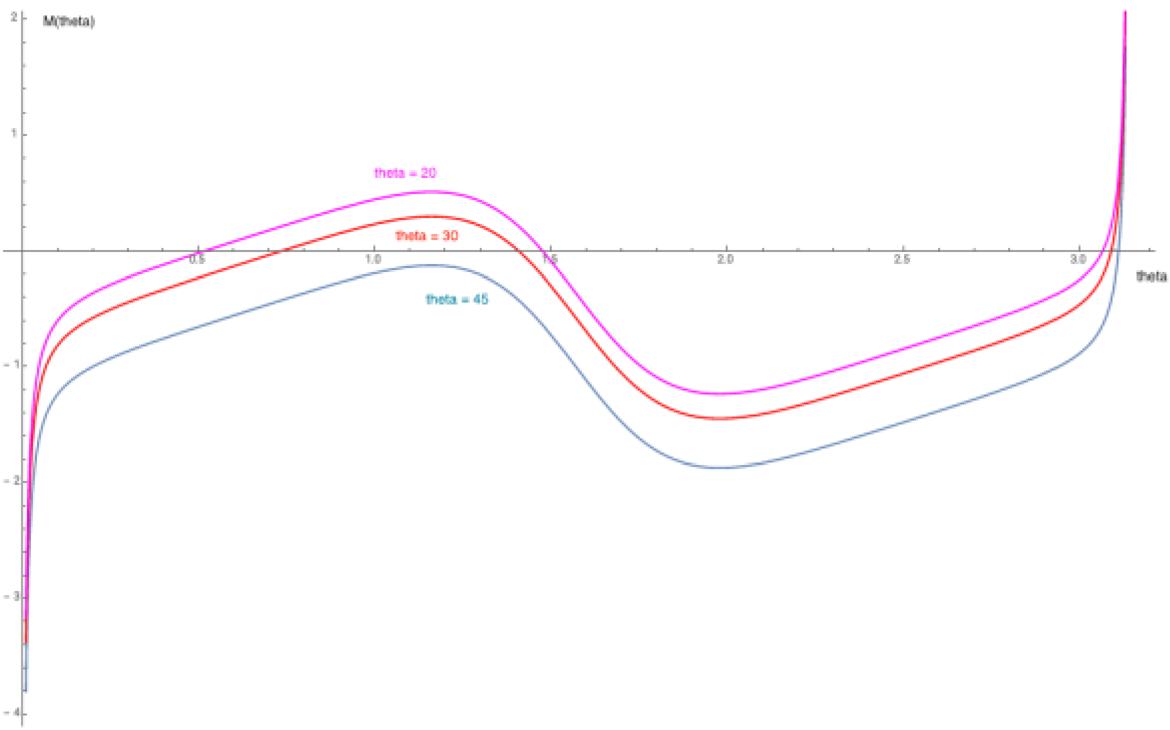
$$0 = 2\cot(\beta)$$
$$0 = \cot(\beta)$$
$$0 = \frac{\cos(\beta)}{\sin(\beta)}$$
$$0 = \cos\beta$$
$$\cos^{-1}(0) = \beta = 90°$$
$$\therefore \beta_u = 90°$$

## Question 2.2: Graphical Solution

a. M = 1.5 and theta = 5, 10 and 15




b. M = 5.0 and theta = 20, 30 and 45

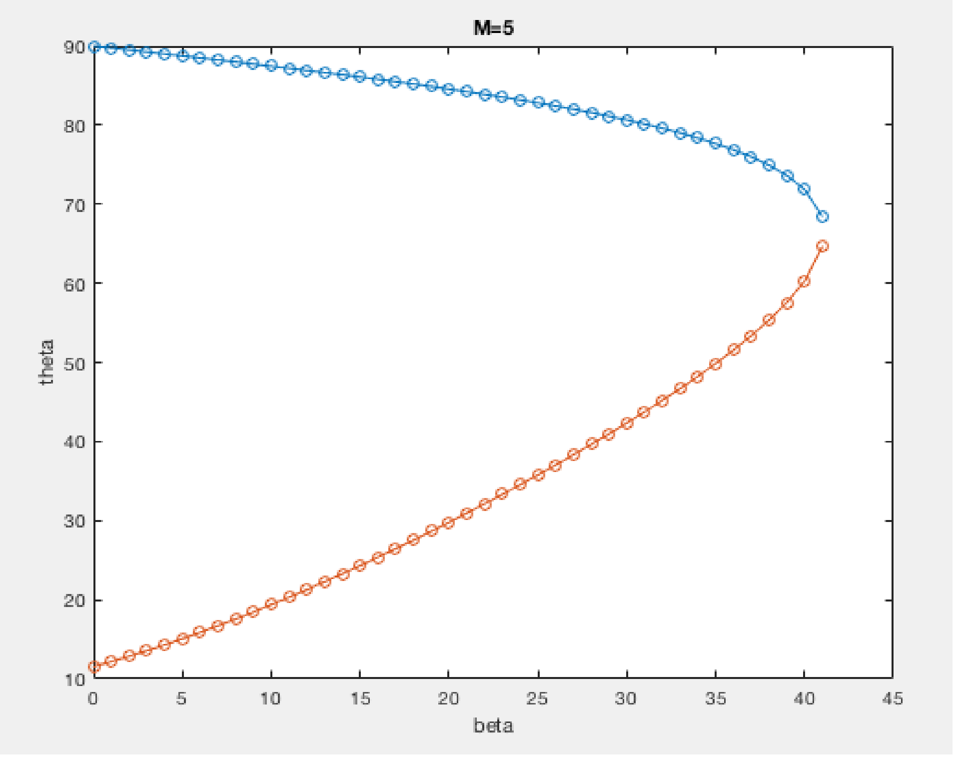


## Question 2.3: C program to solve shock-wave equation

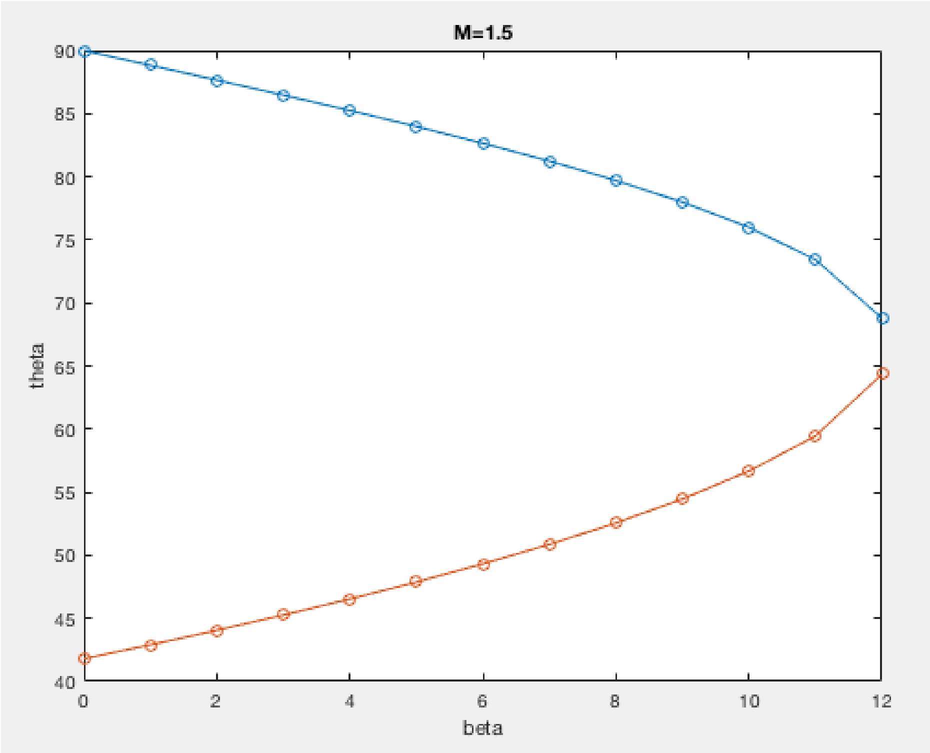
a.

- Initial guess b = 0 doesn't work in Newton Raphson due to the naïve implementation of the derivative evaluation at 0 returns undef, making the program think 0 is a root.
- Guess of $b_{lower} > 54$ doesn't always give a solution
- Guess of $b_{upper} < 77$ doesn't always give a solution.
- The guesses must be within the ranges of [1 54] and [77 90] for $b_{lower}$ and $b_{upper}$ respectively for a solution to be found. This is because the range of [55 77] means that the solution can converge to either the upper or lower values and the is no guarantee to which one it will converge.

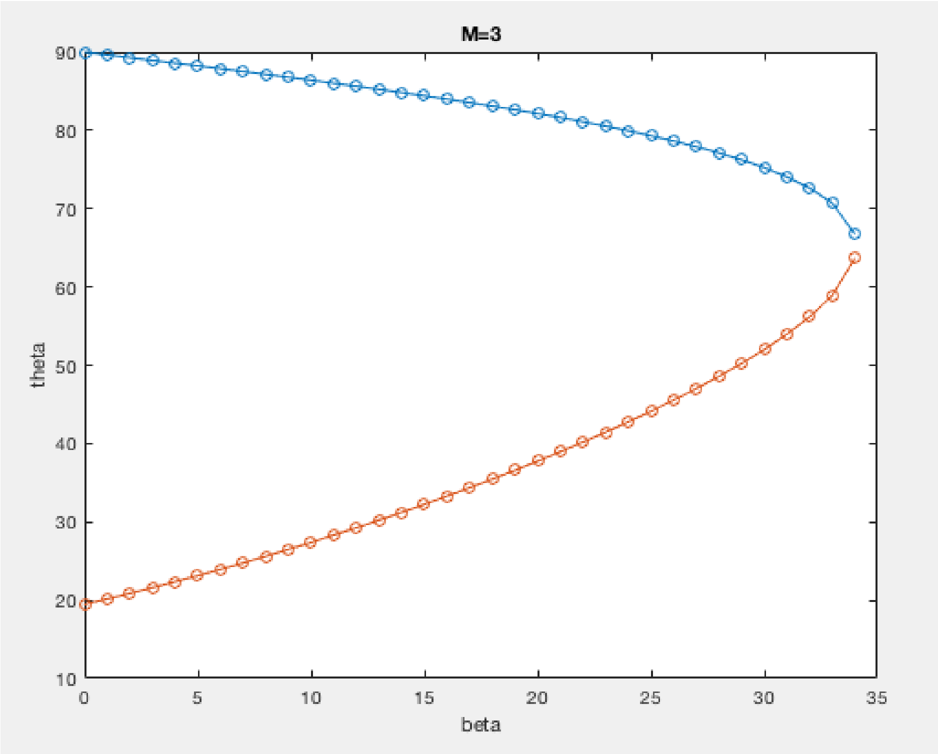
b.

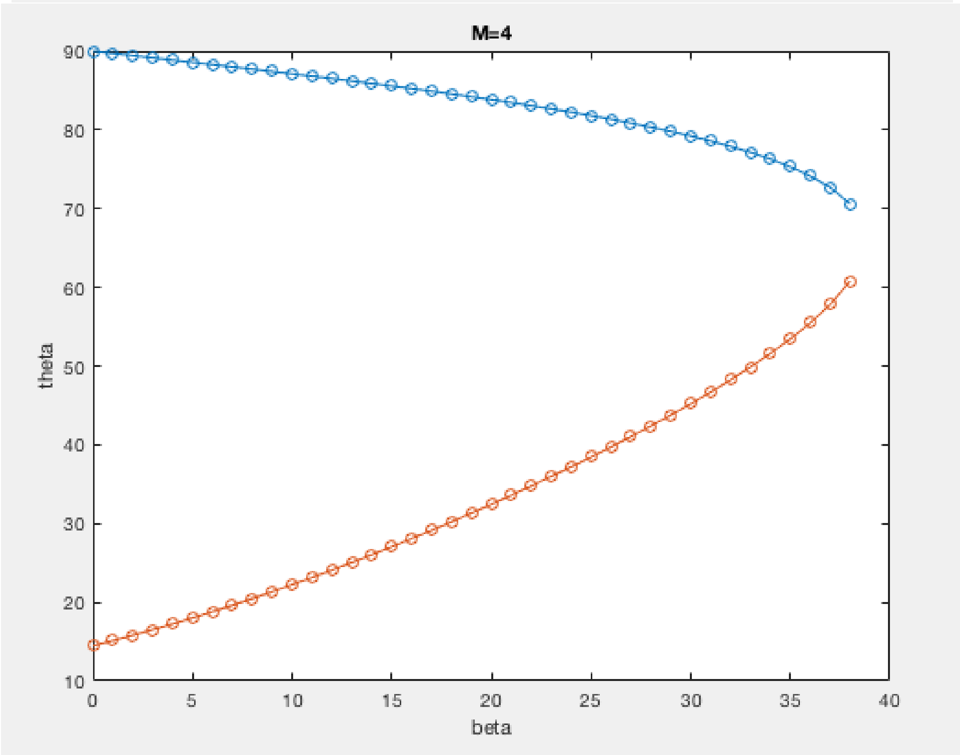


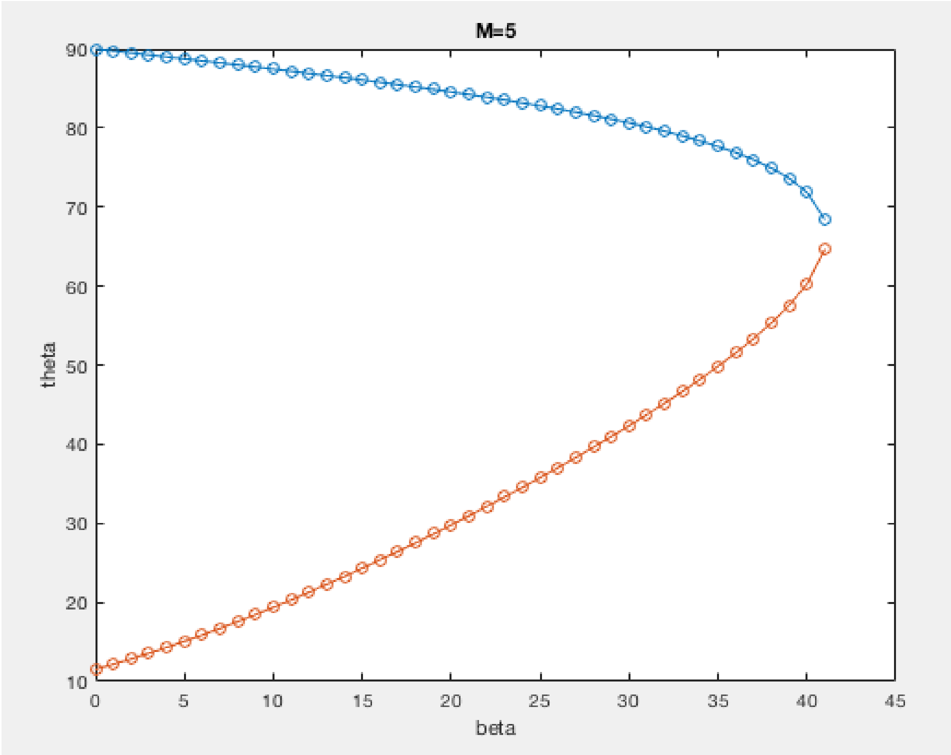

c.

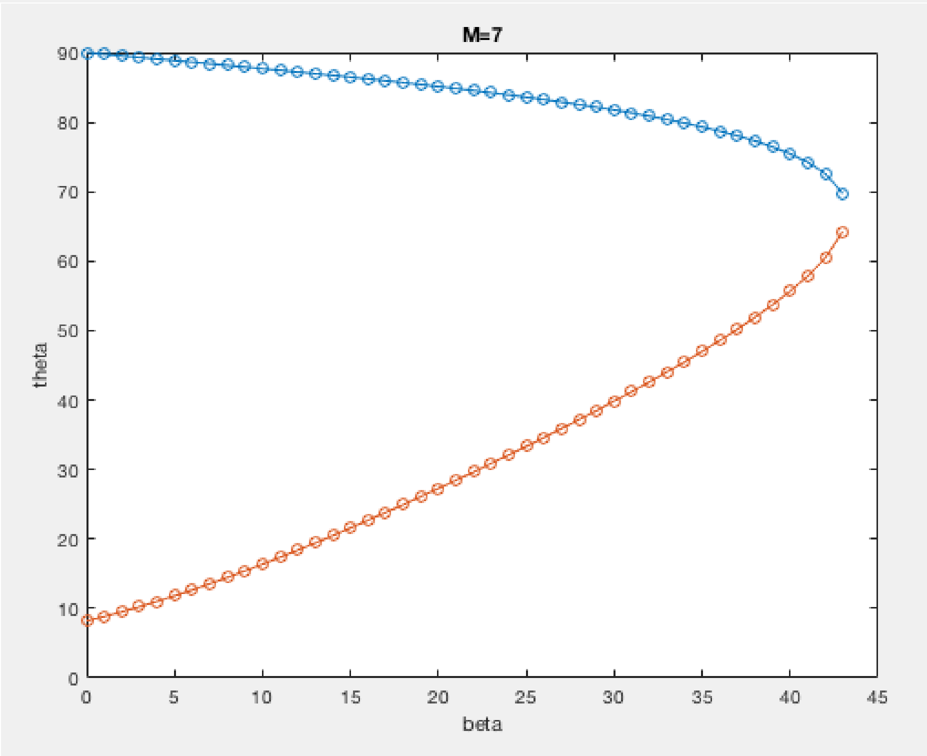


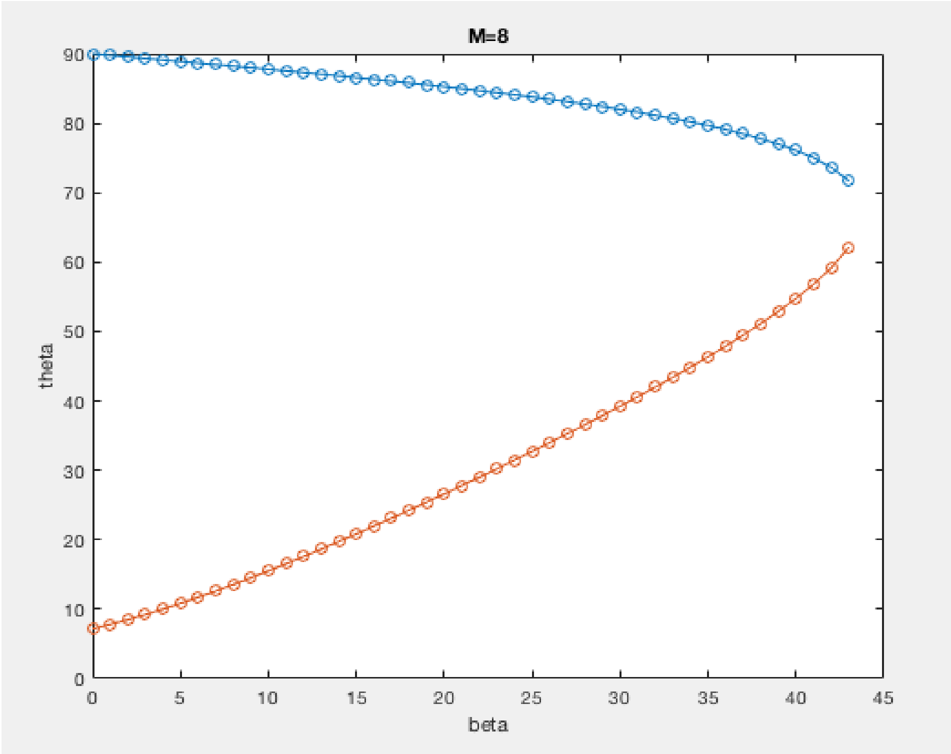
M=1.5
theta
beta
90
85
80
75
70
65
60
55
50
45
40
0
2
4
6
8
10
12


M=3
theta
beta
90
80
70
60
50
40
30
20
10
0
5
10
15
20
25
30
35


M=4
theta
beta
90
80
70
60
50
40
30
20
10
0
5
10
15
20
25
30
35
40


M=5
theta
beta
90
80
70
60
50
40
30
20
10
0
5
10
15
20
25
30
35
40
45


M=7
theta
beta
90
80
70
60
50
40
30
20
10
0
0
5
10
15
20
25
30
35
40
45


M=8
theta
beta
90
80
70
60
50
40
30
20
10
0
0
5
10
15
20
25
30
35
40
45

## Question 3: Linear Algebraic Systems

### Gauss Elimination Proof

3. GUASSIAN ELIMINATION PROOF

IN REDUCED - ROW FORM

$$\begin{pmatrix} a_1^* & b_1 & 0 & 0 \\ 0 & a_2^* & b_2 & 0 \\ 0 & 0 & a_3^* & b_3 \\ & & \ddots & \ddots \end{pmatrix}\begin{pmatrix}x_1\\x_2\\x_3\\\vdots\end{pmatrix}=\begin{pmatrix}Q_1^*\\Q_2^*\\Q_3^*\\\vdots\end{pmatrix}$$

$$\left(\begin{array}{cccc|c} a_1 & b_1 & 0 & 0 & Q_1\\ c_2 & a_2 & b_2 & 0 & Q_2 \\ 0 & c_3 & a_3 & b_3 & Q_3 \\ & \ddots & \ddots & \ddots & \vdots\end{array}\right)$$

For each $i$ : (~~except $i=0$~~, $i \in N$)

$(row_{i-1}) \times (c_i) = row_{i-1}^*$

$(row_i) \times (a_{i-1}) = row_i^*$

(not including $i = 1$)

then $(row_{i-1})^* \cdot (-1) + (row_i)^* \rightarrow (row_i)^*$

in $(row_i)^*$ : $a_i^* \Rightarrow -b_{i-1} \cdot c_i + a_i \cdot a_{i-1}^*$

$b_i^* \Rightarrow b_i \cdot a_{i-1}^*$

$c_i^* \Rightarrow -(a_{i-1}^* \cdot c_i) + (a_{i-1}^* \cdot c_i) = 0$

$Q_i^* \Rightarrow -Q_{i-1} \cdot c_i + Q_i \cdot a_{i-1}^*$

$(row_i)^* / a_{i-1}^*$ gives :

$$a_i^* = \frac{-b_{i-1} \cdot c_i}{a^*_{i-1}} + a_i$$

$$b_i^* = b_i$$

$$c_i^* = 0$$

$$Q_i^* = \frac{-Q_{i-1} \cdot c_i}{a^*_{i-1}} + Q_i$$

thus the matrise can be written in the form:

$$\begin{pmatrix} a_1^* & b_1 & 0 & 0 \\ 0 & a_2^* & b^2 & 0 \\ 0 & 0 & a_3^* & 0 \\ \ddots & \ddots & \ddots & \ddots \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \end{pmatrix} = \begin{pmatrix} Q_1^* \\ Q_2^* \\ Q_3^* \end{pmatrix}$$

with $c_i^* = 0$

## Question 4: Regression

4. REGRESSION PROOF

$$\begin{bmatrix} \sum_{i=1}^{N} x_i^2 & \sum_{i=1}^{N} x_i \\ \sum_{i=1}^{N} x_i & N \end{bmatrix} \begin{Bmatrix} a \\ b \end{Bmatrix} = \begin{Bmatrix} \sum_{i=1}^{N} x_i y_i \\ \sum_{i=1}^{N} y_i \end{Bmatrix}$$

$$\Rightarrow \begin{Bmatrix} a \\ b \end{Bmatrix} = \frac{1}{N\sum_{i=1}^{N} x_i^2 - \sum^{N} x_i^2} \begin{bmatrix} N & -\sum_{i=1}^{N} x_i \\ -\sum_{i=1}^{N} x_i & \sum_{i=1}^{N} x_i^2 \end{bmatrix} \begin{Bmatrix} \sum_{i=1}^{N} x_i y_i \\ \sum_{i=1}^{N} y_i \end{Bmatrix}$$

$$a = \left(N\sum_{i=1}^{N} x_i y_i - \sum_{i=1}^{N} x_i y_i\right) \div \left(N\sum_{i=1}^{N} x_i^2 - \sum_{i=1}^{N} x_i^2\right)$$

$$= (N-1)\sum_{i=1}^{N} y_i \div (N-1)\sum_{i=1}^{N} x_i$$

$$= \frac{(N-1)\sum_{i=1}^{N} y_i}{(N-1)\sum_{i=1}^{N} x_i} \div \times \left(\frac{N}{N}\right)$$

$$= \frac{\sum_{i=1}^{N} y_i - \frac{1}{N}\sum_{i=1}^{N} y_i}{\sum_{i=1}^{N} x_i - \frac{1}{N}\sum_{i=1}^{N} x_i} \times \left(\frac{\sum_{i=1}^{N}(x_i - \bar{x})}{\sum_{i=1}^{N}(x_i - \bar{x})}\right)$$

$$= \frac{\sum_{i=1}^{N} y_i(x_i - \bar{x}) - \sum_{i=1} \bar{y}(x_i - \bar{x})}{\sum_{i=1}^{N}(x_i - \bar{x})^2}$$

$$= \frac{\sum_{i=1}^{N}(y_i x_i - y_i\bar{x} - \bar{y}x_i + \bar{y}\bar{x})}{\sum_{i=1}^{N}(x_i - \bar{x})^2}$$

$$\Rightarrow \quad a = \frac{\sum_{i=1}^{N} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{N} (x_i - \bar{x})^2}$$

$\cancel{b =}$

using matrix $eq^n$ 2

$$a \sum_{i=1}^{N} x_i + b \cdot N = \sum_{i=1}^{N} y_i$$

$$\Rightarrow \quad b \cdot N = \sum_{i=1}^{N} y_i - a \sum_{i=1}^{N} x_i$$

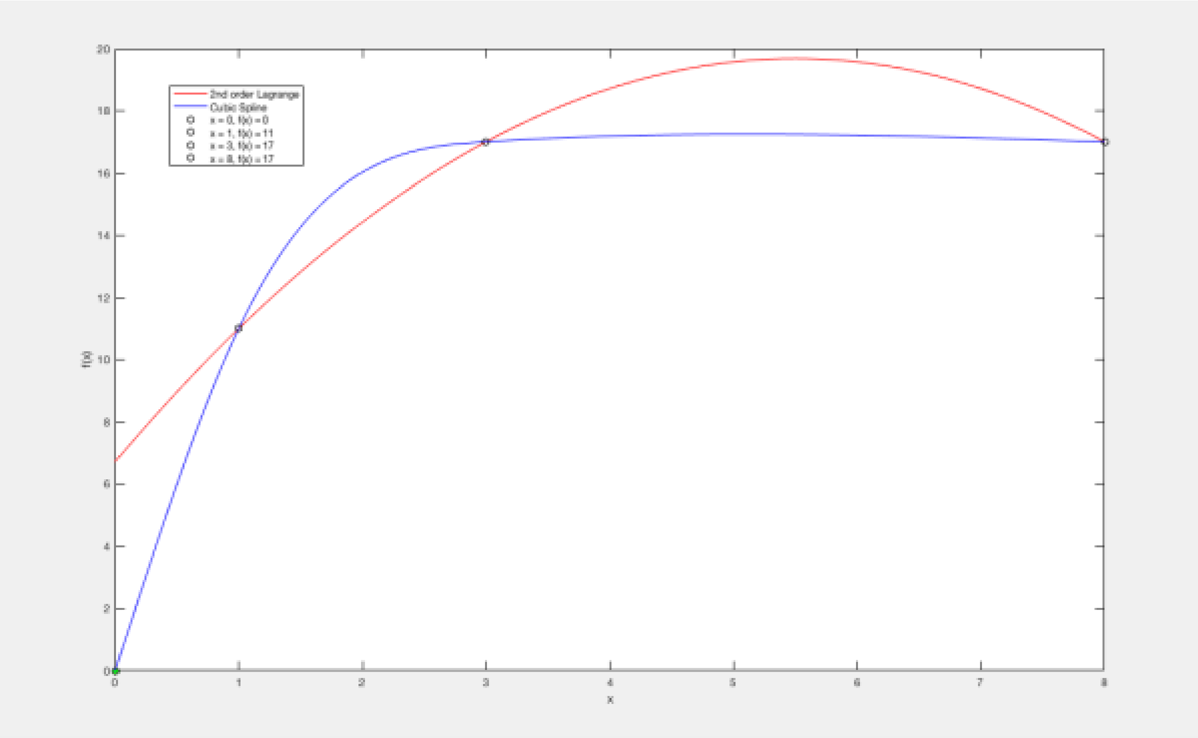
$$b = \sum_{i=1}^{N} y_i / N - a \sum_{i=1}^{N} x_i / N$$

$$b = \bar{y} - a \bar{x}$$

**When linear regression will fail to find a solution**

Linear regression will fail to find a solution when $\sum (x_i - \bar{x})$ is equal to 0, this will only occur when the x points are all on the same point, i.e. when the graph is a vertical line.

Question 5: Interpolation




**Conclusions**

Shown above, the cubic spline has a much greater accuracy of estimating a function than the 2nd order lagrange does. The functions are both very accurate at the points that were given but the further away the function moves from those points the less accurate the 2nd order lagrange function becomes. The cubic spline is a much better estimation for a function but has the disadvantage that it needs extra conditions at the end points, a natural spline is the simplest solution to this drawback when no other information about the function is available.

# Question 6: Differentiation, differential equations

**Discussion**

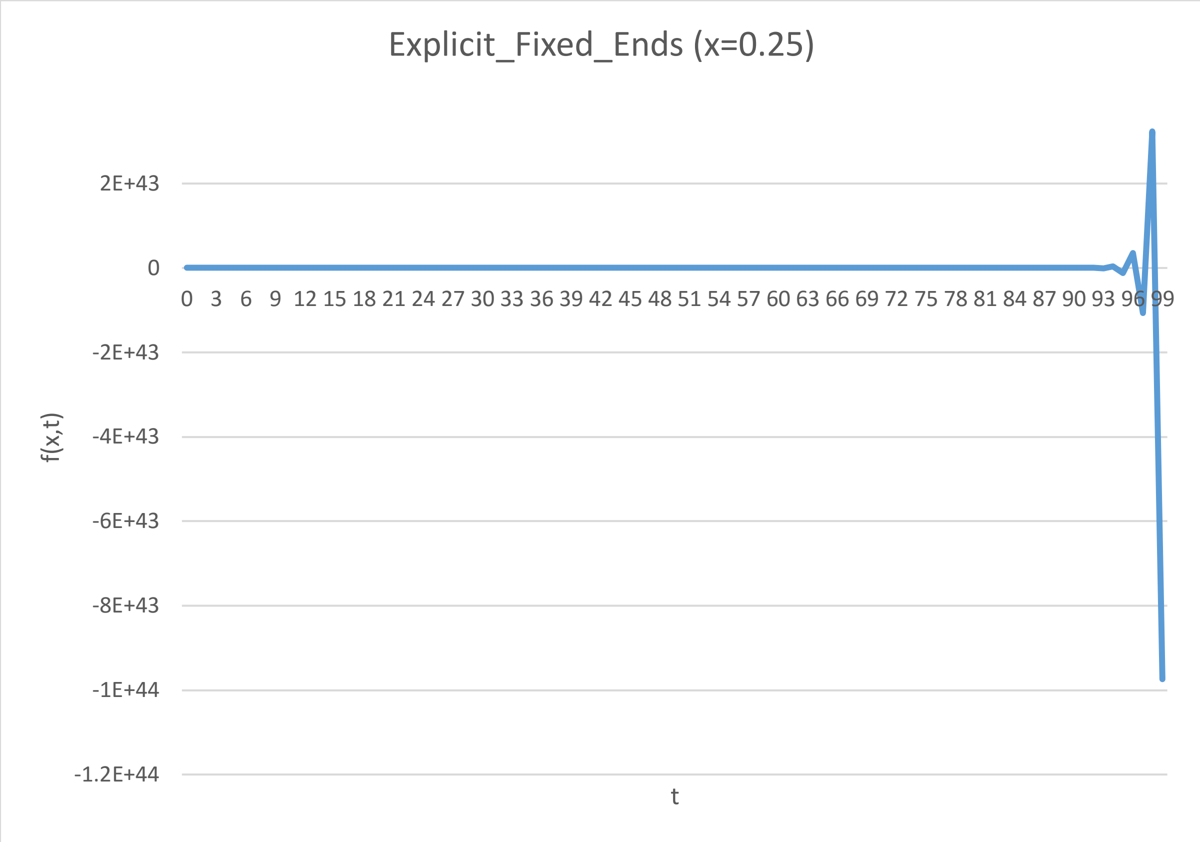
The explicit method is not very accurate for estimating the function values at timesteps that are larger than $t = 12$ (shown in graphs 2 and 4) as the solution continues to expand indefinitely. Whether fixed ends or variable ends are used does not have an effect on the stableness of the explicit solution and only helps to find solutions for the initial and final values. Graph 6 and 7 show the difference between using fixed ends vs variable ends, the system is able to find a solution for the values in the variable end solution in Graph 6 but the values remain zero in Graph 7.

In contrast, the implicit method has a very stable solution which doesn't blow up like the explicit method and returns consistent values that improve with accuracy for each timestep this is shown in Graph 5 where $x = 0.25$. The Graph appears to converge towards a single point. This stableness is explained because the implicit solution uses information from the same time level to calculate the solution, whereas the explicit scheme uses information previously calculated which is independent of the current time level. By using information that does not rely on the current time level an error is accumulated which is then amplified with each timestep. With the error at any one timestep equal to $xn = x0*(1 + h\lambda R + ih\lambda I) = x0*\sigma^n$, the amplification factor $\sigma$ increases for every timestep n which results in $x0*\sigma^n$ increasing exponentially making it very easy for the explicit system to fail the requirement for stability, $|\sigma| \leq 1$.

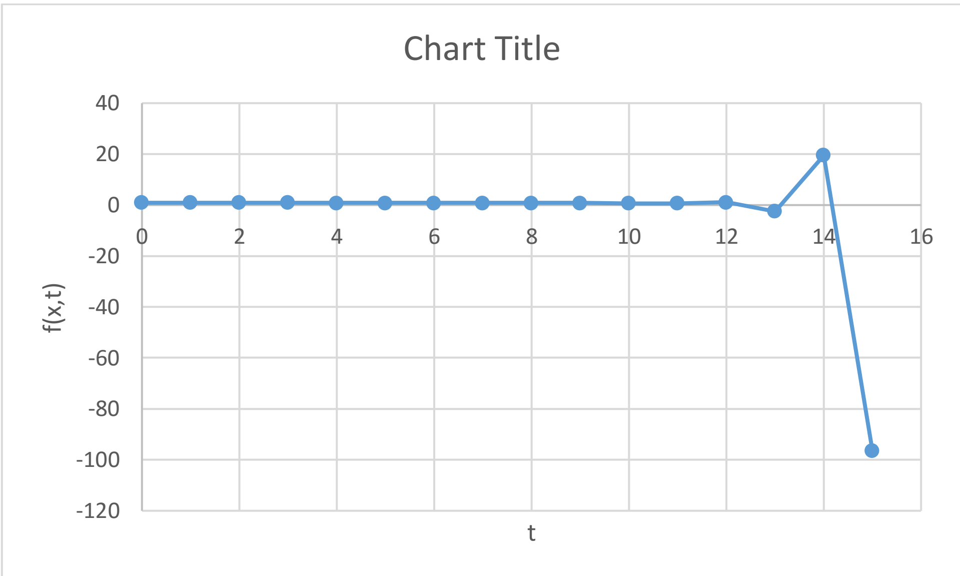
The explicit solution does not converge to a value while the implicit solution seems to converge towards a solution with larger timesteps. The conditional stableness of the implicit solution makes it a much better solution to the system however, there is the trade-off of the complexity of coding up the implicit system and solving for a larger amount of unknowns.

**Graphs**

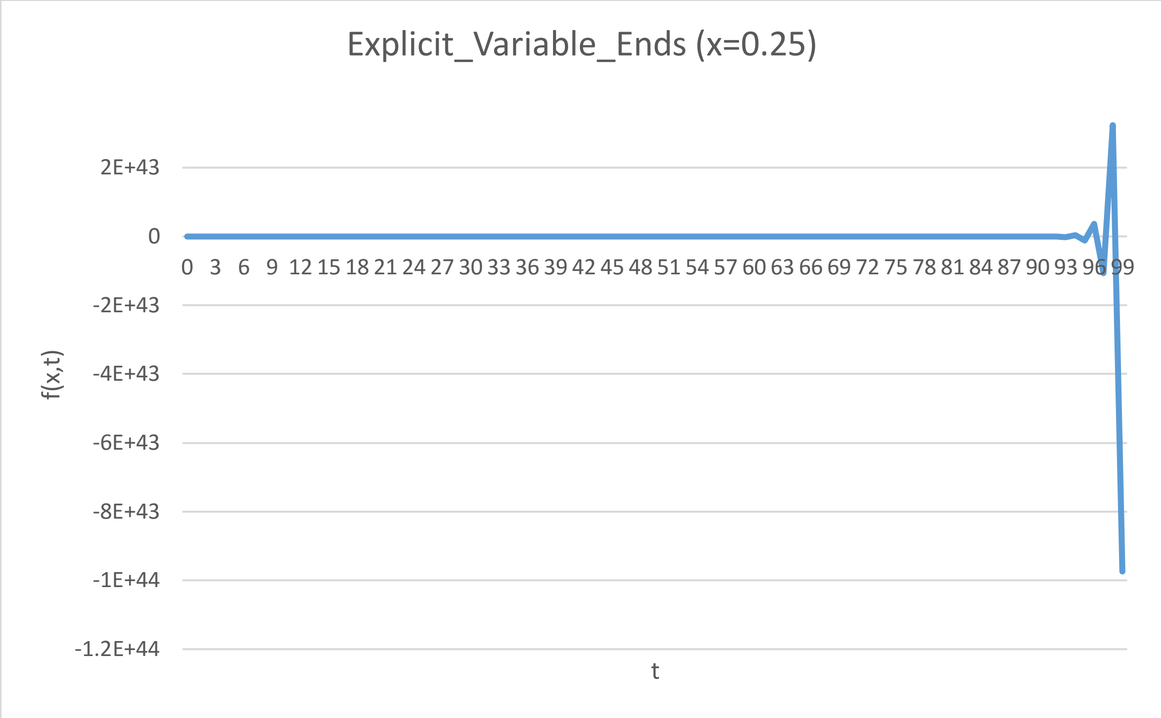
Graph 1: Explicit_fe (x=0.25)



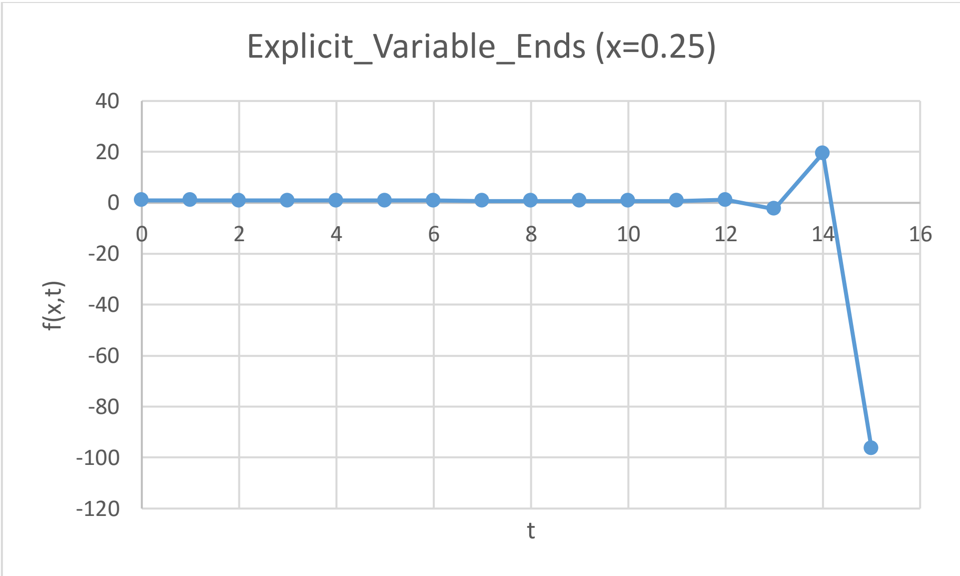

Graph 2: Explicit_fe (x=0.25)



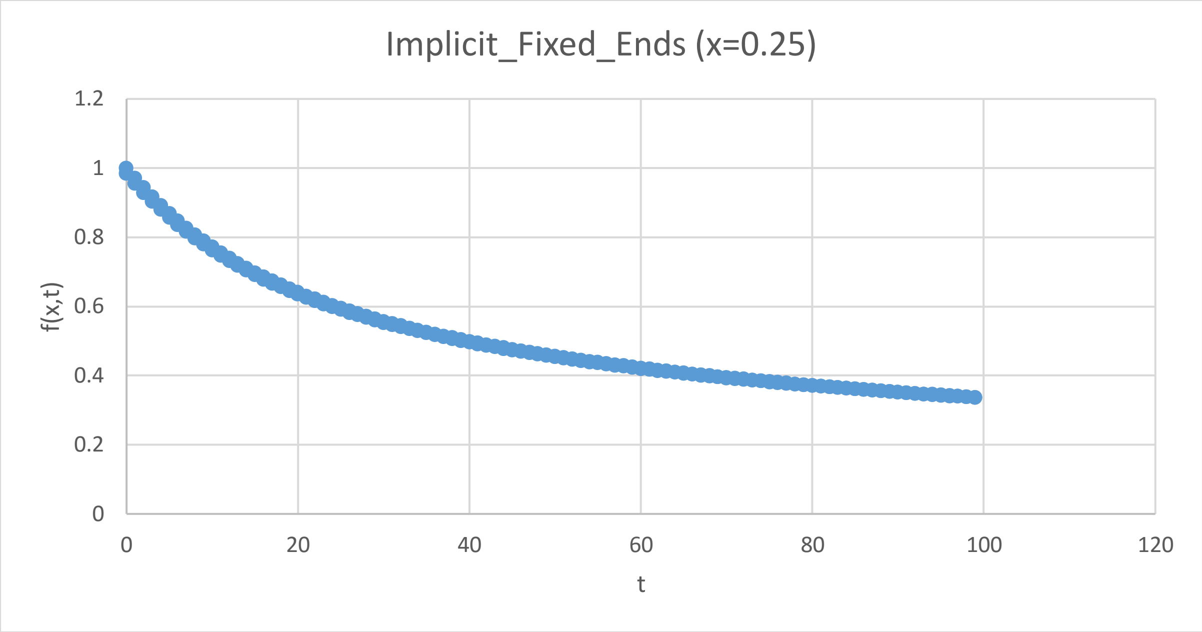

Graph 3: Explicit_ve (x=0.25)



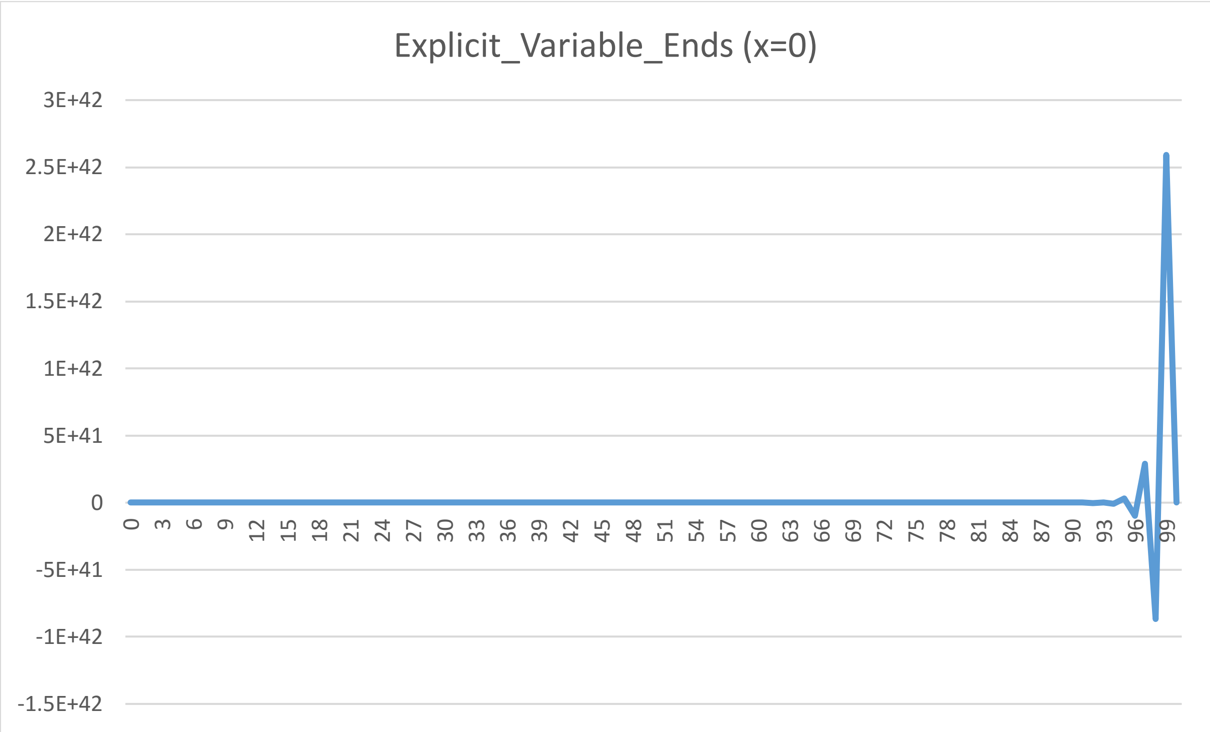

Graph 4: Explicit_ve (x=0.25)



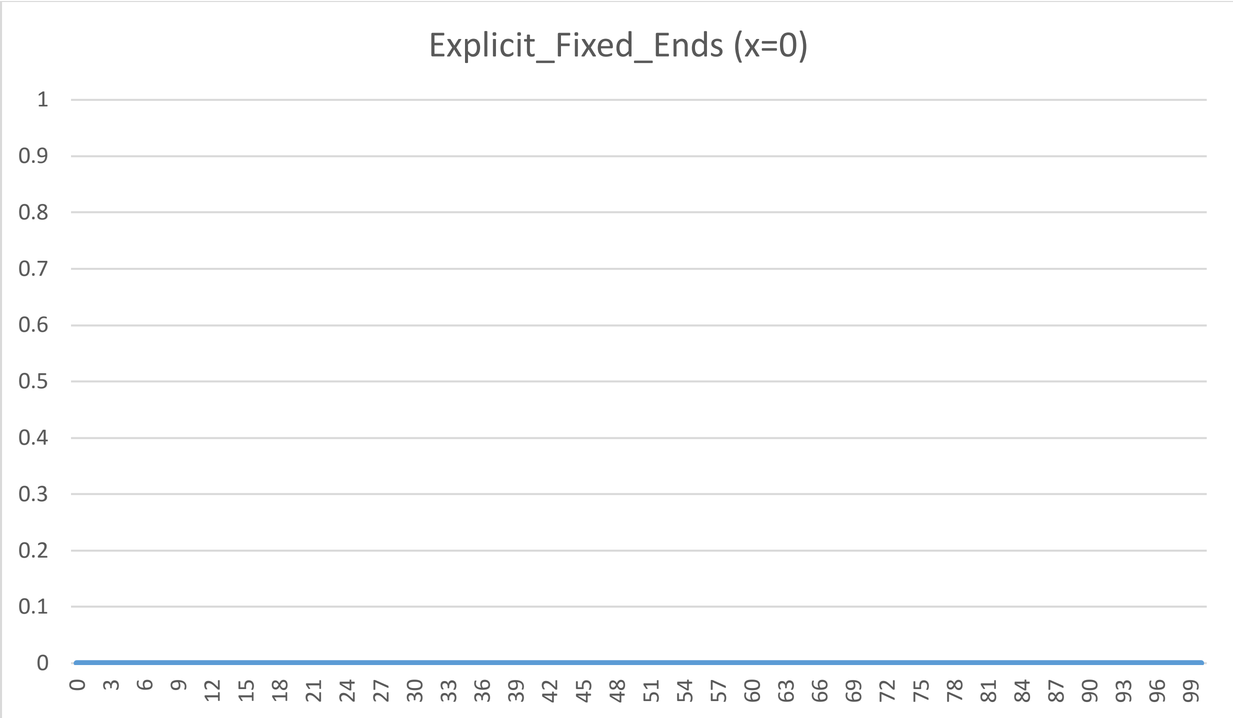

Graph 5: Implicit_fe (x=0.25)




Graph 6: Explicit_ve (x=0.0)




Graph 7: Explicit_fe (x=0.0)



Explicit_Fixed_Ends (x=0)
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0 3 6 9 12 15 18 21 24 27 30 33 36 39 42 45 48 51 54 57 60 63 66 69 72 75 78 81 84 87 90 93 96 99