D3Q15_General_diffusion_solver_Dirichlet

December 5, 2023

[2]:
```
import numpy as np
import matplotlib.pyplot as plt
```

[4]:
```
##physical dimension of the cube
Lx=1
Ly=1
Lz=1
##Number of grid points in z -direction
Nz=50
##Number of grid points in x-direction
Nx=50
##Number of grid points in y-direction
Ny=50
##grid spacing
dx=Lx/(Nx-1)
dy=Ly/(Ny-1)
dz=Lz/(Nz-1)
##Creating grids
x=np.linspace(0,Lx,Nx)
y=np.linspace(0,Ly,Ny)
z=np.linspace(0,Lz,Nz)
[Z,X,Y]=np.meshgrid(z,x,y)
##Weights of digital particles
##15 digital particles
w=np.zeros(15)
##For rest digital particle
w[0]=16/72
##For digital particle on center of faces
for i in np.arange(1,7):
    w[i]=8/72
##For digital particle on corners
for i in np.arange(7,15):
    w[i]=1/72
##Macroscopic property temperature which is to be predicted
T=np.zeros((Nz,Ny,Nx))
##Boundary conditions
##Top face boundary
##Temperature at top boundary
```

```
T_top=0
T[Nz-1,:,:]=T_top
##Bottom face boundary
##Temperature at bottom boundary
T_bot=1
T[0,:,:]=T_bot
##North face boundary
##Temperature at north boundary
Tn=0
T[1:Nz-1,0,:]=Tn
##South face boundary
##Temperature at south boundary
Ts=0
T[1:Nz-1,Ny-1,:]=Ts
##West face boundary
##Temperature at west boundary
Tw=0
T[1:Nz-1,1:Ny-1,0]=Tw
##East face boundary
##Temperature at east boundary
Te=0
T[1:Nz-1,1:Ny-1,Nx-1]=Te
##Initial condition
##Temperature specified at internal of domain
##Initial temperature of domain i.e at t=0
Ti=0
T[1:Nz-1,1:Nx-1,1:Nx-1]=Ti
##Initialise equilibrium particle distribution function
f0eq=np.zeros((Nz,Ny,Nx))
f1eq=np.zeros((Nz,Ny,Nx))
f2eq=np.zeros((Nz,Ny,Nx))
f3eq=np.zeros((Nz,Ny,Nx))
f4eq=np.zeros((Nz,Ny,Nx))
f5eq=np.zeros((Nz,Ny,Nx))
f6eq=np.zeros((Nz,Ny,Nx))
f7eq=np.zeros((Nz,Ny,Nx))
f8eq=np.zeros((Nz,Ny,Nx))
f9eq=np.zeros((Nz,Ny,Nx))
f10eq=np.zeros((Nz,Ny,Nx))
f11eq=np.zeros((Nz,Ny,Nx))
f12eq=np.zeros((Nz,Ny,Nx))
f13eq=np.zeros((Nz,Ny,Nx))
f14eq=np.zeros((Nz,Ny,Nx))
##Computing equilibrium distribution function
f0eq=w[0]*T
f1eq=w[1]*T
f2eq=w[2]*T
```

```
f3eq=w[3]*T
f4eq=w[4]*T
f5eq=w[5]*T
f6eq=w[6]*T
f7eq=w[7]*T
f8eq=w[8]*T
f9eq=w[9]*T
f10eq=w[10]*T
f11eq=w[11]*T
f12eq=w[12]*T
f13eq=w[13]*T
f14eq=w[14]*T
##Initiliase particle distribution function
##D3Q15
f0=np.zeros((Nz,Ny,Nx))
f1=np.zeros((Nz,Ny,Nx))
f2=np.zeros((Nz,Ny,Nx))
f3=np.zeros((Nz,Ny,Nx))
f4=np.zeros((Nz,Ny,Nx))
f5=np.zeros((Nz,Ny,Nx))
f6=np.zeros((Nz,Ny,Nx))
f7=np.zeros((Nz,Ny,Nx))
f8=np.zeros((Nz,Ny,Nx))
f9=np.zeros((Nz,Ny,Nx))
f10=np.zeros((Nz,Ny,Nx))
f11=np.zeros((Nz,Ny,Nx))
f12=np.zeros((Nz,Ny,Nx))
f13=np.zeros((Nz,Ny,Nx))
f14=np.zeros((Nz,Ny,Nx))
##Initial value of particle distribution function
f0=f0eq
f1=f1eq
f2=f2eq
f3=f3eq
f4=f4eq
f5=f5eq
f6=f6eq
f7=f7eq
f8=f8eq
f9=f9eq
f10=f10eq
f11=f11eq
f12=f12eq
f13=f13eq
f14=f14eq
##Collision of digital particles
##collision parameter
```

```
omega=1.2
##Initiliase post collision values of digital particles
f0c=np.zeros((Nz,Ny,Nx))
f1c=np.zeros((Nz,Ny,Nx))
f2c=np.zeros((Nz,Ny,Nx))
f3c=np.zeros((Nz,Ny,Nx))
f4c=np.zeros((Nz,Ny,Nx))
f5c=np.zeros((Nz,Ny,Nx))
f6c=np.zeros((Nz,Ny,Nx))
f7c=np.zeros((Nz,Ny,Nx))
f8c=np.zeros((Nz,Ny,Nx))
f9c=np.zeros((Nz,Ny,Nx))
f10c=np.zeros((Nz,Ny,Nx))
f11c=np.zeros((Nz,Ny,Nx))
f12c=np.zeros((Nz,Ny,Nx))
f13c=np.zeros((Nz,Ny,Nx))
f14c=np.zeros((Nz,Ny,Nx))
##Collision of digital particles
f0c=(1-omega)*f0+(omega*f0eq)
f1c=(1-omega)*f1+(omega*f1eq)
f2c=(1-omega)*f2+(omega*f2eq)
f3c=(1-omega)*f3+(omega*f3eq)
f4c=(1-omega)*f4+(omega*f4eq)
f5c=(1-omega)*f5+(omega*f5eq)
f6c=(1-omega)*f6+(omega*f6eq)
f7c=(1-omega)*f7+(omega*f7eq)
f8c=(1-omega)*f8+(omega*f8eq)
f9c=(1-omega)*f9+(omega*f9eq)
f10c=(1-omega)*f10+(omega*f10eq)
f11c=(1-omega)*f11+(omega*f11eq)
f12c=(1-omega)*f12+(omega*f12eq)
f13c=(1-omega)*f13+(omega*f13eq)
f14c=(1-omega)*f14+(omega*f14eq)
##Streaming of digital particles

##particle number 0 is at rest
f0=f0c
##digital particle numbered from 1,2,3,...6 located on faces

##particle number 3 which is moving upwards towards top boundary
for k in np.arange(0,Nz-1):
    for i in np.arange(0,Ny):
        for j in np.arange(0,Nx):
            f3[k+1,i,j]=f3c[k,i,j]

##particle number 4 which is moving downwards towards bottom boundary
for k in np.arange(0,Nz-1):
```

```
    for i in np.arange(0,Ny):
        for j in np.arange(0,Nx):
            f4[k,i,j]=f4c[k+1,i,j]

##digital particle 1 which moves on x-y plane with unit velocity in x-direction
for k in np.arange(0,Nz):
    for i in np.arange(0,Ny):
        for j in np.arange(0,Nx-1):
            f1[k,i,j+1]=f1c[k,i,j]

##digital particle 2 moves on x-y plane in negative x-direction
for k in np.arange(0,Nz):
    for i in np.arange(0,Ny):
        for j in np.arange(0,Nx-1):
            f2[k,i,j]=f2c[k,i,j+1]

##digital particle 6 which moves on x-y plane upwards in positive y -direction
for k in np.arange(0,Nz):
    for j in np.arange(0,Nx):
        for i in np.arange(0,Ny-1):
            f6[k,i,j]=f6c[k,i+1,j]

##digital particle 5 which moves on x-y plane upwards in negative y -direction
for k in np.arange(0,Nz):
    for j in np.arange(0,Nx):
        for i in np.arange(0,Ny-1):
            f5[k,i+1,j]=f5c[k,i,j]

##digital particle 7
for k in np.arange(0,Nz-1):
    for i in np.arange(0,Ny-1):
        for j in np.arange(0,Nx-1):
            f7[k+1,i+1,j+1]=f7c[k,i,j]

##digital particle 12
for k in np.arange(0,Nz-1):
    for i in np.arange(0,Ny-1):
        for j in np.arange(0,Nx-1):
            f12[k+1,i,j]=f12c[k,i+1,j+1]


##digital particle 9
for k in np.arange(0,Nz-1):
    for i in np.arange(0,Ny-1):
        for j in np.arange(0,Nx-1):
            f9[k+1,i,j+1]=f9c[k,i+1,j]
```

```
##digital particle 14
for k in np.arange(0,Nz-1):
    for i in np.arange(0,Ny-1):
        for j in np.arange(0,Nx-1):
            f14[k+1,i+1,j]=f14c[k,i,j+1]

##digital particle 11
for k in np.arange(0,Nz-1):
    for i in np.arange(0,Ny-1):
        for j in np.arange(0,Nx-1):
            f11[k,i+1,j+1]=f11c[k+1,i,j]

##Digital particle 8
for k in np.arange(0,Nz-1):
    for i in np.arange(0,Ny-1):
        for j in np.arange(0,Nx-1):
            f8[k,i,j]=f8c[k+1,i+1,j+1]

##digital particle 13
for k in np.arange(0,Nz-1):
    for i in np.arange(0,Ny-1):
        for j in np.arange(0,Nx-1):
            f13[k,i,j+1]=f13c[k+1,i+1,j]

##digital particle 10
for k in np.arange(0,Nz-1):
    for i in np.arange(0,Ny-1):
        for j in np.arange(0,Nx-1):
            f10[k,i+1,j]=f10c[k+1,i,j+1]


##Boundary conditions

##West boundaries
##unknowns are f1 f7 f9 f11 f13
for k in np.arange(1,Nz-1):
    j=0
    for i in np.arange(1,Ny-1):
        f1[k,i,j]=w[1]*Tw+ w[2]*Tw-f2[k,i,j]
        f7[k,i,j]=w[7]*Tw+w[8]*Tw-f8[k,i,j]
        f9[k,i,j]=w[9]*Tw+w[10]*Tw-f10[k,i,j]
        f11[k,i,j]=w[11]*Tw+w[12]*Tw-f12[k,i,j]
      ␣
↪f13[k,i,j]=Tw*(1-w[1]-w[2]-w[7]-w[8]-w[9]-w[10]-w[11]-w[12])-(f0[k,i,j]+f3[k,i,j]+f4[k,i,j]


##East boundary
```

```
##unkown df are f2,f8,f10,f12,f14
for k in np.arange(1,Nz-1):
    j=Nx-1
    for i in np.arange(1,Ny-1):
        f2[k,i,j]=w[1]*Te+ w[2]*Te-f1[k,i,j]
        f8[k,i,j]=w[7]*Te+w[8]*Te-f7[k,i,j]
        f10[k,i,j]=w[9]*Te+w[10]*Te-f9[k,i,j]
        f12[k,i,j]=w[11]*Te+w[12]*Te-f11[k,i,j]
       ␣
 ↪f14[k,i,j]=Te*(1-w[1]-w[2]-w[7]-w[8]-w[9]-w[10]-w[11]-w[12])-(f0[k,i,j]+f3[k,i,j]+f4[k,i,j]

##North boundary

##Interior lattice nodes
##unknown df are f5 ,f7 ,f10 , f11 , f14
for k in np.arange(1,Nz-1):
    i=0
    for j in np.arange(1,Nx-1):
        f5[k,i,j]=w[5]*Tn +w[6]*Tn-f6[k,i,j]
        f7[k,i,j]=w[7]*Tn+w[8]*Tn-f8[k,i,j]
        f10[k,i,j]=w[9]*Tn+w[10]*Tn-f9[k,i,j]
        f11[k,i,j]=w[11]*Tn+w[12]*Tn-f12[k,i,j]
       ␣
 ↪f14[k,i,j]=Tn*(1-w[5]-w[6]-w[7]-w[8]-w[9]-w[10]-w[11]-w[12])-(f0[k,i,j]+f1[k,i,j]+f2[k,i,j]

##corner lattice nodes
##Left corner lattice node
##f1,f5,f7,f9,f10,f11,f13,f14
for k in np.arange(1,Nz-1):
    i=0
    j=0
    f1[k,i,j]=w[1]*Tn+w[2]*Tn-f2[k,i,j]
    f5[k,i,j]=w[5]*Tn +w[6]*Tn-f6[k,i,j]
    f7[k,i,j]=w[7]*Tn+w[8]*Tn-f8[k,i,j]
    f11[k,i,j]=w[11]*Tn+w[12]*Tn-f12[k,i,j]
    f9[k,i,j]=0.5*((18/72)*Tn-f0[k,i,j]-f3[k,i,j])
    f10[k,i,j]=f9[k,i,j]
    f13[k,i,j]=0.5*((18/72)*Tn-f4[k,i,j])
    f14[k,i,j]=f13[k,i,j]
##Right corner lattice node
##f2,f5,f7,f8,f10,f11,f12,f14
for k in np.arange(1,Nz-1):
    i=0
    j=Nx-1
    f2[k,i,j]=w[1]*Tn+w[2]*Tn-f1[k,i,j]
    f5[k,i,j]=w[5]*Tn +w[6]*Tn-f6[k,i,j]
    f10[k,i,j]=w[9]*Tn +w[10]*Tn-f9[k,i,j]
```

```
    f14[k,i,j]=w[13]*Tn+w[14]*Tn-f13[k,i,j]
    f7[k,i,j]=0.5*((18/72)*Tn-f0[k,i,j]-f3[k,i,j])
    f8[k,i,j]=f7[1,0,5]
    f11[k,i,j]=0.5*((18/72)*Tn-f4[k,i,j])
    f12[k,i,j]=f11[k,i,j]


##South boundary

##Interior lattice nodes
##unknown vdf f6,f8,f9,f12,f13
for k in np.arange(1,Nz-1):
    i=Ny-1
    for j in np.arange(1,Nx-1):
        f6[k,i,j]=w[5]*Ts +w[6]*Ts-f5[k,i,j]
        f8[k,i,j]=w[7]*Ts+w[8]*Ts-f7[k,i,j]
        f9[k,i,j]=w[9]*Ts+w[10]*Ts-f10[k,i,j]
        f12[k,i,j]=w[11]*Ts+w[12]*Ts-f11[k,i,j]
        ␣
 ↪f13[k,i,j]=Ts*(1-w[5]-w[6]-w[7]-w[8]-w[9]-w[10]-w[11]-w[12])-(f0[k,i,j]+f1[k,i,j]+f2[k,i,j]

##corner lattice nodes
##Left corner node
##f1,f6,f7,f8,f9,f11,f12,f13
for k in np.arange(1,Nz-1):
    i=Ny-1
    j=0
    f1[k,i,j]=w[1]*Ts+w[2]*Ts-f2[k,i,j]
    f6[k,i,j]=w[5]*Ts +w[6]*Ts-f5[k,i,j]
    f9[k,i,j]=w[9]*Ts +w[10]*Ts-f10[k,i,j]
    f13[k,i,j]=w[13]*Ts+w[14]*Ts-f14[k,i,j]
    f7[k,i,j]=0.5*((18/72)*Ts-f0[k,i,j]-f3[k,i,j])
    f8[k,i,j]=f7[k,i,j]
    f11[k,i,j]=0.5*((18/72)*Ts-f4[k,i,j])
    f12[k,i,j]=f11[k,i,j]

##Right corner node
##f2,f6,f8,f9,f10,f12,f13,f14
for k in np.arange(1,Nz-1):
    i=Ny-1
    j=Nx-1
    f2[k,i,j]=w[1]*Ts+w[2]*Ts-f1[k,i,j]
    f6[k,i,j]=w[5]*Ts +w[6]*Ts-f5[k,i,j]
    f8[k,i,j]=w[7]*Ts+w[8]*Ts-f7[k,i,j]
    f12[k,i,j]=w[11]*Ts+w[12]*Ts-f11[k,i,j]
    f9[k,i,j]=0.5*((18/72)*Ts-f0[k,i,j]-f3[k,i,j])
    f10[k,i,j]=f9[k,i,j]
```

```
    f13[k,i,j]=0.5*((18/72)*Ts-f4[k,i,j])
    f14[k,i,j]=f13[k,i,j]

##Top boundary

##Inner lattice nodes
##Unknown distribution functions are f4,f8,f10,f11,,f13
k=Nz-1
for i in np.arange(1,Ny-1):
    for j in np.arange(1,Nx-1):
        f4[k,i,j]=w[4]*T_top +w[3]*T_top-f3[k,i,j]
        f8[k,i,j]=w[7]*T_top+w[8]*T_top-f7[k,i,j]
        f10[k,i,j]=w[9]*T_top+w[10]*T_top-f9[k,i,j]
        f11[k,i,j]=w[11]*T_top+w[12]*T_top-f12[k,i,j]
      ␣
 ↪f13[k,i,j]=T_top*(1-w[5]-w[6]-w[7]-w[8]-w[9]-w[10]-w[11]-w[12])-(f0[k,i,j]+f1[k,i,j]+f2[k,i



##Top north
##Interior
##unknownf4,f5,f7,f8,f10,f11,f13,f14
k=Nz-1
i=0
for j in np.arange(1,Nx-1):
    f4[k,i,j]=w[4]*T_top +w[3]*T_top-f3[k,i,j]
    f5[k,i,j]=w[5]*T_top +w[6]*T_top-f6[k,i,j]
    f10[k,i,j]=w[9]*T_top+w[10]*T_top-f9[k,i,j]
    f11[k,i,j]=w[11]*T_top+w[12]*T_top-f12[k,i,j]
    f7[k,i,j]=0.5*((18/72)*T_top-f0[k,0,1]-f1[k,i,j])
    f8[k,i,j]=f7[5,0,1]
    f13[k,i,j]=0.5*((18/72)*T_top-f2[k,i,j])
    f14[k,i,j]=f13[k,i,j]

##Left corner north lattice node
##unknown df are f1,f4,f5,f7,f8,f9,f10,f11,f13,f14
k=Nz-1
i=0
j=0
f1[k,i,j]=w[1]*T_top+w[2]*T_top-f2[k,i,j]
f4[k,i,j]=w[4]*T_top +w[3]*T_top-f3[k,i,j]
f5[k,i,j]=w[5]*T_top +w[6]*T_top-f6[k,i,j]
f11[k,i,j]=w[11]*T_top+w[12]*T_top-f12[k,i,j]
f7[k,i,j]=0.5*((10/72)*T_top-f0[k,i,j])
f8[k,i,j]=f7[k,i,j]
f9[k,i,j]=0.5*((6/72)*T_top)
f10[k,i,j]=f9[k,i,j]
```

```
f13[k,i,j]=0.5*((6/72)*T_top)
f14[k,i,j]=f13[k,i,j]

##Right corner north lattice node
##unknown df are f2,f4,f5,f7,f8,f10,f11,f13,f14
k=Nz-1
i=0
j=5
f2[k,i,j]=w[1]*T_top+w[2]*T_top-f1[k,i,j]
f4[k,i,j]=w[4]*T_top +w[3]*T_top-f3[k,i,j]
f5[k,i,j]=w[5]*T_top +w[6]*T_top-f6[k,i,j]
f10[k,i,j]=w[9]*T_top+w[10]*T_top-f9[k,i,j]
f7[k,i,j]=0.5*((10/72)*T_top-f0[k,i,j])
f8[k,i,j]=f7[k,i,j]
f11[k,i,j]=0.5*((6/72)*T_top)
f12[k,i,j]=f11[k,i,j]
f13[k,i,j]=0.5*((6/72)*T_top)
f14[k,i,j]=f13[k,i,j]


##Top south
##unknowns f4,f6,f8,f9,f10,f11,f12,f13
k=Nz-1
i=Ny-1
for j in np.arange(1,Nx-1):
    f4[k,i,j]=w[4]*T_top +w[3]*T_top-f3[k,i,j]
    f6[k,i,j]=w[5]*T_top +w[6]*T_top-f5[k,i,j]
    f8[k,i,j]=w[7]*T_top+w[8]*T_top-f7[k,i,j]
    f13[k,i,j]=w[14]*T_top+w[13]*T_top-f14[k,i,j]
    f9[k,i,j]=0.5*((18/72)*T_top-f0[k,i,j]-f1[k,i,j])
    f10[k,i,j]=f9[k,i,j]
    f11[k,i,j]=0.5*((18/72)*T_top-f2[k,i,j])
    f12[k,i,j]=f11[k,i,j]

##Left corner lattice node
##unknown df are f1,f4,f6,f7,f8,f9,f10,f11,f12,f13
k=Nz-1
i=Ny-1
j=0
f1[k,i,j]=w[1]*T_top+w[2]*T_top-f2[k,i,j]
f4[k,i,j]=w[4]*T_top +w[3]*T_top-f3[k,i,j]
f6[k,i,j]=w[5]*T_top +w[6]*T_top-f5[k,i,j]
f13[k,i,j]=w[14]*T_top+w[13]*T_top-f14[k,i,j]
f7[k,i,j]=0.5*((10/72)*T_top-f0[k,5,0])
f8[k,i,j]=f7[k,i,j]
f9[k,i,j]=0.5*((6/72)*T_top)
f10[k,i,j]=f9[k,i,j]
```

```
f11[k,i,j]=0.5*((6/72)*T_top)
f12[k,i,j]=f11[k,i,j]

##Right corner lattice node
##unknown df are f2,f4,f6,f8,f9,f10,f11,f12,f13,f14
k=Nz-1
i=Ny-1
j=Nx-1
f2[k,i,j]=w[1]*T_top+w[2]*T_top-f1[k,i,j]
f4[k,i,j]=w[4]*T_top +w[3]*T_top-f3[k,i,j]
f6[k,i,j]=w[5]*T_top +w[6]*T_top-f5[k,i,j]
f8[k,i,j]=w[7]*T_top+w[8]*T_top-f7[k,i,j]
f9[k,i,j]=0.5*((10/72)*T_top-f0[k,i,j])
f10[k,i,j]=f9[k,i,j]
f11[k,i,j]=0.5*((6/72)*T_top)
f12[k,i,j]=f11[k,i,j]
f13[k,i,j]=0.5*((6/72)*T_top)
f14[k,i,j]=f13[k,i,j]

##Top west
##unknown distribution functions
##f1,f4,f7,f8,f9,f10,f11,f13
k=Nz-1
j=0
for i in np.arange(1,Ny-1):
    f1[k,i,j]=w[1]*T_top+w[2]*T_top-f2[k,i,j]
    f4[k,i,j]=w[4]*T_top +w[3]*T_top-f3[k,i,j]
    f11[k,i,j]=w[11]*T_top+w[12]*T_top-f12[k,i,j]
    f13[k,i,j]=w[14]*T_top+w[13]*T_top-f14[k,i,j]
    f7[k,i,j]=0.5*((18/72)*T_top-f0[k,i,j]-f5[k,i,j])
    f8[k,i,j]=f7[k,i,j]
    f9[k,i,j]=0.5*((18/72)*T_top-f6[5,1,0])
    f10[k,i,j]=f9[k,i,j]


##Top East
##unknown df f2,f4,f8,f10,f11,f12,f13,f14
k=Nz-1
j=Ny-1
for i in np.arange(1,Nx-1):
    f2[k,i,j]=w[1]*T_top+w[2]*T_top-f1[k,i,j]
    f4[k,i,j]=w[4]*T_top +w[3]*T_top-f3[k,i,j]
    f8[k,i,j]=w[7]*T_top+w[8]*T_top-f7[k,i,j]
    f10[k,i,j]=w[9]*T_top+w[10]*T_top-f9[k,i,j]
    f11[k,i,j]=0.5*((18/72)*T_top-f0[k,i,j]-f5[k,i,j])
    f12[k,i,j]=f11[5,1,5]
    f13[k,i,j]=0.5*((18/72)*T_top-f6[k,i,j])
```

```
    f14[k,i,j]=f13[k,i,j]



##Bottom boundary

##Inner lattice nodes
##unknown distribution functions are f3,f7,f9,f12,f14
k=0
for i in np.arange(1,Ny-1):
    for j in np.arange(1,Nx-1):
        f3[k,i,j]=w[4]*T_bot +w[3]*T_bot-f4[k,i,j]
        f7[k,i,j]=w[7]*T_bot+w[8]*T_bot-f8[k,i,j]
        f9[k,i,j]=w[9]*T_bot+w[10]*T_bot-f10[k,i,j]
        f12[k,i,j]=w[11]*T_bot+w[12]*T_bot-f11[k,i,j]
        ␣
 ↪f14[k,i,j]=T_bot*(1-w[5]-w[6]-w[7]-w[8]-w[9]-w[10]-w[11]-w[12])-(f0[k,i,j]+f1[k,i,j]+f2[k,i


##Bottom north
##Interior lattice nods
##unknown distribution functions are f3,f5,f7,f9,f10,f11,f12,f14
k=0
i=0
for j in np.arange(1,Nx-1):
    f3[k,i,j]=w[4]*T_bot +w[3]*T_bot-f4[k,i,j]
    f5[k,i,j]=w[5]*T_bot +w[6]*T_bot-f6[k,i,j]
    f7[k,i,j]=w[7]*T_bot+ w[8]*T_bot-f8[k,i,j]
    f14[k,i,j]=w[14]*T_bot+w[13]*T_bot-f13[k,i,j]
    f9[k,i,j]=0.5*((18/72)*T_bot-f0[k,i,j]-f1[k,i,j])
    f10[k,i,j]=f9[k,i,j]
    f11[k,i,j]=0.5*((18/72)*T_bot-f2[k,i,j])
    f12[k,i,j]=f11[k,i,j]

##left corner lattice node
##unknown df is f1, f3,f5,f7,f9,f10,f11,f12,f13,f14
k=0
i=0
j=0
f1[k,i,j]=w[1]*T_bot+w[2]*T_bot-f2[k,i,j]
f3[k,i,j]=w[4]*T_bot +w[3]*T_bot-f4[k,i,j]
f5[k,i,j]=w[5]*T_bot +w[6]*T_bot-f6[k,i,j]
f7[k,i,j]=w[7]*T_bot+w[8]*T_bot-f8[k,i,j]
f9[k,i,j]=0.5*((10/72)*T_bot-f0[k,i,j])
f10[k,i,j]=f9[k,i,j]
f11[k,i,j]=0.5*((6/72)*T_bot)
f12[k,i,j]=f11[k,i,j]
```

```
f13[k,i,j]=0.5*((6/72)*T_bot)
f14[k,i,j]=f13[k,i,j]

##right corner lattice node
##unknown df is f2,f3,f5,f7,f8,f9,f10,f11,f12,f14
k=0
i=0
j=Nx-1
f2[k,i,j]=w[1]*T_bot+w[2]*T_bot-f1[k,i,j]
f3[k,i,j]=w[4]*T_bot +w[3]*T_bot-f4[k,i,j]
f5[k,i,j]=w[5]*T_bot +w[6]*T_bot-f6[k,i,j]
f14[k,i,j]=w[11]*T_bot+w[12]*T_bot-f13[k,i,j]
f7[k,i,j]=0.5*((10/72)*T_bot-f0[k,i,j])
f8[k,i,j]=f7[k,i,j]
f9[k,i,j]=0.5*((6/72)*T_bot)
f10[k,i,j]=f9[k,i,j]
f11[k,i,j]=0.5*((6/72)*T_bot)
f12[k,i,j]=f11[k,i,j]


##Bottom south

##Interior lattice nodes
##f3,f6,f7,f8,f9,f12,f13,f14
k=0
i=Ny-1
for j in np.arange(1,Nx-1):
    f3[k,i,j]=w[4]*T_bot +w[3]*T_bot-f4[k,i,j]
    f6[k,i,j]=w[5]*T_bot +w[6]*T_bot-f5[k,i,j]
    f9[k,i,j]=w[9]*T_bot+w[10]*T_bot-f10[k,i,j]
    f12[k,i,j]=w[11]*T_bot+w[12]*T_bot-f11[k,i,j]
    f7[k,i,j]=0.5*((18/72)*T_bot-f0[0,5,1]-f1[k,i,j])
    f8[k,i,j]=f7[k,i,j]
    f13[k,i,j]=0.5*((18/72)*T_bot-f2[k,i,j])
    f14[k,i,j]=f13[k,i,j]


##Bottom south left corner lattice node
##unknown vdf is f1,f3,f6,f7,f8,f9,f11,f12,f13,f14
k=0
i=Ny-1
j=0
f1[k,i,j]=w[1]*T_bot+w[2]*T_bot-f2[k,i,j]
f3[k,i,j]=w[4]*T_bot +w[3]*T_bot-f4[k,i,j]
f6[k,i,j]=w[5]*T_bot +w[6]*T_bot-f5[k,i,j]
f9[k,i,j]=w[9]*T_bot+w[10]*T_bot-f10[k,i,j]
f7[k,i,j]=0.5*((10/72)*T_bot-f0[k,5,0])
```

```
f8[k,i,j]=f7[k,i,j]
f11[k,i,j]=0.5*((6/72)*T_bot)
f12[k,i,j]=f11[k,i,j]
f13[k,i,j]=0.5*((6/72)*T_bot)
f14[k,i,j]=f13[k,i,j]
##Bottom south right corner lattice node
##unknown df are f2,f3,f6,f7,f8,f9,f10,f12,f13,f14
k=0
i=Ny-1
j=Nx-1
f1[k,i,j]=w[1]*T_bot+w[2]*T_bot-f2[k,i,j]
f3[k,i,j]=w[4]*T_bot +w[3]*T_bot-f4[k,i,j]
f6[k,i,j]=w[5]*T_bot +w[6]*T_bot-f5[k,i,j]
f12[k,i,j]=w[11]*T_bot+w[12]*T_bot-f11[k,i,j]
f7[k,i,j]=0.5*((10/72)*T_bot-f0[k,i,j])
f8[k,i,j]=f7[k,i,j]
f9[k,i,j]=0.5*((6/72)*T_bot)
f10[k,i,j]=f9[k,i,j]
f13[k,i,j]=0.5*((6/72)*T_bot)
f14[k,i,j]=f13[k,i,j]

##Bottom west
##f1,f3,f7,f9,f11,f12,f13,f14
k=0
j=0
for i in np.arange(1,Ny-1):
    f1[k,i,j]=w[1]*T_bot+w[2]*T_bot-f2[k,i,j]
    f3[k,i,j]=w[4]*T_bot +w[3]*T_bot-f4[k,i,j]
    f7[k,i,j]=w[7]*T_bot+w[8]*T_bot-f8[k,i,j]
    f9[k,i,j]=w[9]*T_bot+w[10]*T_bot-f10[k,i,j]
    f11[k,i,j]=0.5*((18/72)*T_bot-f0[k,i,j]-f5[k,i,j])
    f12[k,i,j]=f11[k,i,j]
    f13[k,i,j]=0.5*((18/72)*T_bot-f6[k,i,j])
    f14[k,i,j]=f13[k,i,j]



##Bottom east
#f2,f3,f7,f8,f9,f10,f12,f14
k=0
j=Nx-1
for i in np.arange(1,Ny-1):
    f2[k,i,j]=w[1]*T_bot+w[2]*T_bot-f1[k,i,j]
    f3[k,i,j]=w[4]*T_bot +w[3]*T_bot-f4[k,i,j]
    f12[k,i,j]=w[11]*T_bot+w[12]*T_bot-f11[k,i,j]
    f14[k,i,j]=w[13]*T_bot+w[14]*T_bot-f13[k,i,j]
    f7[k,i,j]=0.5*((18/72)*T_bot-f0[k,1,5]-f5[k,i,j])
```

```
    f8[k,i,j]=f7[k,i,j]
    f9[k,i,j]=0.5*((18/72)*T_bot-f6[0,1,5])
    f10[k,i,j]=f9[k,i,j]



##Compute macroscopic property
T=f0+f1+f2+f3+f4+f5+f6+f7+f8+f9+f10+f11+f12+f13+f14
```

[37]:

```
print(T)
```

```
[[[ 1.00000000e+00  1.00000000e+00  1.00000000e+00  1.00000000e+00
    1.00000000e+00  1.00000000e+00  1.00000000e+00  1.00000000e+00
    1.00000000e+00  1.00000000e+00]
  [ 1.00000000e+00  1.00000000e+00  1.00000000e+00  1.00000000e+00
    1.00000000e+00  1.00000000e+00  1.00000000e+00  1.00000000e+00
    1.00000000e+00  1.00000000e+00]
  [ 1.00000000e+00  1.00000000e+00  1.00000000e+00  1.00000000e+00
    1.00000000e+00  1.00000000e+00  1.00000000e+00  1.00000000e+00
    1.00000000e+00  1.00000000e+00]
  [ 1.00000000e+00  1.00000000e+00  1.00000000e+00  1.00000000e+00
    1.00000000e+00  1.00000000e+00  1.00000000e+00  1.00000000e+00
    1.00000000e+00  1.00000000e+00]
  [ 1.00000000e+00  1.00000000e+00  1.00000000e+00  1.00000000e+00
    1.00000000e+00  1.00000000e+00  1.00000000e+00  1.00000000e+00
    1.00000000e+00  1.00000000e+00]
  [ 1.00000000e+00  1.00000000e+00  1.00000000e+00  1.00000000e+00
    1.00000000e+00  1.00000000e+00  1.00000000e+00  1.00000000e+00
    1.00000000e+00  1.00000000e+00]
  [ 1.00000000e+00  1.00000000e+00  1.00000000e+00  1.00000000e+00
    1.00000000e+00  1.00000000e+00  1.00000000e+00  1.00000000e+00
    1.00000000e+00  1.00000000e+00]
  [ 1.00000000e+00  1.00000000e+00  1.00000000e+00  1.00000000e+00
    1.00000000e+00  1.00000000e+00  1.00000000e+00  1.00000000e+00
    1.00000000e+00  1.00000000e+00]
  [ 1.00000000e+00  1.00000000e+00  1.00000000e+00  1.00000000e+00
    1.00000000e+00  1.00000000e+00  1.00000000e+00  1.00000000e+00
    1.00000000e+00  1.00000000e+00]
  [ 1.00000000e+00  1.00000000e+00  1.00000000e+00  1.00000000e+00
    1.00000000e+00  1.00000000e+00  1.00000000e+00  1.00000000e+00
    1.00000000e+00  1.00000000e+00]]

 [[ 0.00000000e+00 -1.38777878e-17 -1.38777878e-17 -1.38777878e-17
   -1.38777878e-17 -1.38777878e-17 -1.38777878e-17 -1.38777878e-17
   -1.38777878e-17  5.55555556e-02]
  [ 8.67361738e-18  1.66666667e-01  1.66666667e-01  1.66666667e-01
    1.66666667e-01  1.66666667e-01  1.66666667e-01  1.66666667e-01
```

```
  1.66666667e-01 -1.38777878e-17]
 [ 8.67361738e-18  1.66666667e-01  1.66666667e-01  1.66666667e-01
   1.66666667e-01  1.66666667e-01  1.66666667e-01  1.66666667e-01
   1.66666667e-01 -1.38777878e-17]
 [ 8.67361738e-18  1.66666667e-01  1.66666667e-01  1.66666667e-01
   1.66666667e-01  1.66666667e-01  1.66666667e-01  1.66666667e-01
   1.66666667e-01 -1.38777878e-17]
 [ 8.67361738e-18  1.66666667e-01  1.66666667e-01  1.66666667e-01
   1.66666667e-01  1.66666667e-01  1.66666667e-01  1.66666667e-01
   1.66666667e-01 -1.38777878e-17]
 [ 8.67361738e-18  1.66666667e-01  1.66666667e-01  1.66666667e-01
   1.66666667e-01  1.66666667e-01  1.66666667e-01  1.66666667e-01
   1.66666667e-01 -1.38777878e-17]
 [ 8.67361738e-18  1.66666667e-01  1.66666667e-01  1.66666667e-01
   1.66666667e-01  1.66666667e-01  1.66666667e-01  1.66666667e-01
   1.66666667e-01 -1.38777878e-17]
 [ 8.67361738e-18  1.66666667e-01  1.66666667e-01  1.66666667e-01
   1.66666667e-01  1.66666667e-01  1.66666667e-01  1.66666667e-01
   1.66666667e-01 -1.38777878e-17]
 [ 8.67361738e-18  1.66666667e-01  1.66666667e-01  1.66666667e-01
   1.66666667e-01  1.66666667e-01  1.66666667e-01  1.66666667e-01
   1.66666667e-01 -1.38777878e-17]
 [ 0.00000000e+00 -5.20417043e-18 -5.20417043e-18 -5.20417043e-18
  -5.20417043e-18 -5.20417043e-18 -5.20417043e-18 -5.20417043e-18
  -5.20417043e-18 -1.38777878e-17]]

[[ 0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
 [ 0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
 [ 0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
 [ 0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
 [ 0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
 [ 0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
 [ 0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
 [ 0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
```

```
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]]

 [[ 0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
  [ 0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
  [ 0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
  [ 0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
  [ 0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
  [ 0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
  [ 0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
  [ 0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
  [ 0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
  [ 0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]]]
```

[5]:
```
iter=10000
while(iter>=1):
    ##Computing equilibrium distribution function
    f0eq=w[0]*T
    f1eq=w[1]*T
    f2eq=w[2]*T
    f3eq=w[3]*T
    f4eq=w[4]*T
    f5eq=w[5]*T
    f6eq=w[6]*T
    f7eq=w[7]*T
    f8eq=w[8]*T
    f9eq=w[9]*T
```

```
    f10eq=w[10]*T
    f11eq=w[11]*T
    f12eq=w[12]*T
    f13eq=w[13]*T
    f14eq=w[14]*T
    ##Collision of digital particles
    f0c=(1-omega)*f0+(omega*f0eq)
    f1c=(1-omega)*f1+(omega*f1eq)
    f2c=(1-omega)*f2+(omega*f2eq)
    f3c=(1-omega)*f3+(omega*f3eq)
    f4c=(1-omega)*f4+(omega*f4eq)
    f5c=(1-omega)*f5+(omega*f5eq)
    f6c=(1-omega)*f6+(omega*f6eq)
    f7c=(1-omega)*f7+(omega*f7eq)
    f8c=(1-omega)*f8+(omega*f8eq)
    f9c=(1-omega)*f9+(omega*f9eq)
    f10c=(1-omega)*f10+(omega*f10eq)
    f11c=(1-omega)*f11+(omega*f11eq)
    f12c=(1-omega)*f12+(omega*f12eq)
    f13c=(1-omega)*f13+(omega*f13eq)
    f14c=(1-omega)*f14+(omega*f14eq)
    ##Streaming of digital particles

    ##particle number 0 is at rest
    f0=f0c
    ##digital particle numbered from 1,2,3,...6 located on faces

    ##particle number 3 which is moving upwards towards top boundary
    for k in np.arange(0,Nz-1):
        for i in np.arange(0,Ny):
             for j in np.arange(0,Nx):
                f3[k+1,i,j]=f3c[k,i,j]

    ##particle number 4 which is moving downwards towards bottom boundary
    for k in np.arange(0,Nz-1):
        for i in np.arange(0,Ny):
            for j in np.arange(0,Nx):
                f4[k,i,j]=f4c[k+1,i,j]

    ##digital particle 1 which moves on x-y plane with unit velocity in␣
↪x-direction
    for k in np.arange(0,Nz):
        for i in np.arange(0,Ny):
            for j in np.arange(0,Nx-1):
                f1[k,i,j+1]=f1c[k,i,j]

    ##digital particle 2 moves on x-y plane in negative x-direction
```

```
    for k in np.arange(0,Nz):
        for i in np.arange(0,Ny):
            for j in np.arange(0,Nx-1):
                f2[k,i,j]=f2c[k,i,j+1]

    ##digital particle 6 which moves on x-y plane upwards in positive y-direction
    for k in np.arange(0,Nz):
        for j in np.arange(0,Nx):
            for i in np.arange(0,Ny-1):
                f6[k,i,j]=f6c[k,i+1,j]

    ##digital particle 5 which moves on x-y plane upwards in negative y-direction
    for k in np.arange(0,Nz):
        for j in np.arange(0,Nx):
            for i in np.arange(0,Ny-1):
                f5[k,i+1,j]=f5c[k,i,j]

    ##digital particle 7
    for k in np.arange(0,Nz-1):
        for i in np.arange(0,Ny-1):
            for j in np.arange(0,Nx-1):
                f7[k+1,i+1,j+1]=f7c[k,i,j]

    ##digital particle 12
    for k in np.arange(0,Nz-1):
        for i in np.arange(0,Ny-1):
            for j in np.arange(0,Nx-1):
                f12[k+1,i,j]=f12c[k,i+1,j+1]


    ##digital particle 9
    for k in np.arange(0,Nz-1):
        for i in np.arange(0,Ny-1):
            for j in np.arange(0,Nx-1):
                f9[k+1,i,j+1]=f9c[k,i+1,j]

    ##digital particle 14
    for k in np.arange(0,Nz-1):
        for i in np.arange(0,Ny-1):
            for j in np.arange(0,Nx-1):
                f14[k+1,i+1,j]=f14c[k,i,j+1]

    ##digital particle 11
    for k in np.arange(0,Nz-1):
        for i in np.arange(0,Ny-1):
```

```
            for j in np.arange(0,Nx-1):
                f11[k,i+1,j+1]=f11c[k+1,i,j]

    ##Digital particle 8
    for k in np.arange(0,Nz-1):
        for i in np.arange(0,Ny-1):
            for j in np.arange(0,Nx-1):
                f8[k,i,j]=f8c[k+1,i+1,j+1]

    ##digital particle 13
    for k in np.arange(0,Nz-1):
        for i in np.arange(0,Ny-1):
            for j in np.arange(0,Nx-1):
                f13[k,i,j+1]=f13c[k+1,i+1,j]

    ##digital particle 10
    for k in np.arange(0,Nz-1):
        for i in np.arange(0,Ny-1):
            for j in np.arange(0,Nx-1):
                f10[k,i+1,j]=f10c[k+1,i,j+1]


    ##Boundary conditions

    ##West boundaries
    ##unknowns are f1 f7 f9 f11 f13
    for k in np.arange(1,Nz-1):
        j=0
        for i in np.arange(1,Ny-1):
            f1[k,i,j]=w[1]*Tw+ w[2]*Tw-f2[k,i,j]
            f7[k,i,j]=w[7]*Tw+w[8]*Tw-f8[k,i,j]
            f9[k,i,j]=w[9]*Tw+w[10]*Tw-f10[k,i,j]
            f11[k,i,j]=w[11]*Tw+w[12]*Tw-f12[k,i,j]
            ␣
↪f13[k,i,j]=Tw*(1-w[1]-w[2]-w[7]-w[8]-w[9]-w[10]-w[11]-w[12])-(f0[k,i,j]+f3[k,i,j]+f4[k,i,j]


    ##East boundary
    ##unkown df are f2,f8,f10,f12,f14
    for k in np.arange(1,Nz-1):
        j=Nx-1
        for i in np.arange(1,Ny-1):
            f2[k,i,j]=w[1]*Te+ w[2]*Te-f1[k,i,j]
            f8[k,i,j]=w[7]*Te+w[8]*Te-f7[k,i,j]
            f10[k,i,j]=w[9]*Te+w[10]*Te-f9[k,i,j]
            f12[k,i,j]=w[11]*Te+w[12]*Te-f11[k,i,j]
```

```
        ␣
↪f14[k,i,j]=Te*(1-w[1]-w[2]-w[7]-w[8]-w[9]-w[10]-w[11]-w[12])-(f0[k,i,j]+f3[k,i,j]+f4[k,i,j]

    ##North boundary

    ##Interior lattice nodes
    ##unknown df are f5 ,f7 ,f10 , f11 , f14
    for k in np.arange(1,Nz-1):
        i=0
        for j in np.arange(1,Nx-1):
            f5[k,i,j]=w[5]*Tn +w[6]*Tn-f6[k,i,j]
            f7[k,i,j]=w[7]*Tn+w[8]*Tn-f8[k,i,j]
            f10[k,i,j]=w[9]*Tn+w[10]*Tn-f9[k,i,j]
            f11[k,i,j]=w[11]*Tn+w[12]*Tn-f12[k,i,j]
        ␣
↪f14[k,i,j]=Tn*(1-w[5]-w[6]-w[7]-w[8]-w[9]-w[10]-w[11]-w[12])-(f0[k,i,j]+f1[k,i,j]+f2[k,i,j]

    ##corner lattice nodes
    ##Left corner lattice node
    ##f1,f5,f7,f9,f10,f11,f13,f14
    for k in np.arange(1,Nz-1):
        i=0
        j=0
        f1[k,i,j]=w[1]*Tn+w[2]*Tn-f2[k,i,j]
        f5[k,i,j]=w[5]*Tn +w[6]*Tn-f6[k,i,j]
        f7[k,i,j]=w[7]*Tn+w[8]*Tn-f8[k,i,j]
        f11[k,i,j]=w[11]*Tn+w[12]*Tn-f12[k,i,j]
        f9[k,i,j]=0.5*((18/72)*Tn-f0[k,i,j]-f3[k,i,j])
        f10[k,i,j]=f9[k,i,j]
        f13[k,i,j]=0.5*((18/72)*Tn-f4[k,i,j])
        f14[k,i,j]=f13[k,i,j]
    ##Right corner lattice node
    ##f2,f5,f7,f8,f10,f11,f12,f14
    for k in np.arange(1,Nz-1):
        i=0
        j=Nx-1
        f2[k,i,j]=w[1]*Tn+w[2]*Tn-f1[k,i,j]
        f5[k,i,j]=w[5]*Tn +w[6]*Tn-f6[k,i,j]
        f10[k,i,j]=w[9]*Tn +w[10]*Tn-f9[k,i,j]
        f14[k,i,j]=w[13]*Tn+w[14]*Tn-f13[k,i,j]
        f7[k,i,j]=0.5*((18/72)*Tn-f0[k,i,j]-f3[k,i,j])
        f8[k,i,j]=f7[1,0,5]
        f11[k,i,j]=0.5*((18/72)*Tn-f4[k,i,j])
        f12[k,i,j]=f11[k,i,j]


    ##South boundary
```

```
    ##Interior lattice nodes
    ##unknown vdf f6,f8,f9,f12,f13
    for k in np.arange(1,Nz-1):
        i=Ny-1
        for j in np.arange(1,Nx-1):
            f6[k,i,j]=w[5]*Ts +w[6]*Ts-f5[k,i,j]
            f8[k,i,j]=w[7]*Ts+w[8]*Ts-f7[k,i,j]
            f9[k,i,j]=w[9]*Ts+w[10]*Ts-f10[k,i,j]
            f12[k,i,j]=w[11]*Ts+w[12]*Ts-f11[k,i,j]
            ␣
↪f13[k,i,j]=Ts*(1-w[5]-w[6]-w[7]-w[8]-w[9]-w[10]-w[11]-w[12])-(f0[k,i,j]+f1[k,i,j]+f2[k,i,j]

    ##corner lattice nodes
    ##Left corner node
    ##f1,f6,f7,f8,f9,f11,f12,f13
    for k in np.arange(1,Nz-1):
        i=Ny-1
        j=0
        f1[k,i,j]=w[1]*Ts+w[2]*Ts-f2[k,i,j]
        f6[k,i,j]=w[5]*Ts +w[6]*Ts-f5[k,i,j]
        f9[k,i,j]=w[9]*Ts +w[10]*Ts-f10[k,i,j]
        f13[k,i,j]=w[13]*Ts+w[14]*Ts-f14[k,i,j]
        f7[k,i,j]=0.5*((18/72)*Ts-f0[k,i,j]-f3[k,i,j])
        f8[k,i,j]=f7[k,i,j]
        f11[k,i,j]=0.5*((18/72)*Ts-f4[k,i,j])
        f12[k,i,j]=f11[k,i,j]

    ##Right corner node
    ##f2,f6,f8,f9,f10,f12,f13,f14
    for k in np.arange(1,Nz-1):
        i=Ny-1
        j=Nx-1
        f2[k,i,j]=w[1]*Ts+w[2]*Ts-f1[k,i,j]
        f6[k,i,j]=w[5]*Ts +w[6]*Ts-f5[k,i,j]
        f8[k,i,j]=w[7]*Ts+w[8]*Ts-f7[k,i,j]
        f12[k,i,j]=w[11]*Ts+w[12]*Ts-f11[k,i,j]
        f9[k,i,j]=0.5*((18/72)*Ts-f0[k,i,j]-f3[k,i,j])
        f10[k,i,j]=f9[k,i,j]
        f13[k,i,j]=0.5*((18/72)*Ts-f4[k,i,j])
        f14[k,i,j]=f13[k,i,j]

    ##Top boundary

    ##Inner lattice nodes
    ##Unknown distribution functions are f4,f8,f10,f11,,f13
    k=Nz-1
```

```
    for i in np.arange(1,Ny-1):
        for j in np.arange(1,Nx-1):
            f4[k,i,j]=w[4]*T_top +w[3]*T_top-f3[k,i,j]
            f8[k,i,j]=w[7]*T_top+w[8]*T_top-f7[k,i,j]
            f10[k,i,j]=w[9]*T_top+w[10]*T_top-f9[k,i,j]
            f11[k,i,j]=w[11]*T_top+w[12]*T_top-f12[k,i,j]
            ␣
↪f13[k,i,j]=T_top*(1-w[5]-w[6]-w[7]-w[8]-w[9]-w[10]-w[11]-w[12])-(f0[k,i,j]+f1[k,i,j]+f2[k,i



    ##Top north
    ##Interior
    ##unknownf4,f5,f7,f8,f10,f11,f13,f14
    k=Nz-1
    i=0
    for j in np.arange(1,Nx-1):
        f4[k,i,j]=w[4]*T_top +w[3]*T_top-f3[k,i,j]
        f5[k,i,j]=w[5]*T_top +w[6]*T_top-f6[k,i,j]
        f10[k,i,j]=w[9]*T_top+w[10]*T_top-f9[k,i,j]
        f11[k,i,j]=w[11]*T_top+w[12]*T_top-f12[k,i,j]
        f7[k,i,j]=0.5*((18/72)*T_top-f0[k,0,1]-f1[k,i,j])
        f8[k,i,j]=f7[5,0,1]
        f13[k,i,j]=0.5*((18/72)*T_top-f2[k,i,j])
        f14[k,i,j]=f13[k,i,j]

    ##Left corner north lattice node
    ##unknown df are f1,f4,f5,f7,f8,f9,f10,f11,f13,f14
    k=Nz-1
    i=0
    j=0
    f1[k,i,j]=w[1]*T_top+w[2]*T_top-f2[k,i,j]
    f4[k,i,j]=w[4]*T_top +w[3]*T_top-f3[k,i,j]
    f5[k,i,j]=w[5]*T_top +w[6]*T_top-f6[k,i,j]
    f11[k,i,j]=w[11]*T_top+w[12]*T_top-f12[k,i,j]
    f7[k,i,j]=0.5*((10/72)*T_top-f0[k,i,j])
    f8[k,i,j]=f7[k,i,j]
    f9[k,i,j]=0.5*((6/72)*T_top)
    f10[k,i,j]=f9[k,i,j]
    f13[k,i,j]=0.5*((6/72)*T_top)
    f14[k,i,j]=f13[k,i,j]

    ##Right corner north lattice node
    ##unknown df are f2,f4,f5,f7,f8,f10,f11,f13,f14
    k=Nz-1
    i=0
    j=5
```

```
    f2[k,i,j]=w[1]*T_top+w[2]*T_top-f1[k,i,j]
    f4[k,i,j]=w[4]*T_top +w[3]*T_top-f3[k,i,j]
    f5[k,i,j]=w[5]*T_top +w[6]*T_top-f6[k,i,j]
    f10[k,i,j]=w[9]*T_top+w[10]*T_top-f9[k,i,j]
    f7[k,i,j]=0.5*((10/72)*T_top-f0[k,i,j])
    f8[k,i,j]=f7[k,i,j]
    f11[k,i,j]=0.5*((6/72)*T_top)
    f12[k,i,j]=f11[k,i,j]
    f13[k,i,j]=0.5*((6/72)*T_top)
    f14[k,i,j]=f13[k,i,j]


    ##Top south
    ##unknowns f4,f6,f8,f9,f10,f11,f12,f13
    k=Nz-1
    i=Ny-1
    for j in np.arange(1,Nx-1):
        f4[k,i,j]=w[4]*T_top +w[3]*T_top-f3[k,i,j]
        f6[k,i,j]=w[5]*T_top +w[6]*T_top-f5[k,i,j]
        f8[k,i,j]=w[7]*T_top+w[8]*T_top-f7[k,i,j]
        f13[k,i,j]=w[14]*T_top+w[13]*T_top-f14[k,i,j]
        f9[k,i,j]=0.5*((18/72)*T_top-f0[k,i,j]-f1[k,i,j])
        f10[k,i,j]=f9[k,i,j]
        f11[k,i,j]=0.5*((18/72)*T_top-f2[k,i,j])
        f12[k,i,j]=f11[k,i,j]

    ##Left corner lattice node
    ##unknown df are f1,f4,f6,f7,f8,f9,f10,f11,f12,f13
    k=Nz-1
    i=Ny-1
    j=0
    f1[k,i,j]=w[1]*T_top+w[2]*T_top-f2[k,i,j]
    f4[k,i,j]=w[4]*T_top +w[3]*T_top-f3[k,i,j]
    f6[k,i,j]=w[5]*T_top +w[6]*T_top-f5[k,i,j]
    f13[k,i,j]=w[14]*T_top+w[13]*T_top-f14[k,i,j]
    f7[k,i,j]=0.5*((10/72)*T_top-f0[k,5,0])
    f8[k,i,j]=f7[k,i,j]
    f9[k,i,j]=0.5*((6/72)*T_top)
    f10[k,i,j]=f9[k,i,j]
    f11[k,i,j]=0.5*((6/72)*T_top)
    f12[k,i,j]=f11[k,i,j]

    ##Right corner lattice node
    ##unknown df are f2,f4,f6,f8,f9,f10,f11,f12,f13,f14
    k=Nz-1
    i=Ny-1
    j=Nx-1
```

```
    f2[k,i,j]=w[1]*T_top+w[2]*T_top-f1[k,i,j]
    f4[k,i,j]=w[4]*T_top +w[3]*T_top-f3[k,i,j]
    f6[k,i,j]=w[5]*T_top +w[6]*T_top-f5[k,i,j]
    f8[k,i,j]=w[7]*T_top+w[8]*T_top-f7[k,i,j]
    f9[k,i,j]=0.5*((10/72)*T_top-f0[k,i,j])
    f10[k,i,j]=f9[k,i,j]
    f11[k,i,j]=0.5*((6/72)*T_top)
    f12[k,i,j]=f11[k,i,j]
    f13[k,i,j]=0.5*((6/72)*T_top)
    f14[k,i,j]=f13[k,i,j]

    ##Top west
    ##unknown distribution functions
    ##f1,f4,f7,f8,f9,f10,f11,f13
    k=Nz-1
    j=0
    for i in np.arange(1,Ny-1):
        f1[k,i,j]=w[1]*T_top+w[2]*T_top-f2[k,i,j]
        f4[k,i,j]=w[4]*T_top +w[3]*T_top-f3[k,i,j]
        f11[k,i,j]=w[11]*T_top+w[12]*T_top-f12[k,i,j]
        f13[k,i,j]=w[14]*T_top+w[13]*T_top-f14[k,i,j]
        f7[k,i,j]=0.5*((18/72)*T_top-f0[k,i,j]-f5[k,i,j])
        f8[k,i,j]=f7[k,i,j]
        f9[k,i,j]=0.5*((18/72)*T_top-f6[5,1,0])
        f10[k,i,j]=f9[k,i,j]


    ##Top East
    ##unknown df f2,f4,f8,f10,f11,f12,f13,f14
    k=Nz-1
    j=Ny-1
    for i in np.arange(1,Nx-1):
        f2[k,i,j]=w[1]*T_top+w[2]*T_top-f1[k,i,j]
        f4[k,i,j]=w[4]*T_top +w[3]*T_top-f3[k,i,j]
        f8[k,i,j]=w[7]*T_top+w[8]*T_top-f7[k,i,j]
        f10[k,i,j]=w[9]*T_top+w[10]*T_top-f9[k,i,j]
        f11[k,i,j]=0.5*((18/72)*T_top-f0[k,i,j]-f5[k,i,j])
        f12[k,i,j]=f11[5,1,5]
        f13[k,i,j]=0.5*((18/72)*T_top-f6[k,i,j])
        f14[k,i,j]=f13[k,i,j]



    ##Bottom boundary

    ##Inner lattice nodes
    ##unknown distribution functions are f3,f7,f9,f12,f14
```

```
    k=0
    for i in np.arange(1,Ny-1):
        for j in np.arange(1,Nx-1):
            f3[k,i,j]=w[4]*T_bot +w[3]*T_bot-f4[k,i,j]
            f7[k,i,j]=w[7]*T_bot+w[8]*T_bot-f8[k,i,j]
            f9[k,i,j]=w[9]*T_bot+w[10]*T_bot-f10[k,i,j]
            f12[k,i,j]=w[11]*T_bot+w[12]*T_bot-f11[k,i,j]
            ␣
↪f14[k,i,j]=T_bot*(1-w[5]-w[6]-w[7]-w[8]-w[9]-w[10]-w[11]-w[12])-(f0[k,i,j]+f1[k,i,j]+f2[k,i


    ##Bottom north
    ##Interior lattice nods
    ##unknown distribution functions are f3,f5,f7,f9,f10,f11,f12,f14
    k=0
    i=0
    for j in np.arange(1,Nx-1):
        f3[k,i,j]=w[4]*T_bot +w[3]*T_bot-f4[k,i,j]
        f5[k,i,j]=w[5]*T_bot +w[6]*T_bot-f6[k,i,j]
        f7[k,i,j]=w[7]*T_bot+ w[8]*T_bot-f8[k,i,j]
        f14[k,i,j]=w[14]*T_bot+w[13]*T_bot-f13[k,i,j]
        f9[k,i,j]=0.5*((18/72)*T_bot-f0[k,i,j]-f1[k,i,j])
        f10[k,i,j]=f9[k,i,j]
        f11[k,i,j]=0.5*((18/72)*T_bot-f2[k,i,j])
        f12[k,i,j]=f11[k,i,j]

    ##left corner lattice node
    ##unknown df is f1, f3,f5,f7,f9,f10,f11,f12,f13,f14
    k=0
    i=0
    j=0
    f1[k,i,j]=w[1]*T_bot+w[2]*T_bot-f2[k,i,j]
    f3[k,i,j]=w[4]*T_bot +w[3]*T_bot-f4[k,i,j]
    f5[k,i,j]=w[5]*T_bot +w[6]*T_bot-f6[k,i,j]
    f7[k,i,j]=w[7]*T_bot+w[8]*T_bot-f8[k,i,j]
    f9[k,i,j]=0.5*((10/72)*T_bot-f0[k,i,j])
    f10[k,i,j]=f9[k,i,j]
    f11[k,i,j]=0.5*((6/72)*T_bot)
    f12[k,i,j]=f11[k,i,j]
    f13[k,i,j]=0.5*((6/72)*T_bot)
    f14[k,i,j]=f13[k,i,j]

    ##right corner lattice node
    ##unknown df is f2,f3,f5,f7,f8,f9,f10,f11,f12,f14
    k=0
    i=0
    j=Nx-1
```

```
f2[k,i,j]=w[1]*T_bot+w[2]*T_bot-f1[k,i,j]
f3[k,i,j]=w[4]*T_bot +w[3]*T_bot-f4[k,i,j]
f5[k,i,j]=w[5]*T_bot +w[6]*T_bot-f6[k,i,j]
f14[k,i,j]=w[11]*T_bot+w[12]*T_bot-f13[k,i,j]
f7[k,i,j]=0.5*((10/72)*T_bot-f0[k,i,j])
f8[k,i,j]=f7[k,i,j]
f9[k,i,j]=0.5*((6/72)*T_bot)
f10[k,i,j]=f9[k,i,j]
f11[k,i,j]=0.5*((6/72)*T_bot)
f12[k,i,j]=f11[k,i,j]


##Bottom south

##Interior lattice nodes
##f3,f6,f7,f8,f9,f12,f13,f14
k=0
i=Ny-1
for j in np.arange(1,Nx-1):
    f3[k,i,j]=w[4]*T_bot +w[3]*T_bot-f4[k,i,j]
    f6[k,i,j]=w[5]*T_bot +w[6]*T_bot-f5[k,i,j]
    f9[k,i,j]=w[9]*T_bot+w[10]*T_bot-f10[k,i,j]
    f12[k,i,j]=w[11]*T_bot+w[12]*T_bot-f11[k,i,j]
    f7[k,i,j]=0.5*((18/72)*T_bot-f0[0,5,1]-f1[k,i,j])
    f8[k,i,j]=f7[k,i,j]
    f13[k,i,j]=0.5*((18/72)*T_bot-f2[k,i,j])
    f14[k,i,j]=f13[k,i,j]


##Bottom south left corner lattice node
##unknown vdf is f1,f3,f6,f7,f8,f9,f11,f12,f13,f14
k=0
i=Ny-1
j=0
f1[k,i,j]=w[1]*T_bot+w[2]*T_bot-f2[k,i,j]
f3[k,i,j]=w[4]*T_bot +w[3]*T_bot-f4[k,i,j]
f6[k,i,j]=w[5]*T_bot +w[6]*T_bot-f5[k,i,j]
f9[k,i,j]=w[9]*T_bot+w[10]*T_bot-f10[k,i,j]
f7[k,i,j]=0.5*((10/72)*T_bot-f0[k,5,0])
f8[k,i,j]=f7[k,i,j]
f11[k,i,j]=0.5*((6/72)*T_bot)
f12[k,i,j]=f11[k,i,j]
f13[k,i,j]=0.5*((6/72)*T_bot)
f14[k,i,j]=f13[k,i,j]
##Bottom south right corner lattice node
##unknown df are f2,f3,f6,f7,f8,f9,f10,f12,f13,f14
k=0
```

```
    i=Ny-1
    j=Nx-1
    f1[k,i,j]=w[1]*T_bot+w[2]*T_bot-f2[k,i,j]
    f3[k,i,j]=w[4]*T_bot +w[3]*T_bot-f4[k,i,j]
    f6[k,i,j]=w[5]*T_bot +w[6]*T_bot-f5[k,i,j]
    f12[k,i,j]=w[11]*T_bot+w[12]*T_bot-f11[k,i,j]
    f7[k,i,j]=0.5*((10/72)*T_bot-f0[k,i,j])
    f8[k,i,j]=f7[k,i,j]
    f9[k,i,j]=0.5*((6/72)*T_bot)
    f10[k,i,j]=f9[k,i,j]
    f13[k,i,j]=0.5*((6/72)*T_bot)
    f14[k,i,j]=f13[k,i,j]

    ##Bottom west
    ##f1,f3,f7,f9,f11,f12,f13,f14
    k=0
    j=0
    for i in np.arange(1,Ny-1):
        f1[k,i,j]=w[1]*T_bot+w[2]*T_bot-f2[k,i,j]
        f3[k,i,j]=w[4]*T_bot +w[3]*T_bot-f4[k,i,j]
        f7[k,i,j]=w[7]*T_bot+w[8]*T_bot-f8[k,i,j]
        f9[k,i,j]=w[9]*T_bot+w[10]*T_bot-f10[k,i,j]
        f11[k,i,j]=0.5*((18/72)*T_bot-f0[k,i,j]-f5[k,i,j])
        f12[k,i,j]=f11[k,i,j]
        f13[k,i,j]=0.5*((18/72)*T_bot-f6[k,i,j])
        f14[k,i,j]=f13[k,i,j]



    ##Bottom east
    #f2,f3,f7,f8,f9,f10,f12,f14
    k=0
    j=Nx-1
    for i in np.arange(1,Ny-1):
        f2[k,i,j]=w[1]*T_bot+w[2]*T_bot-f1[k,i,j]
        f3[k,i,j]=w[4]*T_bot +w[3]*T_bot-f4[k,i,j]
        f12[k,i,j]=w[11]*T_bot+w[12]*T_bot-f11[k,i,j]
        f14[k,i,j]=w[13]*T_bot+w[14]*T_bot-f13[k,i,j]
        f7[k,i,j]=0.5*((18/72)*T_bot-f0[k,1,5]-f5[k,i,j])
        f8[k,i,j]=f7[k,i,j]
        f9[k,i,j]=0.5*((18/72)*T_bot-f6[0,1,5])
        f10[k,i,j]=f9[k,i,j]


    ##Compute macroscopic property
    T=f0+f1+f2+f3+f4+f5+f6+f7+f8+f9+f10+f11+f12+f13+f14
```

```
    print(iter)
    iter=iter-1
```

```
10000
9999
9998
9997
9996
9995
9994
9993
9992
9991
9990
9989
9988
9987
9986
9985
9984
9983
9982
9981
9980
9979
9978
9977
9976
9975
9974
9973
9972
9971
9970
9969
9968
9967
9966
9965
9964
9963
9962
9961
9960
9959
9958
9957
```

6308
6307
6306
6305
6304
6303
6302
6301
6300
6299
6298
6297
6296
6295
6294
6293
6292
6291
6290
6289
6288
6287
6286
6285
6284
6283
6282
6281
6280
6279
6278
6277
6276
6275
6274
6273
6272
6271
6270
6269
6268
6267
6266
6265
6264
6263
6262
6261

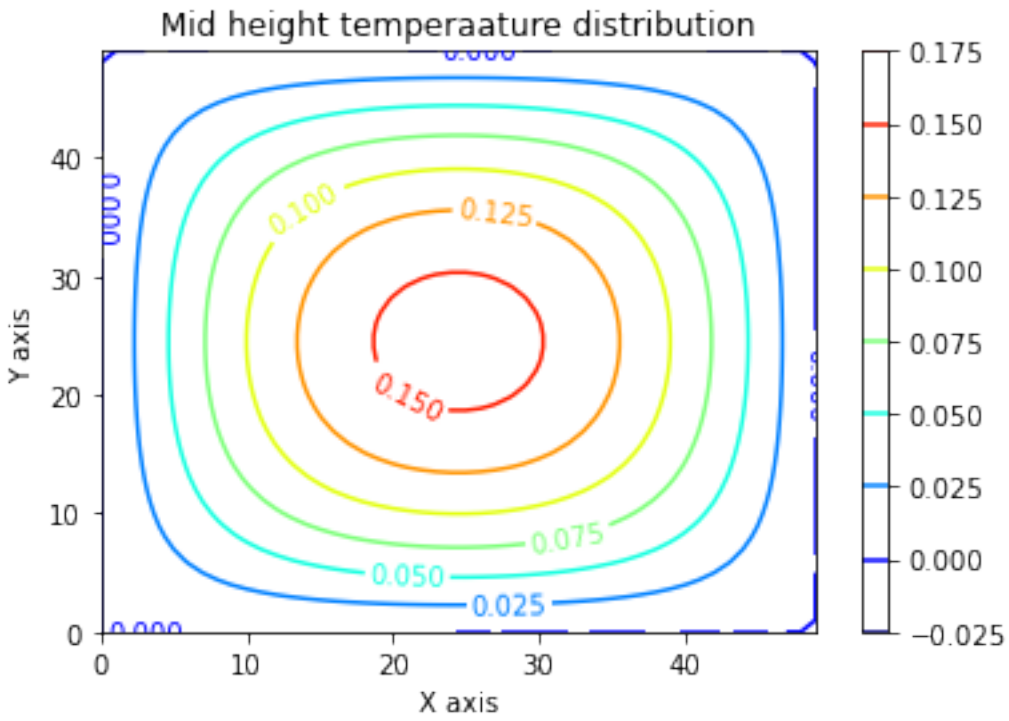
5252
5251
5250
5249
5248
5247
5246
5245
5244
5243
5242
5241
5240
5239
5238
5237
5236
5235
5234
5233
5232
5231
5230
5229
5228
5227
5226
5225
5224
5223
5222
5221
5220
5219
5218
5217
5216
5215
5214
5213
5212
5211
5210
5209
5208
5207
5206
5205

5108
5107
5106
5105
5104
5103
5102
5101
5100
5099
5098
5097
5096
5095
5094
5093
5092
5091
5090
5089
5088
5087
5086
5085
5084
5083
5082
5081
5080
5079
5078
5077
5076
5075
5074
5073
5072
5071
5070
5069
5068
5067
5066
5065
5064
5063
5062
5061

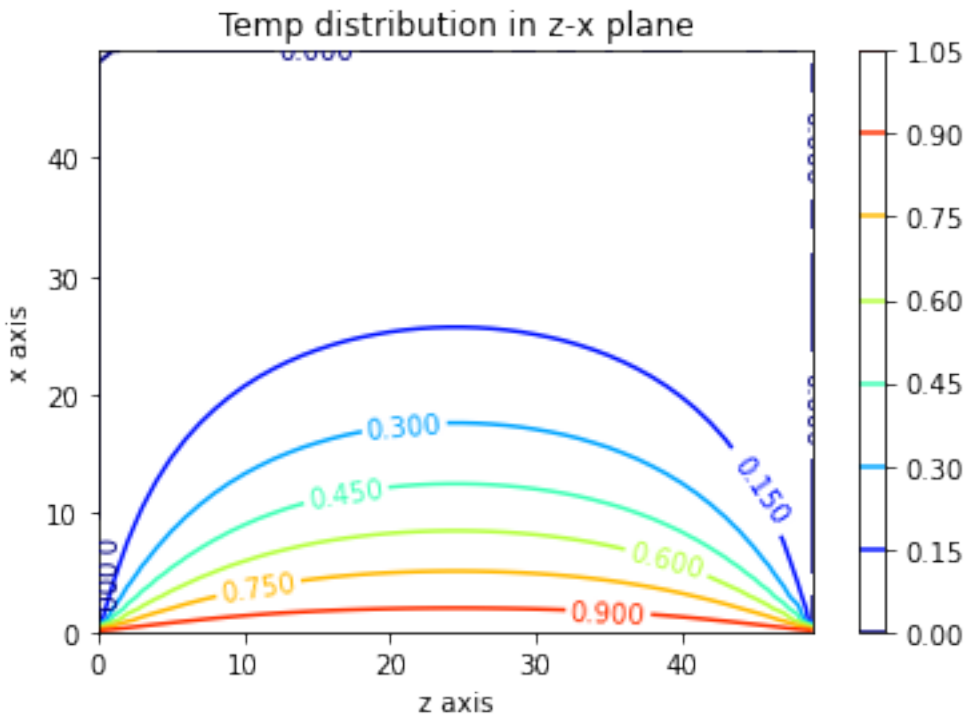
```
20
19
18
17
16
15
14
13
12
11
10
9
8
7
6
5
4
3
2
1
```

print(T)

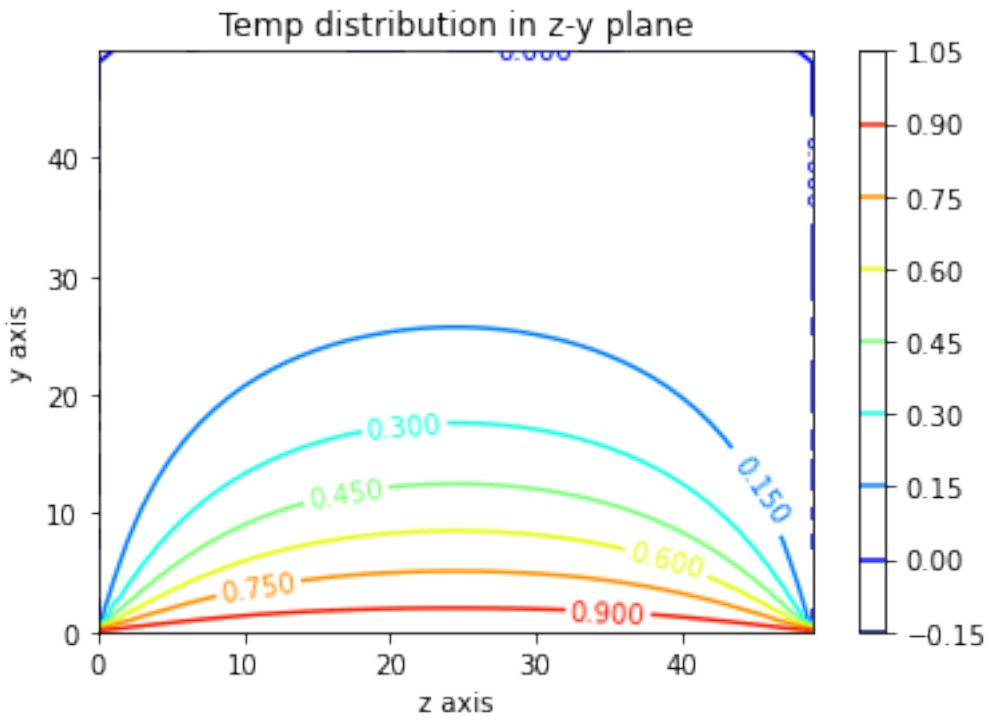
[6]:
```
cs=plt.contour(T[25,:,:],cmap='jet')
plt.clabel(cs, inline=1, fontsize=10)
plt.colorbar()
plt.xlabel('X axis')
plt.ylabel('Y axis')
plt.title('Mid height temperaature distribution')
plt.show()
```



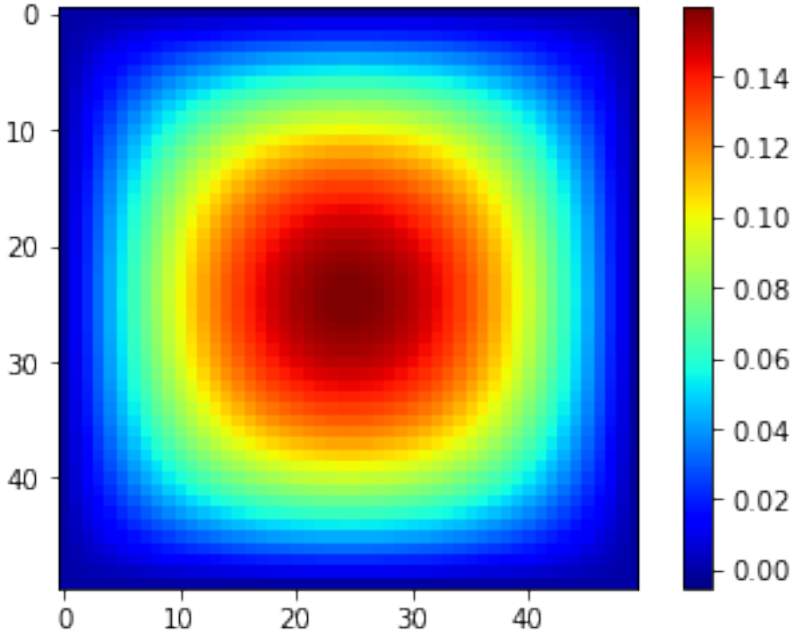
[11]:
```
cs=plt.contour(T[:,25,:],cmap='jet')
plt.clabel(cs,inline=1,fontsize=10)
plt.xlabel('z axis')
plt.ylabel('x axis')
plt.title('Temp distribution in z-x plane')
plt.colorbar()
plt.show()
```



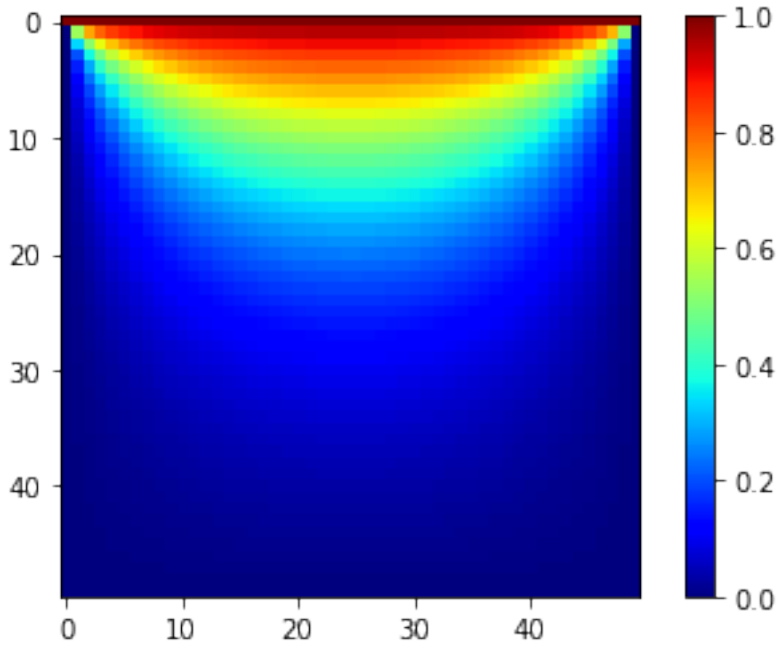
[12]:
```
cs=plt.contour(T[:,:,25],cmap='jet')
plt.clabel(cs,inline=1,fontsize=10)
plt.xlabel('z axis')
plt.ylabel('y axis')
plt.title('Temp distribution in z-y plane')
plt.colorbar()
plt.show()
```



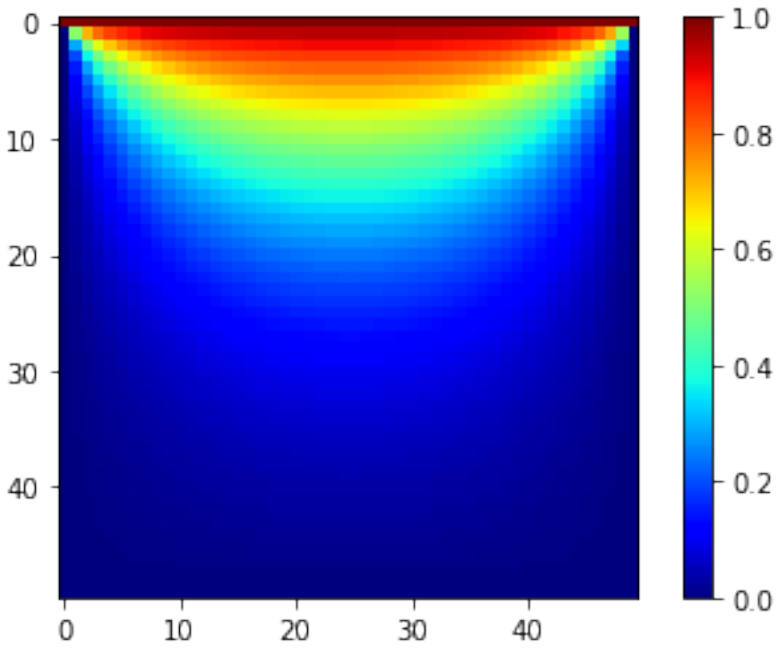
[19]:
```
plt.imshow(T[25,:,:],cmap='jet')
plt.colorbar()
plt.show()
```



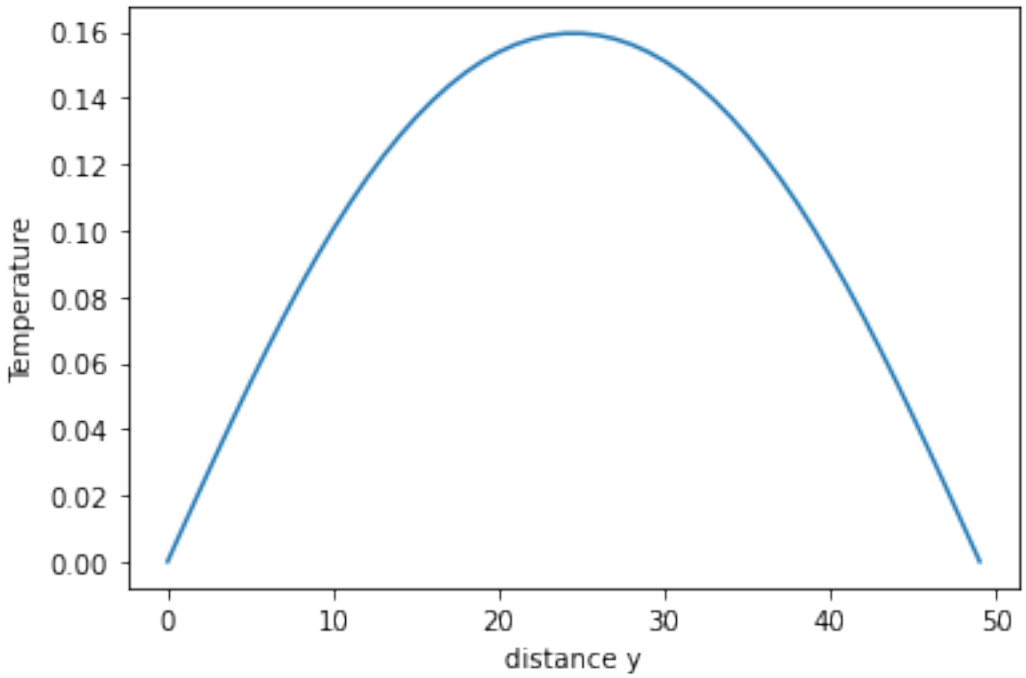
[16]:
```
plt.imshow(T[:,:,25],cmap='jet')
plt.colorbar()
plt.show()
```



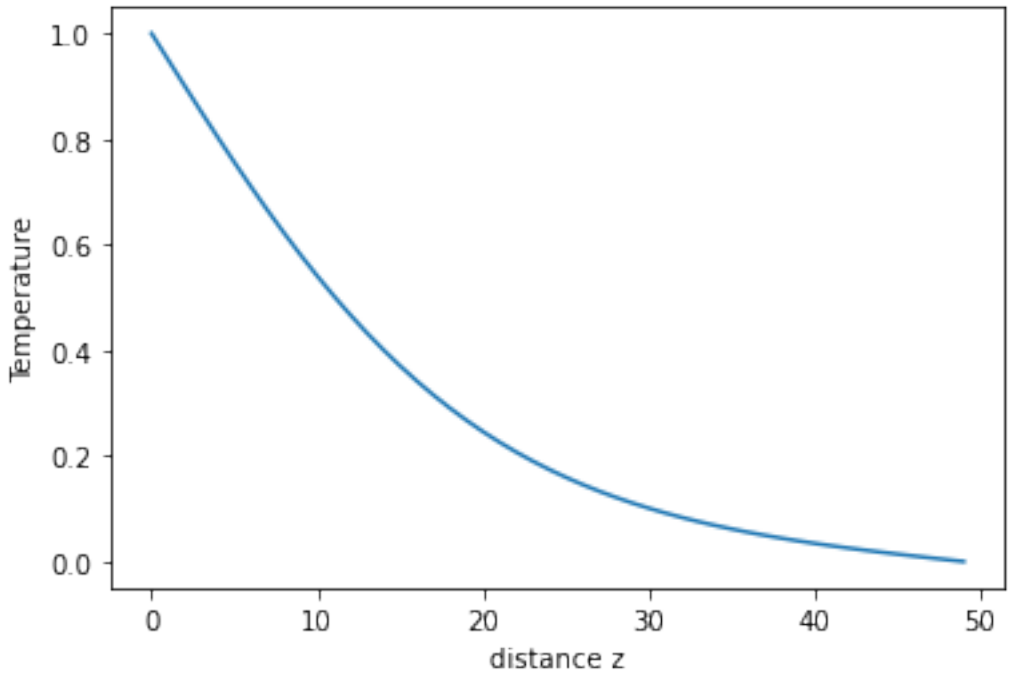
[17]:
```
plt.imshow(T[:,25,:],cmap='jet')
plt.colorbar()
plt.show()
```



[23]:
```
plt.plot(T[25,:,25])
plt.ylabel('Temperature')
plt.xlabel('distance y')
plt.show()
```



[24]:
```
plt.plot(T[:,25,25])
plt.xlabel('distance z')
plt.ylabel('Temperature')
plt.show()
```



[ ]: