DS-288/UE-201 - Assignment 1

Aditi Arun(20438)

September 17, 2023

Answer 1

We need to find E at the time of Lunar Orbit Injection.

Kepler's equation gives us $M = E - e\sin E$, where $e = 0.0549$ (1)

To solve for E we first need to find M at LOI.

We are given that the Moon was at periapsis at the time of launch.
We are also given the date and time of the launch as 14th July 2:40pm, and date and time of entering lunar gravity to be 5th August 7pm.

From this, we can calculate the time elapsed from the launch till the spacecraft entered the lunar orbit.
We get, 22 days + 4hrs + 20 minutes.
$= 22$ days $+ \dfrac{13}{3}$ hrs
$= 22 + \dfrac{13}{72}$ days
$= \dfrac{1597}{72}$ days

Now, M at LOI is the fraction of the orbit covered in this time as an angle.

Since total time period is given to be 27.32 days, we can calculate the fraction M as follows
M $= \dfrac{1597}{72\times 27.32}\times 2\pi = \dfrac{1597}{1967.04}\times 2\pi$

We can now substitute the value of M and e in Eq (1) and solve for E
$E - 0.0549\sin E - \dfrac{1597}{1967.04}\times 2\pi = 0$
Therefore, we have $f(x) = x - 0.0549\sin x - \dfrac{1597}{1967.04}\times 2\pi = x - 0.0549\sin x - 5.101191$
Substituting $x = \pi$, we get $f(\pi) = -1.95959$
Substituting $x = 2\pi$, we get $f(2\pi) = 1.181994$

By IVT, a root exists between $x = \pi$ and $x = 2\pi$

For FPI we need an iteration function of the form $x = g(x)$.
We first try the most obvious iteration function-
$x = g(x) = 0.0549\sin x + 5.101191$

Consider g(x) between $x = \pi$ and $x = 2\pi$
max(g(x)) for x in (π,2π) = 0.0549 max($\sin x$) + 5.101191= 0+ 5.101191 = 5.101191
min(g(x)) for x in (π,2π) = 0.0549 min($\sin x$) + 5.101191= -0.549 + 5.101191 = 5.046291
Therefore we have $\pi < 5.046291 < g(x) < 5.101191 < 2\pi$, for x in (π,2π)

We also have $g'(x) = 0.0549cosx$
Therefore, $|g'(x)| < 0.0549 < 1$

Thus, the conditions of fixed point iteration are satisfied for the given iteration function-
$g(x) = 0.0549\sin x + 5.101191$, between π and 2π

We carry out FPI with the starting point $3\pi/2$.

The root is found to be 5.049378991944495 under relative tolerance of 10^{-8}. The number of iterations are 5.

Therefore, we have found the value of E = 5.049378991944495.

The equations of the coordinates are
x = a(cosE-e)
y = bsinE
We substitute E = 5.049378991944495, e=0.0549, a=384400 and b = 383820 km

Thus, we obtain the coordinates
x= 105997.49471278557 km y = -362231.7315228139 km This is the required position.

Answer 2

a) Consider a function $f(x)$ with a simple root $x = \alpha$
Let the n^{th} iteration of Newton's method be x_n
Define error at iteration n, $e_n = \alpha - x_n$

We have $f(\alpha) = 0$

$$f(\alpha - x_n + x_n) = f(e_n + x_n) = 0$$

From the Taylor series expansion, we get

$$0 = f(e_n + x_n) = f(x_n) + e_n f'(x_n) + \frac{e_n^2}{2} f''(x_n) + \cdots$$

~~We neglect higher order terms to obtain~~ This can now be expressed as

$$0 = f(x_n) + e_n f'(x_n) + \frac{e_n^2}{2} f''(\xi_n) \qquad \xi_n \text{ is between } x_n \text{ and } \alpha$$

Dividing by $f'(x_n)$, we have

$$e_n + \frac{f(x_n)}{f'(x_n)} + \frac{e_n^2}{2} \frac{f''(\xi_n)}{f'(x_n)} = 0$$

$$\alpha - x_n + \frac{f(x_n)}{f'(x_n)} = -\frac{e_n^2}{2} \frac{f''(\xi_n)}{f'(x_n)}$$

$$\alpha - \left(x_n - \frac{f(x_n)}{f'(x_n)}\right) = \frac{-e_n^2}{2} \frac{f''(\xi_n)}{f'(x_n)}$$

$$\alpha - x_{n+1} = \frac{-e_n^2}{2} \frac{f''(\xi_n)}{f'(x_n)}$$

$$e_{n+1} = \frac{-e_n^2}{2} \frac{f''(\xi_n)}{f'(x_n)}$$

Taking mod on both sides, we obtain

$$|e_{n+1}| \leq \frac{1}{2} \left|\frac{f''(\xi_n)}{f'(x_n)}\right| (e_n)^2$$

Now, consider $\left|\frac{f''(\xi_n)}{f'(x_n)}\right|$. ~~When multiplicity is k, this is bounded~~.

Let us choose an appropriate value δ and define

$$M = \frac{1}{2}\frac{\max\limits_{x\in(\alpha-\delta,\alpha+\delta)} |f''(x)|}{\min\limits_{x\in(\alpha-\delta,\alpha+\delta)} |f'(x_n)|} \quad (1)$$

$\therefore$ We get $|e_{n+1}| \leq M e_n^2$

Thus we have proved quadratic convergence for a simple root.

Now, when multiplicity $m>1$, we have $f'(\alpha)=0$

Here, we cannot express the term in the form of equation (1) since the denominator will tend to 0.

Let the multiplicity of the root be m.

$\therefore f(\alpha)=f'(\alpha)=f''(\alpha)\ldots f^{m-1}(\alpha)=0,\ f^{m}(\alpha)\neq 0$

Thus, the Taylor series expansion at α reduces to

$$f(x_n) = \underbrace{f(\alpha)+f'(\alpha)(\alpha-x_n)+\cdots}_{=0} + \frac{f^m(\alpha)}{m!}(x-\alpha)^m + \cdots$$

$$f(x_n) = \frac{f^m(\alpha)}{m!}(x_n-\alpha)^m + \cdots \quad (2)$$

This can be expressed as $f(x_n) = \frac{1}{2}f''(\xi_n)(x_n-\alpha)^2$ $\quad$ ξ_n is in between x_n and α

Differentiating equation (2) we get

$$f'(x_n) = \frac{f^m(\alpha)}{m!}\, m\,(x-\alpha)^{m-1} + \cdots$$

Like earlier, this can be condensed and expressed as $f'(x_n)=f''(\gamma_n)(x_n-\alpha)$ where γ is in between x_n and α

$\therefore f'(x_n) = -f''(\gamma_n)(\alpha-x_n)$

Subitituting $f'(x_n)$ in the eq $e_{n+1} = \dfrac{-e_n^2}{2} \dfrac{f''(\xi_n)}{f'(x_n)}$, we get

$$e_{n+1} = \frac{-e_n^2}{2} \frac{f''(\xi_n)}{-f''(\gamma_n)(\alpha - x_n)}$$

$$e_{n+1} = \frac{e_n}{2} \frac{f''(\xi_n)}{f''(\gamma_n)}$$

Taking mod both sides, we have

$$|e_{n+1}| \leq \left| \frac{f''(\xi_n)}{2f''(\gamma_n)} \right| |e_n|$$

like earlier, we can choose an appropriate value of M

$\therefore$ We get $|e_{n+1}| \leq M|e_n|$

Thus, we have ~~quadratic~~ linear convergence for $m >$ [redacted]

Now, we use Modified Newton's method

Since $f(x)$ has a root α with multiplicity m, we can express

$$f(x) = (x-\alpha)^m q(x)$$

let us define $u(x) = \dfrac{f(x)}{f'(x)}$

$$\therefore u(x) = \frac{(x-\alpha)^m q(x)}{m(x-\alpha)^{m-1} q(x) + (x-\alpha)^m q'(x)}$$

$$= (x-\alpha)\left[\frac{q(x)}{m q(x) + (x-\alpha) q'(x)}\right]$$

Therefore, $x-\alpha$ is now a simple root of $u(x)$

$\therefore$ We now perform Newton's method on $u(x)$ ~~to get~~

$$x_{n+1} = x_n - \frac{u(x_n)}{u'(x_n)} = x_n - \frac{f(x_n) f'(x_n)}{((f'(x_n))^2 - f(x_n) f''(x_n))}$$ Modified Newton's Method

Since multiplicity is 1 for $u(x)$ we have quadratic convergence (Proved earlier)

b) In Secant Method, we have the following iteration

$$x_{n+1} = x_n - \frac{f(x_n)(x_n - x_{n-1})}{f(x_n) - f(x_{n-1})}$$

We have error $e_n = x_n - \alpha$, $e_{n+1} = x_{n+1} - \alpha$

$\therefore x_n = e_n + \alpha,\ x_{n+1} = e_{n+1} + \alpha,\ x_{n-1} = e_{n-1} + \alpha$

Substituting above, we get

$$e_{n+1} + \cancel{\alpha} = e_n + \cancel{\alpha} - \frac{f(x_n)(e_n - e_{n-1})}{f(e_n+\alpha) - f(e_{n-1}+\alpha)}$$

Now, consider the Taylor series expansion of $f(x_n)$ around α

$$f(x_n) = f(e_n + \alpha) = f(\alpha) + f'(\alpha)e_n + \frac{1}{2}f''(\alpha)e_n^2 + \cdots$$

$f(\alpha) = 0$, and we neglect the higher order terms

$$\therefore f(e_n+\alpha) \approx f'(\alpha)e_n + \frac{1}{2}f''(\alpha)e_n^2$$

Similarly $f(e_{n-1}+\alpha) \approx f'(\alpha)e_{n-1} + \frac{1}{2}f''(\alpha)e_{n-1}^2$

$$\therefore f(e_n+\alpha) - f(e_{n-1}+\alpha) = (e_n - e_{n-1})\left[f'(\alpha) + \frac{f''(\alpha)}{2}(e_n + e_{n-1})\right]$$

We get $e_{n+1} = e_n - \dfrac{f(x_n)\cancel{(e_n - e_{n-1})}}{\cancel{(e_n - e_{n-1})}\left[f'(\alpha) + \dfrac{f''(\alpha)}{2}(e_n + e_{n-1})\right]}$

From the Taylor series, we had

$f(x_n) = f(e_n + \alpha) = f'(\alpha)e_n + \frac{1}{2}f''(\alpha)e_n^2$

$\therefore e_{n+1} = e_n - \dfrac{f'(\alpha)e_n + \frac{1}{2}f''(\alpha)e_n^2}{f'(\alpha) + \dfrac{f''(\alpha)}{2}(e_n + e_{n-1})}$

Dividing numerator and denominator by $f'(\alpha)$, we get

$e_{n+1} = e_n - \dfrac{e_n\left[1 + \dfrac{f''(\alpha)}{2f'(\alpha)}e_n\right]}{1 + \dfrac{f''(\alpha)}{2f'(\alpha)}(e_n + e_{n-1})}$

Let $M = \dfrac{f''(\alpha)}{2f'(\alpha)}$

$\therefore e_{n+1} = e_n - \dfrac{e_n(1 + Me_n)}{1 + M(e_n + e_{n-1})} = \dfrac{M e_n e_{n-1}}{1 + M(e_n + e_{n-1})}$

Now, $M(e_n + e_{n-1}) = \dfrac{f''(\alpha)}{2f'(\alpha)}(e_n + e_{n-1}) <<< 1$

Therefore e_{n+1} can be approximated to $M e_n e_{n-1}$

$e_{n+1} \approx M e_n e_{n-1}$ (i)

Now, let us assume secant method has order of convergence o

$\therefore |e_{n+1}| \leq C|e_n|^o$ (ii)

Substituting the value from eq (i) we get

$M|e_n||e_{n-1}| \approx C|e_n|^o$

$\dfrac{M}{C}|e_{n-1}| = |e_n|^{o-1}$

$\therefore |e_n| = \left(\frac{M}{c}|e_{n-1}|\right)^{\frac{1}{o-1}}$

Substituting $n=n+1$ and Comparing with equation (ii)

$|e_{n+1}| = \left(\frac{M}{c}\right)^{\frac{1}{o-1}} |e_n|^{\frac{1}{o-1}}$

$\therefore$ c' here $= \left(\frac{M}{c}\right)^{\frac{1}{o-1}}$, and $o = \frac{1}{o-1}$

$\therefore$ $o = \frac{1}{o-1}$, Solving, we get $o = \frac{1+\sqrt{5}}{2} \approx 1.618$

This is the value of the Golden ratio.

$\therefore$ We have proved that secant method converges with the order of ~~Muller's method~~ the Golden ratio.

Given $f(x) = \cos x - x e^{x}$

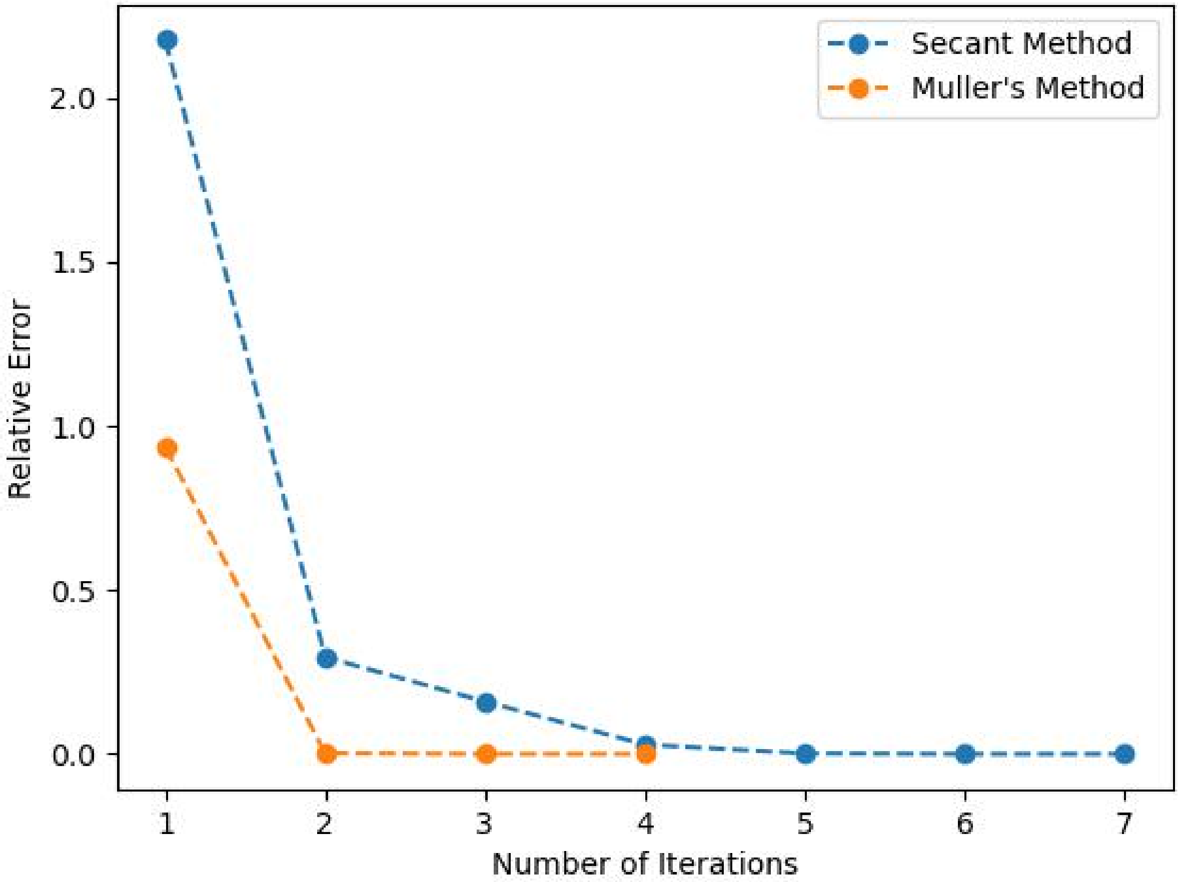
We find the root using Secant Method under 10^{-6} relative error tolerance to be 0.51177573636823997. It takes 7 iterations to converge.

Now, we find the same root using Muller's method under 10^{-6} relative tolerance. The root is found in 4 iterations to be 0.517757363624583

Thus, we see that Muller's Method converges faster than the ~~bisection~~ secant method. This is as expected, because the order of convergence of the secant's method is around 1.6. This is obtained by solving the quadratic equation $x^2-x-1=0$ as shown above.

Similarly, the order of convergence of Muller's method will be given by the solution of $x^3-x^2-x-1=0$. this gives us an order of convergence value of about 1.8. Thus, Muller's method converges faster.

The below plot shows the relative error with iterations for both methods.



The code is provided in q2.py with comments.

Answer 3

We are given that there exists less than 30 roots in (0,1)

We can use the Bisection method to locate the roots.
Let us start with a very small value d. We test whether a root is present in the interval [0,d]

If $f(0)f(d) < 0$, then a root exists by IVT.
We find the root by Bisection method. Then, we move on to check the next interval [d,2d].

If $f(0)f(d) > 0$, then a root doesn't exist, we move on to check the interval [d,2d].

This process is repeated and the interval is shifted by d each time, until the upper limit of the interval crosses 1.

In order for this method to work, the following assumption needs to hold. In the intervals [0,d],[d,2d], [2d,3d].... obtained, more than 1 root does not exist within each interval.
For a small enough choice of d, this condition will be satisfied.

Based on trial and error, the value of d was chosen to be 0.005.

26 roots were obtained. They are given below

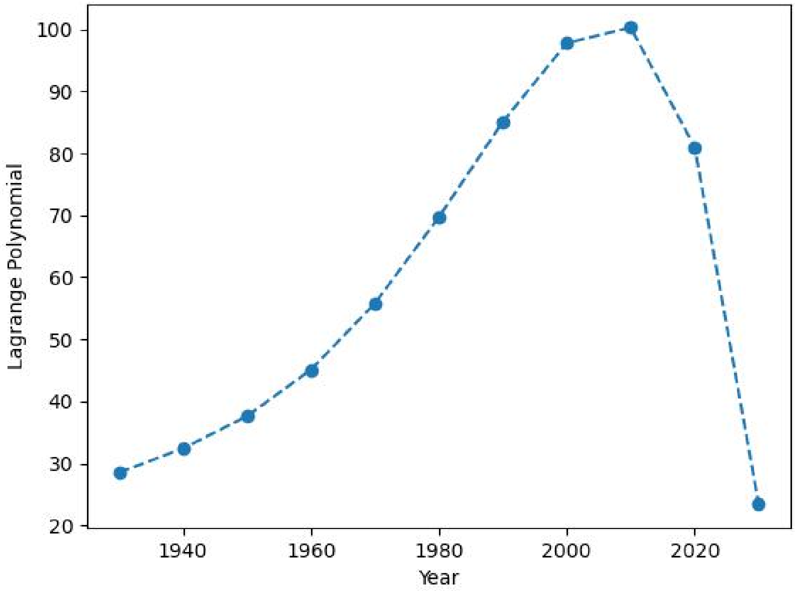
0.019268665313720706, 0.05775142669677733, 0.0964347076416016, 0.13462654113769534, 0.17374740600585944, 0.2113572692871095, 0.25120803833007826, 0.28794387817382827, 0.3288186645507814, 0.3643850708007815, 0.406582946777344, 0.44067779541015645, 0.48450897216796907, 0.5168154907226565, 0.5626095581054691, 0.5927841186523441, 0.6409075927734379, 0.6685614013671878, 0.719443969726563, 0.744104614257813, 0.798302612304688, 0.8193316650390632, 0.8776690673828133, 0.8940545654296882, 0.9582098388671881, 0.9676080322265631

The code is provided in q3.py with comments.

Answer 4

We write the code to find the Lagrange interpolation polynomial p(x). It is shown in q4.py

We plot this Lagrange polynomial from 1930 to 2030 by taking the population every 10 years.



We can now find the estimates of the population from 1990 to 2020 by substituting the year in the polynomial p(x)

We get:
p(1990)=85.10000000000002 Cr
p(2000)=97.79999999999995 Cr
p(2010)=100.30000000000064 Cr
p(2020)=81.00000000000364 Cr

The mean square error of this population estimate is calculated to be 1016.7024999998857 from the code.

This is a very large mean square error value. As we can see from the graph, the population is decreasing from the year 2010 to 2030. This is clearly in contradiction to our knowledge that the population of India increased between 2010 and 2020 and is expected to increase from 2020 to 2030 as well. The reason we are getting this trend is because Lagrange interpolation only fits the initial given 6 points but does not care about the future trend. Hence, it initially gives a good increasing trend as it fits the first 6 points, but then has a very steep decline. This is the reason for such a large error. Thus, this method is absolutely not suitable for predicting the population of India in the future years.