Assignment 4 - Numerov Method

SGTB Khalsa College, University of Delhi

Preetpal Singh(2020PHY1140)(20068567043)

Unique Paper Code: 32221501

Paper Title: Quantum Mechanics and Applications

Submitted on: August 19, 2022

B.Sc(H) Physics Sem V

Submitted to: Dr. Mamta

Name → Preetpal Singh Roll No. → 2020PHY1140

Assignment → Numerov Method (Lab Assignment-4)

Subject: Quantum Mechanics (Lab)

Date ..../..../.........

Q- Derive Numerov method algorithm for solving IVP

$$u''(x) + f(x)u(x) = 0 \quad \text{with } u(a) = u_0 \quad u(a+h) = u_1$$

$x \in [a,b]$ and with $h = (b-a)/N$

And $u''(x) + k^2(x)u(x) = 0$

Taking taylor expansion of $u(x \pm h)$

$$u(x+h) = u(x) + hu'(x) + \frac{h^2 u^2(x)}{2!} + \frac{h^3 u^3(x)}{3!} + \frac{h^4 u^4(x)}{4!} + \frac{h^5 u^5(x)}{5!} + \cdots$$

$$u(x-h) = u(x) - hu'(x) + \frac{h^2 u^2(x)}{2!} - \frac{h^3 u^3(x)}{3!} + \frac{h^4 u^4(x)}{4!} - \frac{h^5 u^5(x)}{5!}$$

$$u(x+h) + u(x-h) = 2u(x) + h^2 u^2(x) + \cancel{2} \times \frac{h^4 u^4(x)}{\underset{2}{\cancel{4}} \times 3!} + O(h^6)$$

$$u^2(x) = \frac{u(x+h) + u(x-h) - 2u(x)}{h^2} - \frac{h^2 u^4(x)}{12} + O(h^4) \quad -(1)$$

$$u^2(x) + \frac{h^2 u^4(x)}{12} = \left(1 + \frac{h^2}{12}\frac{d^2}{dx^2}\right) u^2(x)$$

$\cancel{u^2(x) + \frac{h^2 u^4(x)}{12}}$

Applying this operator on given differential equation

$$\left(1 + \frac{h^2}{12}\frac{d^2}{dx^2}\right)\left(u''(x) + k^2(x)u(x)\right) = 0$$

$$u''(x) + k^2(x)u(x) + \frac{h^2}{12}\frac{d^4 u(x)}{dx^4} + \frac{h^2}{12}\frac{d^2}{dx^2}k^2(x)u(x) = 0$$

$$\frac{u(x+h) + u(x-h) - 2u(x)}{h^2} + k^2(x)u(x) + \frac{h^2}{12}\frac{d^2}{dx^2}k^2(x)u(x) = 0 \quad -(1)$$

$$\frac{d^2}{dx^2}k^2(x)u(x) = \frac{k^2(x+h)u(x+h) + k^2(x-h)u(x-h) - 2k^2(x)u(x)}{h^2} + O(h^2)$$

Spiral

Date ..../..../........

Take $x = x_i$ $\quad x+h = x_{i+1}$ $\quad x-h = x_{i-1}$

$\cancel{y(x) = y_i}$ $\quad \cancel{y(x+h) = y_{i+1}}$ $\quad \cancel{y(x-h) = y_{i-1}}$

$u(x) = u_i$ $\quad u(x+h) = u_{i+1}$ $\quad u(x-h) = u_{i-1}$

putting these in Eqn (1)

$$u_{i+1} + u_{i-1} - 2u_i + h^2 K^2 u_i + \frac{h^2}{12}\left[\frac{K_{i+1}^2 u_{i+1} + K_{i-1}^2 u_{i-1} - 2K_i^2 u_i}{h^2}\right] = 0$$

$$u_{i+1} + u_{i-1} - 2u_i + h^2 u_i + \frac{h^2 K_{i+1}^2 u_{i+1}}{12} + \frac{h^2 K_{i-1}^2 u_{i-1}}{12} - \frac{h^2 2K_i^2 u_i}{12} = 0$$

$$= u_{i+1}\left[1 + \frac{h^2}{12}K_{i+1}^2\right] + u_{i-1}\left[1 + \frac{h^2}{12}K_{i-1}\right] + u_i\left[-2 + h^2 - \frac{h^2 \times 2K_i^2}{12}\right]$$

$$= u_{i+1}\left[1 + \frac{h^2 K_{i+1}^2}{12}\right] + u_{i-1}\left[1 + \frac{h^2}{12}K_{i-1}\right] - 2u_i\left[1 - \frac{h^2}{2} + \frac{h^2 K_i^2}{12}\right] = 0$$

$$= u_{i+1}\left[1 + \frac{h^2}{12}K_{i+1}^2\right] = 2u_i\left[1 - \frac{h^2}{2} + \frac{h^2 K_i^2}{12}\right] - u_{i-1}\left[1 + \frac{h^2}{12}K_{i-1}\right]$$

$\Rightarrow$

$$u_{i+1} = \frac{2u_i\left[1 - \frac{5h^2K_i^2}{12}\right] - u_{i-1}\left[1 + \frac{h^2}{12}K_{i-1}\right]}{\left[1 + \frac{h^2}{12}K_{i+1}^2\right]} + O(h^6)$$

So, this Numerov method can be used to determine ~~y_p~~ for $p = 2, 3, ----$ given two initial conditions u_0 and u_1 are required to solve 2nd order differential equation

Take

Date ..../..../........

(b) Discuss local and global truncation errors...

Ans. The local truncation errors are the errors which are caused by one iteration. In this case,

The error in one x-step is $O(h^6)$

Global Truncation error is total error due to cumulative error caused by many iterations.

Now, number of steps needed to integrate over a fix range of x, from a to b is

$$\frac{b-a}{n} \propto \frac{1}{n}$$

here ~~n~~ n is no. of intervals between a and b

We can expect that error at each step would be roughly comparable to total error in Numerov method that would be $O(h^5)$

$\therefore$ Local Truncation Error $\sim O(h^6)$

Global Error $\sim O(h^5)$

(c) Numerov Method to solve IVP with initial conditions $u(a) = u_0$, $u'(a) = du_0$ without affecting the order of local truncation error.

Ans. To approximate with Numerov method, we need 2 initial conditions $u(a)$ and $u(a+h)$,

Here we have $u(a) = u_0$ and $u'(a) = du_0$, we will approximate 2nd initial condition using Taylor expansion

$$u(a+h) = u(a) + hu'(a) + \frac{h^2 u^2(a)}{2!} + \frac{h^3 u^3(a)}{3!} + \frac{h^4 u^4(a)}{4!} +$$

From differential Equation we have

$$u''(x) = -f(x)u(x)$$

$$u''(a) = -f(a)\,u(a)$$

Date ..../..../........

$$u'''(a) = -\left[f(a)\underset{du}{u'(a)} + u(a) f'(a) \right]$$

$$u^4(a) = -\left[f(a)u''(a) + f'(a)u'(a) + u(a)f''(a) + u'(a)f'(a) \right]$$

$$= -\left[f(a)u''(a) + 2u'(a)f'(a) + u(a)f''(a) \right]$$

$$u(a+h) = u(a) + h\left[u'(a)\right] + \frac{h^2}{2!}\left[-f(a)u(a)\right] - \frac{h^3}{3!}\left[f(a)u'(a) + u(a)f'(a)\right]$$

$$- \frac{h^4}{4!}\left[f(a)u''(a) + 2u'(a)f'(a) + u(a)f''(a)\right]$$

putting $u(a) = u_0$ $\quad u'(a) = du_0$

$$u(a+h) = u_0 + h\,du_0 - \frac{h^2}{2!}u_0 - \frac{h^3}{3!}f(a)du_0 - \frac{h^3}{3!}u_0 f'(a) - \frac{h^4}{4!}f(a)du_0$$

$$-\frac{h^4}{4!} \times 2du_0 f'(a) - \frac{h^4}{4!}u_0 f''(a)$$

Q. Now Derive algorithm for

$u''(x) + f(x)u(x) = r(x)$ with $u(a) = u_0$ $\quad u(a+h) = u_1$

$u''(x) + k^2(x)u(x) = r(x)$ $\qquad f(x) = k^2(x)$

Taylor Expansion of $u(x+h)$ & $u(x-h)$

$$u(x+h) + u(x-h) = 2u(x) + h^2u^2(x) + \frac{h^4 u^4(x)}{6} + O(h^6)$$

$$u^2(x) = \frac{u(x+h) + u(x-h) - 2u(x)}{h^2} - \frac{h^2}{12}u^4(x)$$

$$u^2(x) + \frac{h^2}{12}u^4(x) = \left(1 + \frac{h^2}{12}\frac{d^2}{dx^2}\right)u^2(x)$$

Date ..../..../........

Applying this operator on differential Equation

$$\left(1+\frac{h^2}{12}\frac{d^2}{dx^2}\right)\left(u''(x)+k^2(x)u(x)-r(x)\right)=0$$

$$u''(x)+k^2(x)u(x)-r(x)+\frac{h^2}{12}\frac{d^2}{dx^2}u''(x)\Big)+\frac{h^2}{12}\frac{d^2}{dx^2}\left(k^2(x)u(x)-r(x)\right)=0$$

$$\frac{d^2}{dx^2}\left(k^2(x)u(x)-r(x)\right)=\frac{(k^2_{i+1}u_{i+1}-r_{i+1})+(k^2_{i-1}u_{i-1}-r_{i-1})-2k^2_i u_i-2r_i}{h^2}+O(h^2)$$

$$\frac{u_{i+1}+u_{i-1}-2u_i}{h^2}+k^2_i u_i-r_i+\frac{h^2}{12}\left[\frac{k^2_{i+1}u_{i+1}-r_{i+1}+k^2_{i-1}u_{i-1}-r_{i-1}-2k^2_i u_i-2r_i}{h^2}\right]=0$$

$$u_{i+1}+u_{i-1}-2u_i+h^2k_i^2u_i-h^2r_i+\frac{h^2}{12}\left[k^2_{i+1}u_{i+1}-r_{i+1}+k^2_{i-1}u_{i-1}-r_{i-1}-2k_i^2u_i-2r_i\right]=0$$

$$u_{i+1}\left[1+\frac{h^2}{12}k^2_{i+1}\right]+u_{i-1}\left[1+\frac{h^2}{12}k^2_{i-1}\right]-2u_i\left[1-\frac{h^2k_i^2}{2}+\frac{h^2k_i^2}{12}\right]$$

$$=\frac{h^2}{12}\left[r_{i+1}+r_{i-1}+10r_i\right]$$

$$u_{i+1}=\frac{\frac{h^2}{12}\left[r_{i+1}+r_{i-1}+10r_i\right]+2u_i\left[1-\frac{h^2k_i^2}{2}+\frac{h^2}{12}k_i^2\right]-u_{i-1}\left[1+\frac{h^2}{12}k^2_{i-1}\right]}{\left[1+\frac{h^2}{12}k^2_{i+1}\right]}$$

Date ..../..../........

Q. Show steps of numerical computation to solve IVP with Numerov Method with N=4

$$u''(x) + (1+x^2)u(x) = 0$$

$$u(0)=1 \quad , u'(0) = 0$$

As. To implement Numerov Method, we require $u(a+h)$, So we will extract $u(a+h)$ ~~using taylo~~ with ~~the~~ the help of $u(0)$ and $u(1)$ using taylor expansion

$$u''(x) = (1+x^2)u(x) \quad -(1)$$

$$u(0)=1 \quad u'(0)=0$$

To find $u(0+h) = ?$

$$u(a+h) = u(a) + h u'(a) + \frac{h^2}{2!}u''(a) + \frac{h^3}{3!}u'''(a) + \frac{h^4}{4!}u''''(a)$$

where $a \to 0$

$$u(0+h) = u(0) + hu'(0) + \frac{h^2}{2!}u''(0) + \frac{h^3}{3!}u'''(0) + \frac{h^4}{4!}u''''(0)$$

From (1)

$$u''(0) = (1+0)u(0) = u(0) = 1$$

$$u'''(x) = \frac{d}{dx}(1+x^2)u(x) = u'(x)(1+x^2) + (2x)u(x)$$

$$u'''(0) = u'(0)(1+0^2) + 2\times 0 = 0$$

$$u''''(x) = \frac{d}{dx}(u'(x)(1+x^2) + 2xu(x)) = u''(x)(1+x^2) + (2x)u'(x) + (2x)u'(x) + 2u(x)$$

$$u^4(0) = u''(0) + 2u(0) = 1 + 2 = 3$$

$$u(0+h) = 1 + h\times 0 + \frac{h^2}{2!}(1) + \frac{h^3}{3!}\times(0) + \frac{h^4}{4!}\times 3$$

$$u(0+h) = 1 + \frac{h^2}{2!} + \frac{h^4}{4!}\times 3$$

Spiral

Date ..../..../.........

(c) ~~Show steps of computation~~

Let's take $x_{min} = 0$ $x_{max} = 1$ $N = 4$

Then $h = \dfrac{x_{max} - x_{min}}{N} = \dfrac{1}{4} = 0.25$

$x_0 = x_{min} = 0$

$x_1 = x_0 + 1 \times h = 0.25$

$x_2 = x_0 + 2 \times h = 0.50$

$x_3 = x_0 + 3 \times h = 0.75$

$x_4 = x_0 + 4 \times h = 1 = x_{max}$

As given $u(0) = 1$ $u(0+h) = 1 + \dfrac{h^2}{2!} \cancel{+ \dfrac{h^3}{3!}} + \dfrac{h^4}{4!} \times 3 \Rightarrow u(0.25) = 1.0317$

As we know,

$$u_{i+1} = \frac{2u_i\left[1 - \dfrac{5h^2 k_i^2}{12}\right] - u_{i-1}\left[1 + \dfrac{h^2 k_{i-1}^2}{12}\right]}{\left[1 + \dfrac{h^2 k_{i+1}^2}{12}\right]}$$

Here i is index, which goes from ~~1~~ 2 to $N+1$ as u_0 and u_1 are given as initial conditions, which will be used to approximate further values---

First, ~~calculate~~ Take $i = 2$

$$u_3 = \frac{2u_2\left[1 - \dfrac{5h^2 k_2^2}{12}\right] - u_1\left[1 + \dfrac{h^2 k_1^2}{12}\right]}{\left[1 + \dfrac{h^2 k_3^2}{12}\right]}$$

Here $k_i^2 = -(1 + x_i^2)$

$u_3 = 1.33157$

Date ..../..../........

$i = 3$

$$y_4 = \frac{2y_3\left[1 - \frac{5h^2 k_3^2}{12}\right] - y_2\left[1 + \frac{h^2 k_2^2}{12}\right]}{\left[1 + \frac{h^2 k_4^2}{12}\right]}$$

$y_4 = 1.324840$

$i = 4$

$$y_5 = \frac{2y_4\left[1 - \frac{5h^2 k_4^2}{12}\right] - y_3\left[1 + \frac{h^2 k_3^2}{12}\right]}{\left[1 + \frac{h^2 k_5^2}{12}\right]}$$

$y_5 = 1.648892$

Programming

```
import math
import numpy as np
import matplotlib.pyplot as plt
from scipy.integrate import solve_ivp
import pandas as pd
def alpha(x):
    return -(1+x**2)
def func_(x,x_vec):
    ans_vec = np.zeros((2))
    ans_vec[0] = x_vec[1]
    ans_vec[1] = (1+x**2)*x_vec[0]
    return ans_vec

def sub_plot(ax,a,b,d,title):
    ax.scatter(a,b,label="Numerical Value")
    ax.plot(a,d,label="Inbuilt Solution")
    ax.set_title(title)
    ax.set_xlabel("x")
    ax.set_ylabel("u")
    ax.legend()
    ax.grid(True)

def numerov(x_min, x_max, u_0, u_prime, N):
    c_i =[];u=[]
    x = np.linspace(x_min,x_max,N+1)
    Alpha = alpha(x)
    h = x[1]-x[0]
    u_1 = 1 + ((h**2)/math.factorial(2)) + 3*((h**4)/math.factorial(4))
    u.append(u_0);u.append(u_1)
    ddx_12 = (h**2)/12
    for i in range(0,N+1):
        c_i_ = 1 + np.multiply(ddx_12,Alpha[i])
        c_i.append(c_i_)
    for i in range(2,N+1):
        u_ = (1/c_i[i])*(((12-10*c_i[i-1])*u[i-1])-c_i[i-2]*u[i-2])
        u.append(u_)
    sol = solve_ivp(func_, [x_min,x_max], [u_0,u_prime],dense_output=True)
    inbuilt = sol.sol(x)
    return x, u, inbuilt[0],c_i
'''-------------------------------------------------N=2-----------------------------------------------'''
p = numerov(0,1,1,0,2)
data = {
    "x":p[0],
    "u_num":p[1],
    "u_inbuilt": p[2],
    "E = u_inbuilt - u_num": abs(p[2]-p[1])
}
print(pd.DataFrame(data))

'''-------------------------------------------------N=4-----------------------------------------------'''
p = numerov(0,1,1,0,4)
data = {
    "x":p[0],
    "u_num":p[1],
    "u_inbuilt": p[2],
    "E = u_inbuilt - u_num": abs(p[2]-p[1])
}
print(pd.DataFrame(data))
'''---------------------------------------------------------------------------------------------------'''
n = np.arange(1,7,1)
# fig2 = plt.figure(figsize=(12,12))
```

```
# fig, (ax1, ax2) = plt.subplots(2)
fig2, ((axx1, axx2), (axx3, axx4), (axx5, axx6)) = plt.subplots(3,2)
fig2.suptitle('u(x) vs x for N intervals')
dict = {'0': axx1,'1': axx2,'2': axx3,'3': axx4,'4': axx5,'5': axx6}
for i in range(1,7):
    p = numerov(0,1,1,0,2**i)
    sub_plot(dict[str(i-1)],p[0],p[1],p[2],f'N={2**i}')
plt.tight_layout()
plt.show()
```

Result and Discussion

```
PS C:\Users\adn19> & C:/Users/adn19/anaconda3/python.exe "d:/Sem 5/Quantum Mechanics/Lab/Assignments/Assignment 4/1140_Preetp
     x     u_num  u_inbuilt  E = u_inbuilt - u_num
0  0.0  1.000000   1.000000               0.000000
1  0.5  1.132812   1.133098               0.000286
2  1.0  1.650221   1.648745               0.001475
      x     u_num  u_inbuilt  E = u_inbuilt - u_num
0  0.00  1.000000   1.000000               0.000000
1  0.25  1.031738   1.031790               0.000051
2  0.50  1.133157   1.133098               0.000059
3  0.75  1.324840   1.324811               0.000029
4  1.00  1.648892   1.648745               0.000147
```

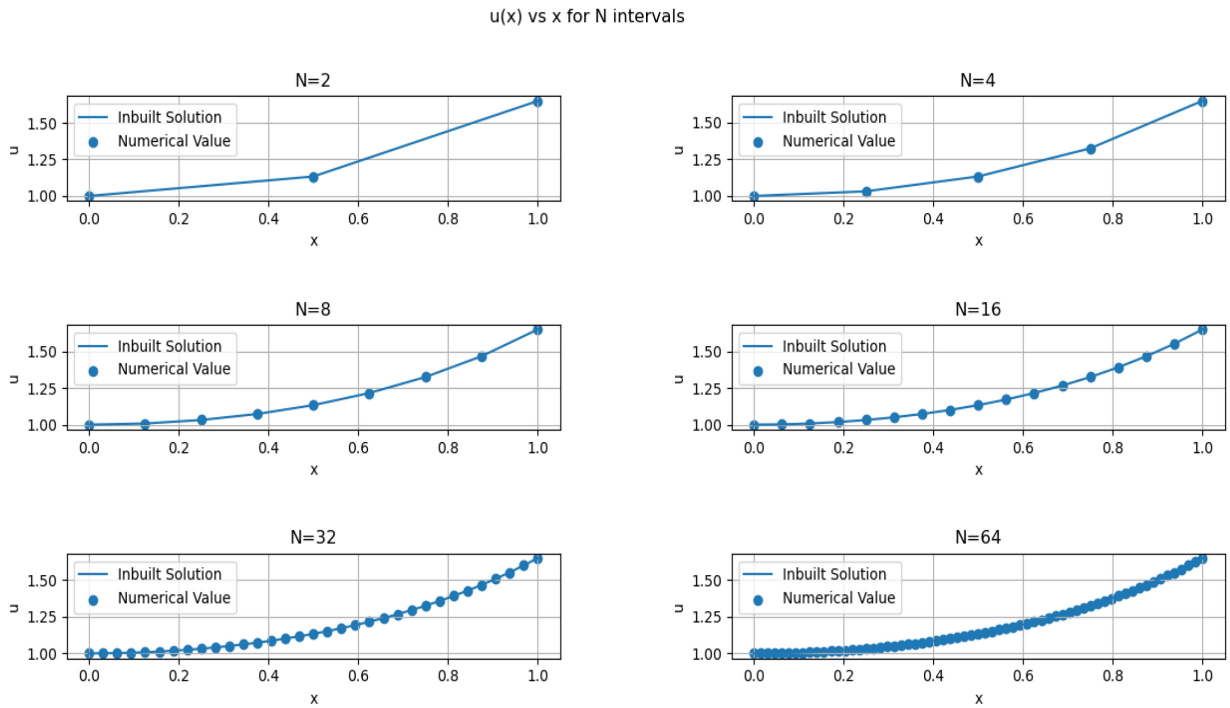

Figure 1: The plot between x and u_x for different number of intervals($N + 1$) shows as the value of N increases the results of Numerov Method matches with more efficient scipy's inbuilt method.