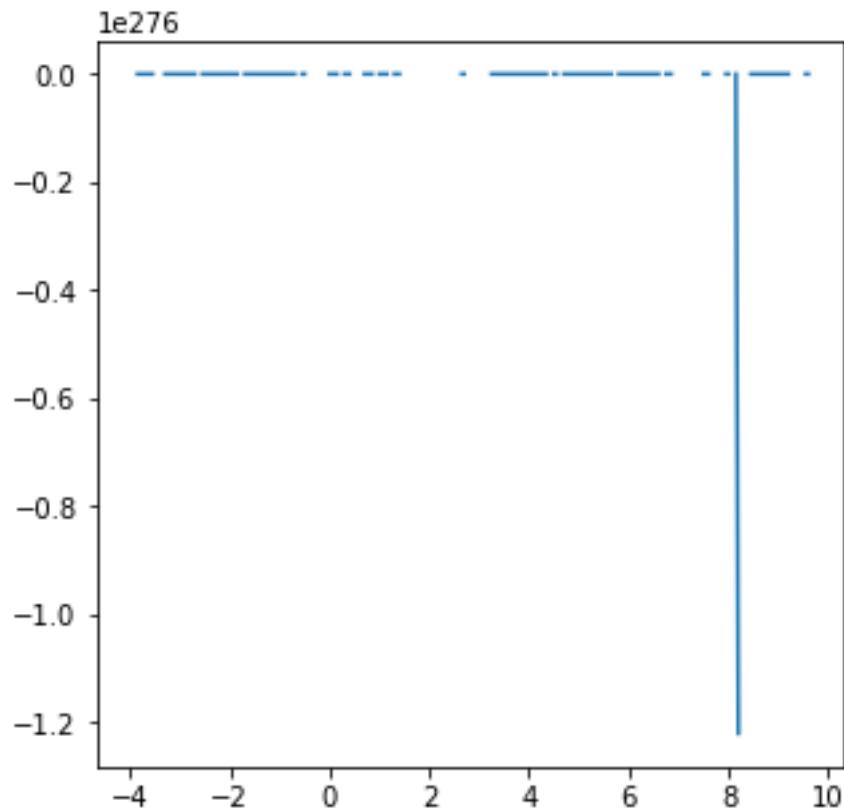


姓名: 徐昶亘 學號: r07525117



Numerical Result @ t = 3.5s

我的結果好奇怪.....，目前還找不到 Bug

```
# -*- coding: utf-8 -*-
"""
Created on Sat May 23 20:24:48 2020
```

```
@author: r07525117
"""
```

```
import numpy as np
import os
os.chdir(r"D:\ShallowWaterComputation\HW6")

g = 9.806
```

```

def eta0(x, H0 = 0.15, h = 0.3):
    K = ( 1.0 / h ) * np.sqrt( (3*H0) / (4*h) )
    ETA = H0 * ( 1.0 / np.cosh(K*x) )**2
    return ETA


def U0( eta, h = 0.3 ):
    return (eta/h)*np.sqrt(g*h)


def MUSCL(u):
    GN = 1
    DelP = u[GN+1:] - u[GN:-GN]
    DelN = u[GN:-GN] - u[GN-1:-GN-1]

    DelBar = ( DelP*abs(DelN) + abs(DelP)*DelN )/( abs(DelP) +
    abs(DelN) )
    DelBar[np.isnan(DelBar)] = 0

    uP = u[GN:-GN] - 0.5*DelBar
    uN = u[GN:-GN] + 0.5*DelBar
    uP = np.concatenate( ( uP[[1]] , uP , uP[[-2]] ) )
    uN = np.concatenate( ( uN[[1]] , uN , uN[[-2]] ) )
    return uP,uN


def HLLC(EtaN, EtaP, UN, UP, h):
    HP = EtaP + h
    HN = EtaN + h
    HU_N = HN*UN
    HU_P = HP*UP

    NegPara = HN*(UN**2) + 0.5*g*(EtaN**2 + 2*EtaN*h)
    PosPara = HP*(UP**2) + 0.5*g*(EtaP**2 + 2*EtaP*h)

    us = 0.5*(UN+UP) + (g*HN)**0.5 - (g*HP)**0.5
    cs = 0.5*( (g*HN)**0.5 + (g*HP)**0.5 ) + 0.25*( UN - UP )

```

```

SN = np.array( [ min(i,j) for i , j in zip( UN -
(g*HN)**0.5 , us - cs ) ] )
SP = np.array( [ max(i,j) for i , j in zip( UP +
(g*HP)**0.5 , us + cs ) ] )
PhiSN = HN*(SN - UN)/(SN - us)*np.array( [[1]*len(us),
list(us)] )
PhiSP = HP*(SP - UP)/(SP - us)*np.array( [[1]*len(us),
list(us)] )
PhiN = np.array( [ HN , HU_N] )
PhiP = np.array( [ HP , HU_P] )

fN = np.array( [ HU_N , NegPara ] )
fP = np.array( [ HU_P , PosPara ] )

f = np.zeros((2,len(SN)))
for i in range( len(SN) ):
    if ( 0 <= SN[i] ):
        f[:,[i]] = fN[:,[i]]
    elif ( SN[i] < 0 <= us[i] ):
        f[:,[i]] = fN[:,[i]] + SN[i]*( PhiSN[:,[i]] -
PhiN[:,[i]] )
    elif ( us[i] < 0 <= SP[i] ):
        f[:,[i]] = fP[:,[i]] + SP[i]*( PhiSP[:,[i]] -
PhiP[:,[i]] )
    elif ( SP[i] < 0 ):
        f[:,[i]] = fP[:,[i]]
return f[0] , f[1]

```

$h = 0.3$

$x_{\text{Max}} = 10$
 $x_{\text{Min}} = -4$
 $\Delta x = 0.05$

$t_{\text{Min}} = 0$
 $t_{\text{Max}} = 3.5$
 $CFL = 0.9$

```

deltaT = CFL*( float(deltaX) / np.sqrt(g*h) )

x = np.arange(xMin-1*deltaX, xMax+1*deltaX, deltaX)
t = np.arange(0, tMax, deltaT)

xlen = len(x)
tlen = len(t)

ETA = np.zeros( ( tlen, xlen) )
U = np.zeros( ( tlen, xlen) )

H = np.zeros( ( tlen, xlen) )
HU = np.zeros( ( tlen, xlen ) )

Hs1 = np.zeros( xlen )
HUs1 = np.zeros( xlen )

Hs2 = np.zeros( xlen )
HUs2 = np.zeros( xlen )

TempEta = np.zeros( xlen )
TempU = np.zeros( xlen )

GN = 1
##Initialization
ETA[0][GN:-GN] = eta0( x[GN:-GN] )
U[0][GN:-GN] = U0( ETA[0][GN:-GN] )
H[0][GN:-GN] = ETA[0][GN:-GN] + h
HU[0][GN:-GN] = H[0][GN:-GN]*U[0][GN:-GN]

##BC
ETA[0][[0,-1]] = ETA[0][[2,-3]]
U[0][[0,-1]] = U[0][[2,-3]]

```

```

n = 0
while( n < (tlen-1) ):
    EtaN, EtaP = MUSCL( ETA[n] )
    UN, UP = MUSCL( U[n] )
    F, G = HLLC(EtaN, EtaP, UN, UP, h)
    #Cell-first round
    Hs1[GN:-GN] = H[n][GN:-GN] - ( (deltaT) /
(deltaX) )*( F[GN+1:] - F[GN:-GN] )
    HUs1[GN:-GN] = HU[n][GN:-GN] - ( deltaT /
deltaX )*( G[GN+1:] - G[GN:-GN] )

    #Recovery eta, u first round
    TempEta[GN:-GN] = Hs1[GN:-GN]-h
    TempU[GN:-GN] = HUs1[GN:-GN]/Hs1[GN:-GN]

    #BC-first round
    Hs1[[0,-1]] = Hs1[[2,-3]]
    HUs1[[0,-1]] = HUs1[[2,-3]]
    TempEta[[0,-1]] = TempEta[[2,-3]]
    TempU[[0,-1]] = TempU[[2,-3]]

    #Cell-Second round
    EtaN, EtaP = MUSCL( TempEta )
    UN, UP = MUSCL( TempU )
    F, G = HLLC(EtaN, EtaP, UN, UP, h)
    Hs2[GN:-GN] = (3.0/4.0)*H[n][GN:-GN] + (1.0/4.0)*Hs1[GN:-
GN]\-
-( deltaT/(4.0*deltaX) )*( F[GN+1:] - F[GN:-GN] )

    HUs2[GN:-GN] = (3.0/4.0)*HU[n][GN:-GN] +
(1.0/4.0)*HUs1[GN:-GN]\-
- ( (deltaT) / (4.0*deltaX) )*( G[GN+1:] - G[GN:-GN] )

    #Recovery eta, u Second round
    TempEta[GN:-GN] = Hs2[GN:-GN]-h
    TempU[GN:-GN] = HUs2[GN:-GN]/Hs2[GN:-GN]

    #BC-Second round

```

```

Hs2[[0,-1]] = Hs2[[2,-3]]
HUs2[[0,-1]] = HUs2[[2,-3]]
TempEta[[0,-1]] = TempEta[[2,-3]]
TempU[[0,-1]] = TempU[[2,-3]]

#Cell-Third round
EtaN, EtaP = MUSCL( TempEta )
UN, UP = MUSCL( TempU )
F, G = HLLC(EtaN, EtaP, UN, UP, h)
H[n+1][GN:-GN] = (1.0/3.0)*H[n][GN:-GN] +
(2.0/3.0)*Hs2[GN:-GN]\-
- ( (2*deltaT) / (3.0*deltaX) )*( F[GN+1:] - F[GN:-GN] )

HU[n+1][GN:-GN] = (1.0/3.0)*HU[n][GN:-GN] +
(2.0/3.0)*HUs2[GN:-GN]\-
- ( (2*deltaT)/(3.0*deltaX) )*( G[GN+1:] - G[GN:-GN] )

#Recovery eta, u Second round
ETA[n+1][GN:-GN] = H[n+1][GN:-GN]-h
U[n+1][GN:-GN] = HU[n+1][GN:-GN]/H[n+1][GN:-GN]

#BC-Third round
H[n+1][[0,-1]] = H[n+1][[2,-3]]
HU[n+1][[0,-1]] = HU[n+1][[2,-3]]
ETA[n+1][[0,-1]] = ETA[n+1][[2,-3]]
U[n+1][[0,-1]] = U[n+1][[2,-3]]
n = n+1

```

```

import matplotlib.pyplot as plt
plt.figure(figsize=(5,5))
plt.plot( x , ETA[3])

```