FINITE DIFFERENCES METHOD - TYPICAL PROBLEMS (part IV)

STEADY FREE CONVECTION IN A RECTANGULAR CAVITY FILLED WITH A POROUS MEDIUM

Introduction

Natural convective heat transfer in fluid-saturated porous media has occupied the center stage in many fundamental heat transfer analyses and has received a considerable attention over the last several decades. This interest was estimated due to its wide range of applications in, for example, packed sphere beds, high performance insulation for buildings, chemical catalytic reactors, grain storage and such geophysical problems as frost heave. Porous media are also of interest in relation to the underground spread of pollutants, solar power collectors, and to geothermal energy systems.

(D.B. Ingham and I. Pop (eds.), Transport Phenomena in Porous Media, Pergamon, Oxford, Vol. III, 2005)
(D.A. Nield and A. Bejan, Convection in Porous Media (3rd edition), Springer, New York, 2006)
(K. Vafai (ed.), Handbook of Porous Media (2nd edition), Taylor & Francis, New York, 2005)

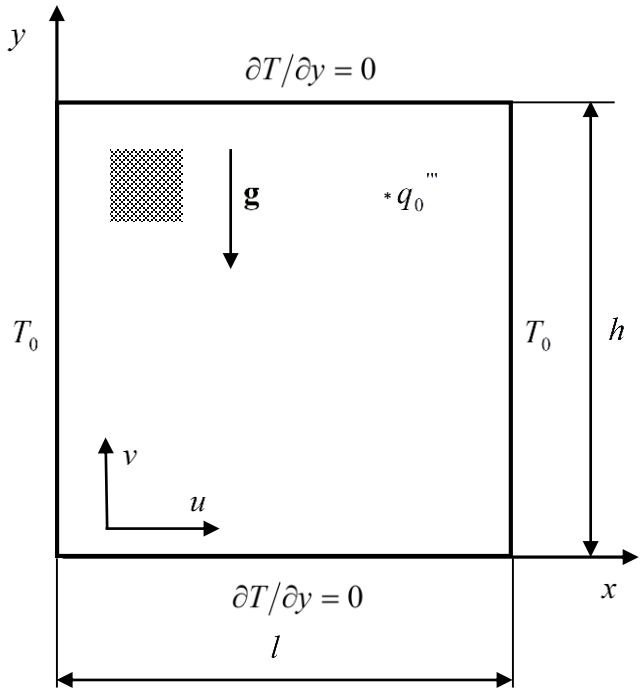
Natural convection in an enclosure in which internal heat generation is present is of prime importance in certain technological applications. Examples are post-accident heat removal in nuclear reactors and geophysical problems associated with the underground storage of nuclear water, among others.

(J.-H. Lee and R.J. Goldstein, ASME J. Heat Transfer 110 (1988), 345-349)

(T. Fusegi, J.M. Hyun and K. Kuwahara, Numer. Heat Transper, Part A 21 (1992) 215-229)

Mathematical model

Consider the steady natural convection flow in a rectangular cavity filled with a fluid-saturated porous medium and an internal heat generation. The geometry and the Cartesian coordinate system are schematically shown in Fig. 1, where the dimensional coordinates x and y are measured along the horizontal bottom wall and normal to it along the left vertical wall, respectively. The height of the cavity is denoted by h and the width by l, respectively. It is assumed that the vertical walls are maintained at a constant temperature T_0, while the horizontal walls are adiabatic.



We also bring into account the effect of a uniform heat generation in the flow region. The constant volumetric rate of heat generation is $q_0'''[W/m^3]$. It is also assumed that the effect of buoyancy is included through the well-known Boussinesq approximation.. The resulting convective flow is governed by the combined mechanism of the driven buoyant force, and internal heat generation.

Under the above assumptions, the conservation equations for mass, Darcy, energy and electric transfer are

$$\nabla \cdot \mathbf{V} = 0 \tag{1}$$

$$\mathbf{V} = \frac{K}{\mu}\left(-\nabla p + \rho \mathbf{g}\right) \tag{2}$$

$$\left(\mathbf{V} \cdot \nabla\right)T = \alpha_m \nabla^2 T + \frac{q_0'''}{\rho_0 c_p} \tag{3}$$

$$\rho = \rho_0\left[1 - \beta\left(T - T_0\right)\right] \tag{4}$$

where V is the velocity vector, T is the fluid temperature, p is the pressure, g is the acceleration vector, K is the permeability of the porous medium, α_m is the effective

thermal diffusivity, ρ is the density, μ is the dynamic viscosity, β is the coefficient of thermal expansion, c_p is the specific heat at constant pressure, ρ_0 is the reference density.

Eliminating the pressure term in Eq. (2) in the usual way, the governing equations (1) to (3) can be written as

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}=0 \tag{5}$$

$$\frac{\partial u}{\partial y}-\frac{\partial v}{\partial x}=-\frac{gK\beta}{\nu}\frac{\partial T}{\partial x} \tag{6}$$

$$u\frac{\partial T}{\partial x}+v\frac{\partial T}{\partial y}=\alpha_m\left(\frac{\partial^2 T}{\partial x^2}+\frac{\partial^2 T}{\partial y^2}\right)+\frac{q_0'''}{\rho c_p} \tag{7}$$

and are subjected to the boundary conditions

$$\begin{aligned} &u=0, \quad T=T_0, \quad \text{at} \quad x=0 \quad \text{and} \quad x=h \\ &v=0, \quad \frac{\partial T}{\partial y}=0, \quad \text{at} \quad y=0 \quad \text{and} \quad y=h \end{aligned} \tag{8}$$

where υ is the kinematic viscosity. Further, we introduce the following non-dimensional variables

$$X = \frac{x}{l}, \quad Y = \frac{y}{h}, \quad U = \frac{h}{\alpha_m} u, \quad V = \frac{l}{\alpha_m} v, \quad \theta = \frac{T - T_0}{\left(q_0''' l^2 / k\right)} \tag{9}$$

where k is the thermal conductivity.

Introducing the stream function ψ defined as $U = \partial\psi / \partial Y$ and $V = -\partial\psi / \partial X$, and using (9) in Eqs. (5) - (8), we obtain the following partial differential equations in non-dimensional form:

$$\frac{\partial^2 \psi}{\partial X^2} + a^2 \frac{\partial^2 \psi}{\partial Y^2} = -Ra \frac{\partial \theta}{\partial X} \tag{10}$$

$$\frac{\partial^2 \theta}{\partial X^2} + a^2 \frac{\partial^2 \theta}{\partial Y^2} + 1 = a\left(\frac{\partial \psi}{\partial Y}\frac{\partial \theta}{\partial X} - \frac{\partial \psi}{\partial X}\frac{\partial \theta}{\partial Y}\right) \tag{11}$$

subject to the boundary conditions

$$\begin{aligned} &U = 0, \quad \psi = 0, \quad \theta = 0, \quad \text{at} \quad X = 0 \quad \text{and} \quad X = 1 \\ &V = 0, \quad \psi = 0, \quad \frac{\partial \theta}{\partial Y} = 0, \quad \text{at} \quad Y = 0 \quad \text{and} \quad Y = 1 \end{aligned} \tag{12}$$

where $a = l/h$ is the aspect ratio of the cavity and *Ra* is the Rayleigh number.

Once we know the numerical values of the temperature function we may obtain the rate of heat flux from each of the vertical walls. The non-dimensional heat transfer rate, q_w, per unit length in the depthwise direction for the left vertical wall is given by

$$Nu_Y = -\left(\frac{\partial\theta}{\partial X}\right)_{X=0}, Nu = -\int_0^1 \left(\frac{\partial\theta}{\partial X}\right)_{X=0} dY \tag{13}$$

Discretization

$$\frac{\Psi_{i+1,j} - 2\Psi_{i,j} + \Psi_{i-1,j}}{(\Delta x)^2} + a^2 \frac{\Psi_{i,j+1} - 2\Psi_{ij} + \Psi_{i,j-1}}{(\Delta y)^2} = -Ra \frac{\theta_{i+1,j} - \theta_{i-1,j}}{2\cdot\Delta x}$$

$$\frac{\theta_{i+1,j} - 2\theta_{ij} + \theta_{i-1,j}}{(\Delta x)^2} + a^2 \frac{\theta_{i,j+1} - 2\theta_{ij} + \theta_{i,j-1}}{(\Delta y)^2} + 1 =$$

$$a\left(\frac{\Psi_{i,j+1} - \Psi_{i,j-1}}{2\cdot\Delta y} \cdot \frac{\theta_{i+1,j} - \theta_{i-1,j}}{2\cdot\Delta x} - \frac{\Psi_{i+1,j} - \Psi_{i-1,j}}{2\cdot\Delta x} \cdot \frac{\theta_{i,j+1} - \theta_{i,j-1}}{2\cdot\Delta y}\right)$$

$i = 2, Nx-1$

$j = 2, Ny-1$

$$\begin{cases} \psi_{ij}\left(\frac{2}{(\Delta x)^2}+\frac{2a^2}{(\Delta y)^2}\right)=\frac{1}{(\Delta x)^2}\left(\tilde{\psi}_{i+1,j}+\tilde{\psi}_{i-1,j}\right)+\frac{a^2}{(\Delta y)^2}\left(\tilde{\psi}_{i,j+1}+\tilde{\psi}_{i,j-1}\right)+\frac{Ra}{2\cdot\Delta x}\left(\tilde{\theta}_{i+1,j}-\tilde{\theta}_{i-1,j}\right) \\ \theta_{ij}\left(\frac{2}{(\Delta x)^2}+\frac{2a^2}{(\Delta y)^2}\right)=1+\frac{1}{(\Delta x)^2}\left(\tilde{\theta}_{i+1,j}+\tilde{\theta}_{i-1,j}\right)+\frac{a^2}{(\Delta y)^2}\left(\tilde{\theta}_{i,j+1}+\tilde{\theta}_{i,j-1}\right)- \\ -\frac{a}{4\,\Delta x\,\Delta y}\left[\left(\tilde{\psi}_{i,j+1}-\tilde{\psi}_{i,j-1}\right)\left(\tilde{\theta}_{i+1,j}-\tilde{\theta}_{i-1,j}\right)-\left(\tilde{\psi}_{i+1,j}-\tilde{\psi}_{i-1,j}\right)\left(\tilde{\theta}_{i,j+1}-\tilde{\theta}_{i,j-1}\right)\right] \end{cases}$$

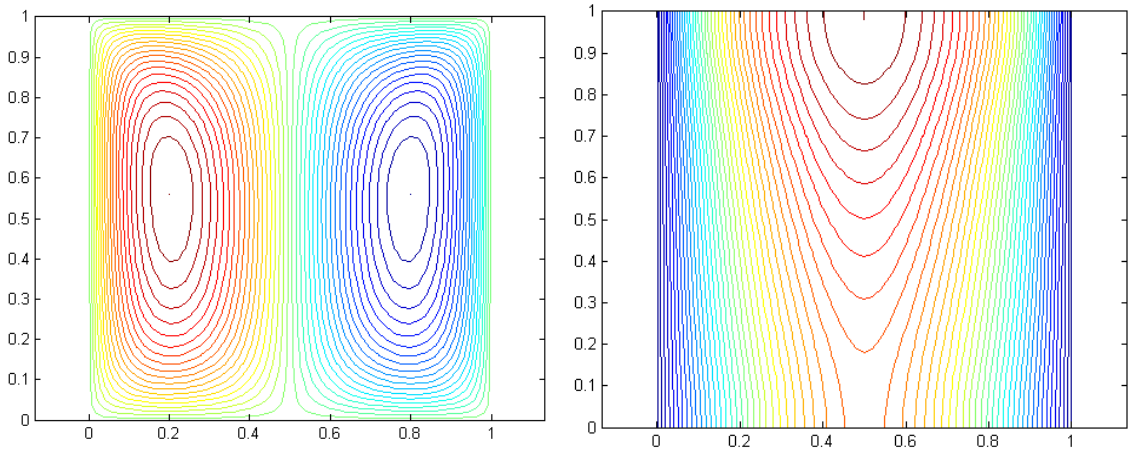
where "~" means the value from the previous iteration.

Matlab program

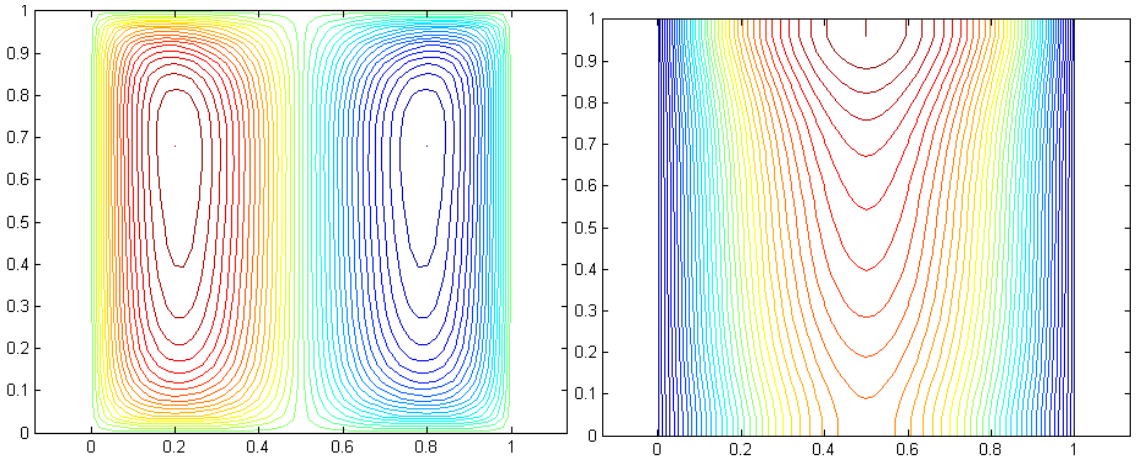
```
format long
%Cavity
%parameters
a=2;
Ra=100;
%discretization and initialization%%%%%%%%%%
N=51; %number of nodes in x- direction
M=(N-1)*a+1;%%number of nodes in y- direction
h=1/(N-1);
k=1/(M-1);
u=zeros(N,M);u_new=u;
T=zeros(N,M);T_new=T;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%coefficients%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
t1=2/(h*h);
t2=2/(k*k)*a*a;
c1=t1/(t1+t2)/2;
c2=t2/(t1+t2)/2;
c4=Ra/2/h/(t1+t2);
c5=k*k/2/(a*a*h*h+k*k);
c6=a*a*h*h/2/(a*a*h*h+k*k);
c7=-a*h*k/8/(a*a*h*h+k*k);
c8=h*h*k*k/2/(a*a*h*h+k*k);
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
nr_it=0;stop=0;

while (stop~=1)
    nr_it=nr_it+1;
    err_u=0;err_T=0;
    for i=2:N-1
        for j=2:M-1
         u_new(i,j)=c1*(u(i+1,j)+u(i-1,j))+c2*(u(i,j+1)+u(i,j-1))+
                    c4*(T(i+1,j)-T(i-1,j));
         T_new(i,j)=c5*(T(i+1,j)+T(i-1,j))+c6*(T(i,j+1)+T(i,j-1))+
                    c7*((u(i,j+1)-u(i,j-1))*(T(i+1,j)-T(i-1,j))-
                       (u(i+1,j)-u(i-1,j))*(T(i,j+1)-T(i,j-1)))+c8;
            if abs(u(i,j)-u_new(i,j))>err_u
                err_u=abs(u(i,j)-u_new(i,j));
            end
            if abs(T(i,j)-T_new(i,j))>err_T
                err_T=abs(T(i,j)-T_new(i,j));
            end
            u(i,j)=u_new(i,j);
            T(i,j)=T_new(i,j);
        end
    end;

    T_new(:,1)=T_new(:,2);T_new(:,M)=T_new(:,M-1); % adiabatic condition;
```

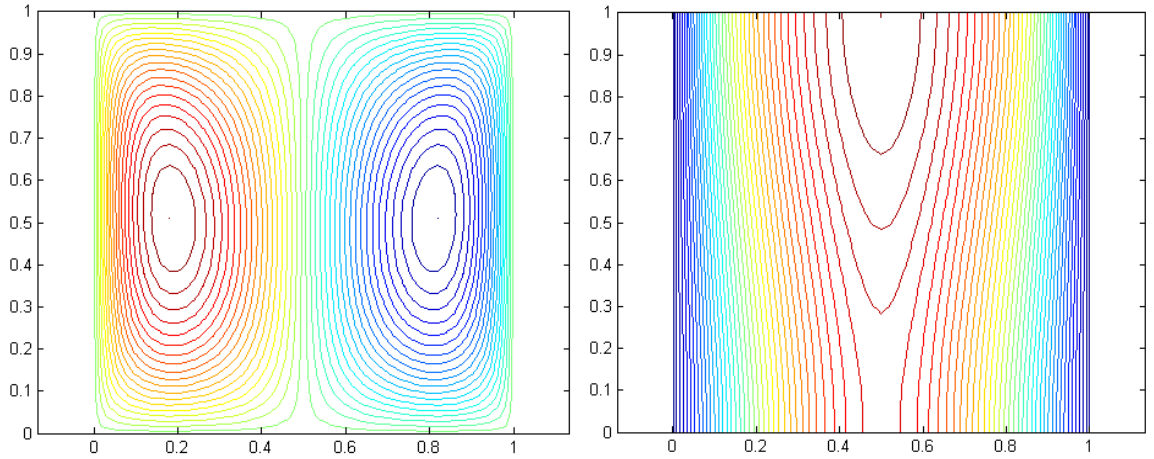
```
%%error evaluation
    if (err_u<1e-9 & err_T<1e-9)
          stop=1;
        end


    u=u_new;
    T=T_new;
%%print intermediate values of the error
   if mod(nr_it,500)==0
    fprintf('nr_it=%g err_u=%g err_T=%g\n', nr_it, err_u, err_T);
   end
end %%while

  nr_it
  max(max(T))
  max(max(u))
  x=0:h:(N-1)*h;y=0:k:(M-1)*k;

 figure(1);
  contour(x,y,u',40)
  axis equal
  figure(2);
  contour(x,y,T',40)
  axis equal
```



Linii de current si izoterme pentru Ra = 100, a=1



Linii de current si izoterme pentru Ra = 100, a=0.5



Linii de current si izoterme pentru Ra = 100, a=2

Table 1
Accuracy test for $Ra = 10^3$ and $a = 1$

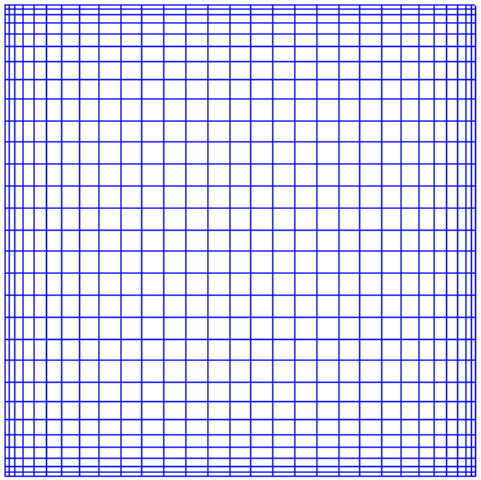
Nodes	$\psi(0.24, 0.24)$	$\theta(0.24, 0.24)$
26×26	2.6368	0.0389
51×51	2.5987	0.0384
101×101	2.5800	0.0382
201×201	2.5707	0.0381
Richardon extrapolation	2.5614	0.0380

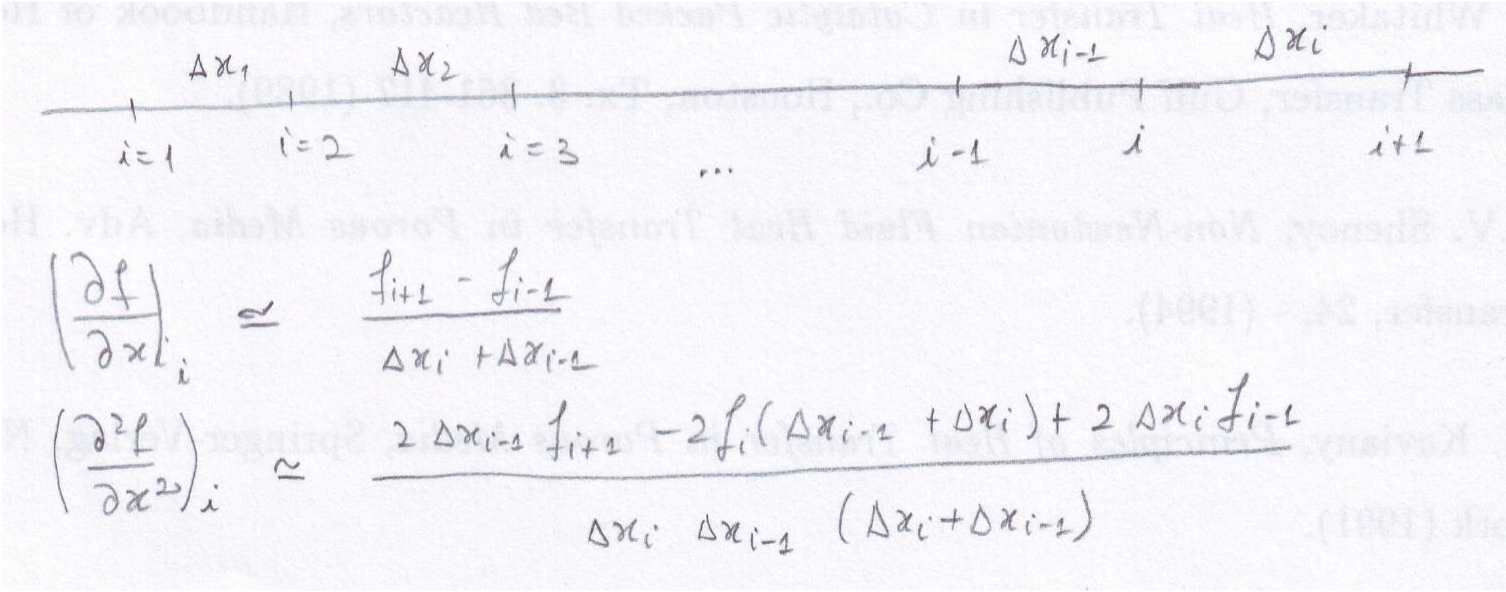
Table 2
Comparison o θ_{max} for $Ha = 0$ and $a = 0.5$

Ra	Haajizadeh *et al.* [27]		Present (Richardson extrapolation)	
	$\psi_{\max}$	$\theta_{\max}$	$\psi_{\max}$	$\theta_{\max}$
10	0.078	0.130	0.079	0.127
10^3	4.880	0.118	4.833(4.832)	0.116(0.116)

(M. Haajizadeh, A.F. Ozguc, C.L. Tien, Natural convection in a vertical porous enclosure with internal heat generation, Int. J. Heat Mass Transfer 27 (1984) 1893-1902)

FINITE DIFFERENCE METHOD. NON-UNIFORM GRID



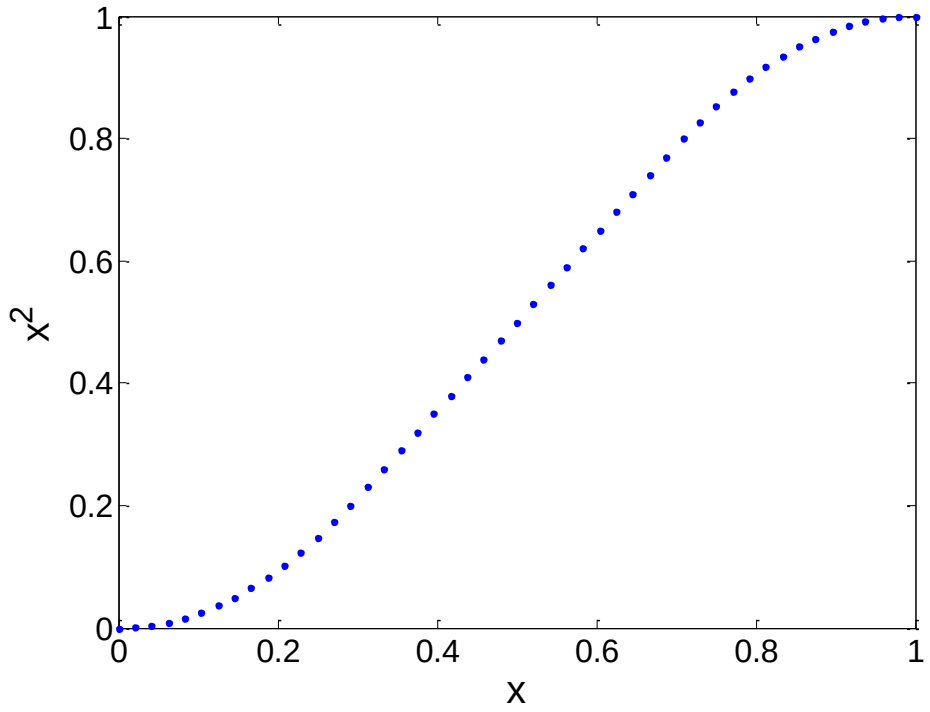


$$\left(\frac{\partial f}{\partial x}\right)_i \simeq \frac{f_{i+1} - f_{i-1}}{\Delta x_i + \Delta x_{i-1}}$$

$$\left(\frac{\partial^2 f}{\partial x^2}\right)_i \simeq \frac{2\Delta x_{i-1} f_{i+1} - 2f_i(\Delta x_{i-1} + \Delta x_i) + 2\Delta x_i f_{i-1}}{\Delta x_i \Delta x_{i-1}(\Delta x_i + \Delta x_{i-1})}$$

How to obtain a non-uniform grid?

$$x_{non-uniform} = f\left(x_{uniform}\right)$$



$$x_{non-uniform} = {x_{uniform}}^{p}, \quad p = 2, 3;$$

```
function [x,y,N]=BLGrid(bl, N_bl)

%x,y grid
%N total number of nodes
% bl boundary layer thickness
%N_bl the number of nodes in the boundary layer region
x1=zeros(N_bl,1);
hunif=(bl^(1/2))/(N_bl-1);
for i=1:N_bl
    x1(i)=((i-1)*hunif)^2;
end
hmin=x1(2)-x1(1);
hmax=(x1(N_bl)-x1(N_bl-1));
Nm=floor(((1-bl-(bl))/hmax))-1;
N=Nm+2*N_bl-1;
h=(1-bl-(bl))/Nm;
x2=zeros(Nm-1,1);
for i=1:Nm-1
    x2(i)=bl+i*h;
end

x3=zeros(N_bl,1);
hunif=(1^(1/2)-(1-bl)^(1/2))/(N_bl-1);
x3(N_bl)=1;
```

```
for i=N_bl-1:-1:1
    x3(i)= 1-x1(N_bl-i+1);
end
xunif=0:1/(N-1):1;

x=[x1;x2;x3];
y=x;
[N,M]=size(x);

figure(1)
plot(xunif,x,'.')

figure(3)
hold on
for i=1:N
plot(x,ones(N,1)*y(i));
end
for i=1:N
plot(ones(N,1)*x(i),y);
end
axis equal
axis off
```

Internal heat generation effects on the unsteady free convection in a square cavity filled with a porous medium

Mathematical model

Consider the unsteady natural convection flow in a rectangular cavity filled with a porous medium and internal heat generation.

It is assumed that the vertical walls are maintained at constant temperatures T_c and T_h, while the horizontal walls are adiabatic. We also take into account the effect of uniform heat generation in the flow region. The constant volumetric rate of heat generation is q_0''' $[W/m^3]$. It is also assumed that the effect of buoyancy is included through the well-known Boussinesq approximation. The resulting convective flow is governed by the combined mechanism of the driven buoyancy force and internal heat generation.

Under the above assumptions, the conservation equations for mass, Darcy and energy are given by

$$\nabla \cdot \mathbf{V} = 0 \tag{1}$$

$$\mathbf{V} = \frac{K}{\mu}\left(-\nabla p + \rho \mathbf{g}\right) \tag{2}$$

$$\frac{\partial T}{\partial t'} + \left(\mathbf{V} \cdot \nabla\right) T = \alpha_m \nabla^2 T + \frac{q_0'''}{\rho_0 c_p} \tag{3}$$

$$\rho = \rho_0 \left[1 - \beta \left(T - T_0\right)\right] \tag{4}$$

$$\begin{aligned} &u = 0, \quad T = T_h \qquad \text{at} \quad x = 0, \qquad 0 < y < L \\ &u = 0, \quad T = T_c \qquad \text{at} \quad x = L, \qquad 0 < y < L \\ &v = 0, \quad \frac{\partial T}{\partial y} = 0, \quad \text{at} \quad y = 0 \quad \text{and} \quad y = L, \quad 0 < x < L \end{aligned} \tag{5}$$

Eliminating the pressure, introducing the following non-dimensional variables

$$t = \frac{\alpha_m}{L^2} t', \quad X = \frac{x}{L}, \quad Y = \frac{y}{L}, \quad U = \frac{L}{\alpha_m} u, \quad V = \frac{L}{\alpha_m} v, \quad \theta = \frac{T - T_0}{T_h - T_c} \tag{5}$$

where $T_0 = (T_h + T_c)/2$ is the characteristic temperature and using the stream function ψ ($U = \partial\psi / \partial Y$ and $V = -\partial\psi / \partial X$) we obtain the dimensionless differential equations:

$$\frac{\partial^2 \psi}{\partial X^2} + \frac{\partial^2 \psi}{\partial Y^2} = -Ra \frac{\partial \theta}{\partial X} \tag{6}$$

$$\frac{\partial \theta}{\partial t} + \frac{\partial \psi}{\partial Y}\frac{\partial \theta}{\partial X} - \frac{\partial \psi}{\partial X}\frac{\partial \theta}{\partial Y} = \frac{\partial^2 \theta}{\partial X^2} + \frac{\partial^2 \theta}{\partial Y^2} + \frac{Ra_I}{Ra} \tag{7}$$

$$\begin{aligned} &\psi = 0, \quad \theta = 1/2, \quad \text{at} \quad X = 0, \quad 0 < Y < 1 \\ &\psi = 0, \quad \theta = -1/2, \quad \text{at} \quad X = 1, \quad 0 < Y < 1 \\ &\psi = 0, \quad \frac{\partial \theta}{\partial Y} = 0, \quad \text{at} \quad Y = 0 \quad \text{and} \quad Y = 1, \quad 0 < X < 1 \end{aligned} \tag{8}$$

where $Ra = gK\beta(T_h - T_c)L/\alpha_m\nu$ is the Rayleigh number and $Ra_I = gq_0'''K\beta L^3/k\alpha_m\nu$ is the heat generation parameter.

The non-dimensional heat transfer rate, per unit length in the depth-wise direction for the left vertical wall is given by

$$Nu_Y = -\left(\frac{\partial\theta}{\partial X}\right)_{X=0}, Nu = -\int_0^1 \left(\frac{\partial\theta}{\partial X}\right)_{X=0} dY \tag{9}$$

Discretization

$$\frac{2\Delta x_{i-1}\,\psi_{i+1,j} - 2\psi_{ij}(\Delta x_i + \Delta x_{i-1}) + 2\Delta x_i\,\psi_{i-1,j}}{\Delta x_i\,\Delta x_{i-1}(\Delta x_i + \Delta x_{i-1})} +$$

$$+ \frac{2\Delta y_{j-1}\,\psi_{i,j+1} - 2\psi_{ij}(\Delta y_j + \Delta y_{j-1}) + 2\Delta y_j\,\psi_{i,j-1}}{\Delta y_j\,\Delta y_{j-1}(\Delta y_j + \Delta y_{j-1})} = -Ra\,\frac{\theta_{i+1,j} - \theta_{i-1,j}}{\Delta x_i + \Delta x_{i-1}} \qquad (a)$$

Implicit in ***Ox***

$$\frac{\theta_{ij}^{(m+1/2)} - \theta_{ij}^{(m)}}{\left(\frac{\Delta t}{2}\right)} + \frac{\psi_{i,j+1}^{(m)} - \psi_{i,j-1}^{(m)}}{\Delta y_j + \Delta y_{j-1}} \cdot \frac{\theta_{i+1,j}^{(m+1/2)} - \theta_{i-1,j}^{(m+1/2)}}{\Delta x_i + \Delta x_{i-1}} - \frac{\psi_{i+1,j}^{(m)} - \psi_{i-1,j}^{(m)}}{\Delta x_i + \Delta x_{i-1}} \cdot \frac{\theta_{i,j+1}^{(m)} - \theta_{i,j-1}^{(m)}}{\Delta y_j + \Delta y_{j-1}} =$$

$$= \frac{2\Delta x_{i-1}\,\theta_{i+1,j}^{(m+1/2)} - 2(\Delta x_i + \Delta x_{i-1})\,\theta_{ij}^{(m+1/2)} + 2\Delta x_i\,\theta_{i-1,j}^{(m+1/2)}}{\Delta x_i\,\Delta x_{i-1}(\Delta x_i + \Delta x_{i-1})} +$$

$$+ \frac{2\Delta y_{j-1}\,\theta_{i,j-1}^{(m)} - 2(\Delta y_{j-1} + \Delta y_j)\,\theta_{i,j}^{(m)} + 2\Delta y_j\,\theta_{i,j-1}^{(m)}}{\Delta y_j\,\Delta y_{j-1}(\Delta y_j + \Delta y_{j-1})} + Q. \qquad (b)$$

Implicit in ***Oy***

$$\frac{\theta_{i,j}^{n+1} - \theta_{i,j}^{(n+1/2)}}{\left(\frac{\Delta t}{2}\right)} + \frac{\psi_{i,j+1}^{(n)} - \psi_{i,j-1}^{(n)}}{\Delta y_j + \Delta y_{j-1}} \cdot \frac{\theta_{i+1,j}^{(n+1/2)} - \theta_{i-1,j}^{(n+1/2)}}{\Delta x_i + \Delta x_{i-1}} - \frac{\psi_{i+1,j}^{(n)} - \psi_{i-1,j}^{(n)}}{\Delta x_i + \Delta x_{i-1}} \cdot \frac{\theta_{i,j+1}^{(n+1)} - \theta_{i,j-1}^{(n+1)}}{\Delta y_j + \Delta y_{j-1}} =$$

$$= \frac{2\Delta x_{i-1}\,\theta_{i+1,j}^{(n+1/2)} - 2(\Delta x_i + \Delta x_{i-1})\,\theta_{i,j}^{(n+1/2)} + 2\Delta x_i\,\theta_{i-1,j}^{(n+1/2)}}{\Delta x_i \cdot \Delta x_{i-1}\,(\Delta x_i + \Delta x_{i-1})} +$$

$$+ \frac{2\Delta y_{j-1}\,\theta_{i,j-1}^{(n+1)} - 2(\Delta y_j + \Delta y_{j-1})\,\theta_{i,j}^{(n+1)} + 2\Delta y_j\,\theta_{i,j-1}^{(n+1)}}{\Delta y_j\,\Delta y_{j-1}\,(\Delta y_j + \Delta y_{j-1})} + Q. \quad CC)$$

Algorithm

1. Solve (b)
2. Solve (c)
3. Solve (a) . Verify if $|\psi - \psi_{old}| < \varepsilon$
4. Next time step.

```
clear all
format long g
tic
%Cavity
%parameters
A=1; %aspect ratio
Ra=10;
RaI=0;
RaE=1;
Q=RaI/RaE;
Ha=0;
g=0;
%discretization and initialization%%%%%%%%%%
```

```
%discretization%%%%%%%%%%
[x,y,N]=BLGrid(0.2,15);
for i=1:N-1
    dx(i)=x(i+1)-x(i);
    dy(i)=y(i+1)-y(i);
end
dx(1)
N %%number of nodes in x- direction

M=N;%%number of nodes in y- direction

dt=dx(1)/5;

uo=zeros(N,M);un=zeros(N,M);%u_old,u_new
uo(2:N-1,2:M-1)=ones(N-2,M-2);

To=zeros(N,M);Ti=zeros(N,M);Tn=zeros(N,M);%T_old,T_intermediar, T_new
To(1,:)=0.5;To(N,:)=-0.5;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%coefficients in energy discretized equation
%coefficients in momentum and energy discretized equations
for i=2:N-1
    c1(i)=2/dx(i)/dx(i-1);
    c2(i)=2/dy(i)/dy(i-1);
    c3(i)=2/dx(i)/(dx(i)+dx(i-1));
```

```
    c4(i)=2/dx(i-1)/(dx(i)+dx(i-1));
    c5(i)=2/dy(i)/(dy(i)+dy(i-1));
    c6(i)=2/dy(i-1)/(dy(i)+dy(i-1));
    c7(i)=Ra/(dx(i)+dx(i-1));
    c8(i)=(dx(i)+dx(i-1));
    c9(i)=(dy(i)+dy(i-1));
end;
a1=c4/2;a3=c1/2;a4=c3/2;a5=A*A*a1;a6=A*A*a3;a7=A*A*a4;

for i=2:N-1
    for j=2:N-1
Ku1(i,j)=c3(i)*(1+Ha*cos(g)*cos(g))/(c1(i)*(1+Ha*cos(g)*cos(g))+c2(j)*(A*A+Ha*sin(g)*s
in(g)));
Ku2(i,j)=c4(i)*(1+Ha*cos(g)*cos(g))/(c1(i)*(1+Ha*cos(g)*cos(g))+c2(j)*(A*A+Ha*sin(g)*s
in(g)));
Ku3(i,j)=c5(j)*(A*A+Ha*sin(g)*sin(g))/(c1(i)*(1+Ha*cos(g)*cos(g))+c2(j)*(A*A+Ha*sin(g)
*sin(g)));
Ku4(i,j)=c6(j)*(A*A+Ha*sin(g)*sin(g))/(c1(i)*(1+Ha*cos(g)*cos(g))+c2(j)*(A*A+Ha*sin(g)
*sin(g)));
Ku5(i,j)=c7(i)/(c1(i)*(1+Ha*cos(g)*cos(g))+c2(j)*(A*A+Ha*sin(g)*sin(g)));
Ku6(i,j)=2*A*Ha*sin(g)*cos(g)/c8(i)/c9(j)/(c1(i)*(1+Ha*cos(g)*cos(g))+c2(j)*(A*A+Ha*si
n(g)*sin(g)));
a2(i,j)=0.5*A/(dx(i)+dx(i-1))/(dy(j)+dy(j-1));
    end
end
```

```
kk=0;
t=0;
tf=0.5;
titer=1;
steady=1;
errT=1;
ur=1.8;
i_Nu=0;
while (errT>1e-6)

t=t+dt;
clear Ax bx
%solve temperature
%implicit in x direction
    for j=2:M-1
    %%%%%%%solve system for each fixed j %%%%%%%%%%%%%%%%

        Ax=zeros(N,N);bx=zeros(N,1);
        %boundary conditions for x=0
        Ax(1,1)=1;
        %bx(1)=0;
        bx(1)=0.5;
        %%%%%%%%%%system%%%%%%%%%%%%%
        for i=2:N-1
```

```
          Ax(i,i-1)=-a1(i)-a2(i,j)*(uo(i,j+1)-uo(i,j-1));
Ax(i,i)=1/dt+a3(i); Ax(i,i+1)=-a4(i)+a2(i,j)*(uo(i,j+1)-uo(i,j-1));
bx(i)=To(i,j-1)*(a5(j)-a2(i,j)*(uo(i+1,j)-uo(i-1,j)))+To(i,j)*
(1/dt-a6(j))+To(i,j+1)*(a7(j)+a2(i,j)*(uo(i+1,j)-uo(i-1,j)))+0.5*Q;
        end
        %boundary conditions for x=1
        Ax(N,N)=1;
        %bx(N)=0;
        bx(N)=-0.5;

        Ti(:,j)=Ax\bx;
    end;

    clear Ax bx
%implicit in y direction
    for i=2:N-1
    %%%%%%%solve system for each fixed i %%%%%%%%%%%%%%%

        Ax=zeros(M,M);bx=zeros(M,1);
        %boundary conditions for y=0
        Ax(1,1)=1;Ax(1,2)=-1;bx(1)=0;
        %%%%%%%%%system%%%%%%%%%%%%%
        for j=2:M-1
         Ax(j,j-1)=-a5(j)+a2(i,j)*(uo(i+1,j)-uo(i-1,j));Ax(j,j)=1/dt+a6(j);
Ax(j,j+1)=-a7(j)-a2(i,j)*(uo(i+1,j)-uo(i-1,j));
```

```
bx(j)=Ti(i-1,j)*(a1(i)+a2(i,j)*(uo(i,j+1)-uo(i,j-1)))+Ti(i,j)*
(1/dt-a3(i))+Ti(i+1,j)*(a4(i)-a2(i,j)*(uo(i,j+1)-uo(i,j-1)))+0.5*Q;
        end
        %boundary conditions for x=0
        Ax(M,M-1)=-1;Ax(M,M)=1;bx(M)=0;
        Tn(i,:)=(Ax\bx)';
    end;
   Tn(1,:)=0.5;Tn(N,:)=-0.5;
  errT=norm(To-Tn);
%   steady=abs(max(max(To-Tn))/max(max(Tn)));
   To=Tn;

     %solve stream
     nr_it=0;err_u=1;
   while (err_u>1e-6)
   nr_it=nr_it+1;
   err_u=0;
    for i=2:N-1
        for j=2:M-1
un(i,j)=Ku1(i,j)*uo(i+1,j)+Ku2(i,j)*uo(i-1,j)+
        Ku3(i,j)*uo(i,j+1)+Ku4(i,j)*uo(i,j-1)+
        Ku5(i,j)*(To(i+1,j)-To(i-1,j))+
        Ku6(i,j)*(uo(i+1,j+1)-uo(i+1,j-1)-uo(i-1,j+1)+uo(i-1,j-1));
un(i,j)=uo(i,j)+ur*(un(i,j)-uo(i,j));
```

```
            if abs(uo(i,j)-un(i,j))>err_u
                  err_u=abs(uo(i,j)-un(i,j));
              end%if
              uo(i,j)=un(i,j);
          end%for j
      end;%for i

     uo=un;

%     if mod(nr_it,10)==0
%      fprintf('nr_it=%g err_u=%g err_T=%g\n', nr_it, err_u, errT);
%     end
   end %while stream

   titer=titer+1;
      if mod(titer,10)==0
         kk=kk+1;
        i_Nu=i_Nu+1;
        timp(i_Nu)=t;
        Nuloc=(Tn(2,:)-Tn(1,:))/dx(1);
        Nu(i_Nu)=trapz(y,Nuloc);
         fprintf('\nt=%g  errT=%g \n', t, errT);
      end

end%while main
```

```
Nuloc=(Tn(2,:)-Tn(1,:))/dx(1);
i_Nu=i_Nu+1;
timp(i_Nu)=t;
Nuloc=(Tn(2,:)-Tn(1,:))/dx(1);
Nu(i_Nu)=trapz(y,Nuloc); %using trapezoidal rule

figure(1)
 contour(x,y,To',20)
 axis equal
 axis([0,1,0,1])

 figure(2)
 contour(x,y,uo',20)
 axis equal
 axis([0,1,0,1])
toc
```

Grid dependence

Table 1. Mean Nusselt number $\overline{Nu}$ for different grids at $Ra = 10^3$

Boundary layer thickness	Δx_1- first step in b.l.	N_{bl} - number of nodes in b.l.	N – total number of nodes	$\overline{Nu}$ (Nusselt number)
0.1	0.00123	10	56	13.4523
0.1	0.00027	20	116	13.5982
0.2	0.00054	20	67	13.5521
0.2	0.00023	30	102	13.6027
0.3	0.00035	30	77	13.5822
0.3	0.00019	40	104	13.6085
0.4	0.00026	40	87	13.5968
0.4	0.00016	50	110	13.6131

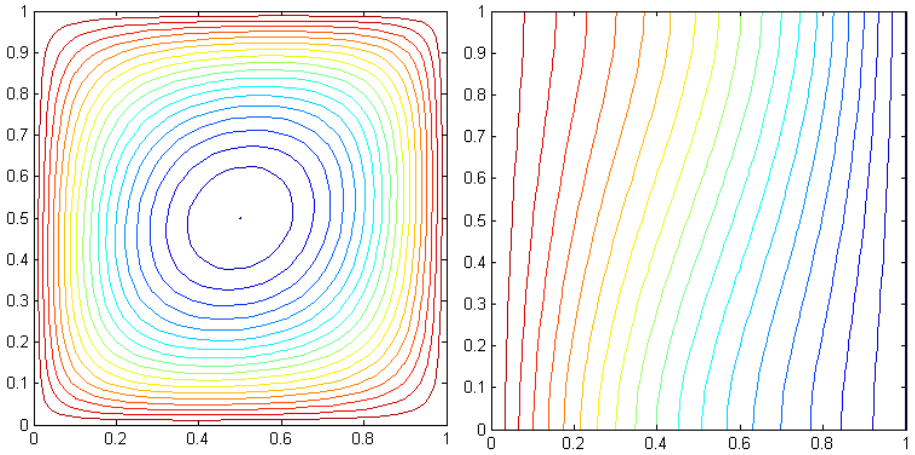
Validation

We use a non-uniform grid and in the absence of internal heat generation effect we obtain:

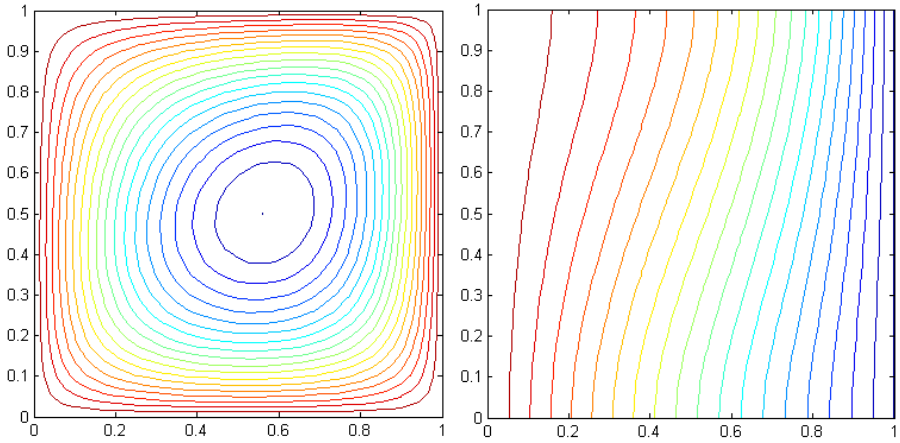
Table 2. Comparison of the mean Nusselt number $\overline{Nu}$ for different values of Ra when the steady state is reached

Authors	Ra			
	10	100	1000	10000
Walker and Hosmy [20]		3.097	12.96	51
Bejan [21]		4.2	15.8	50.80
Beckerman et al. [22]		3.113		48.9
Gross et al. [23]		3.141	13.448	42.583
Manole and Lage [24]		3.118	13.637	48.117
Moya et al. [25]	1.065	2.801		
Batas and Pop [26]	1.079	3.16	14.06	48.33
Present results (110 x 110)	1.079	3.108	13.613	48.208

Results and discussion



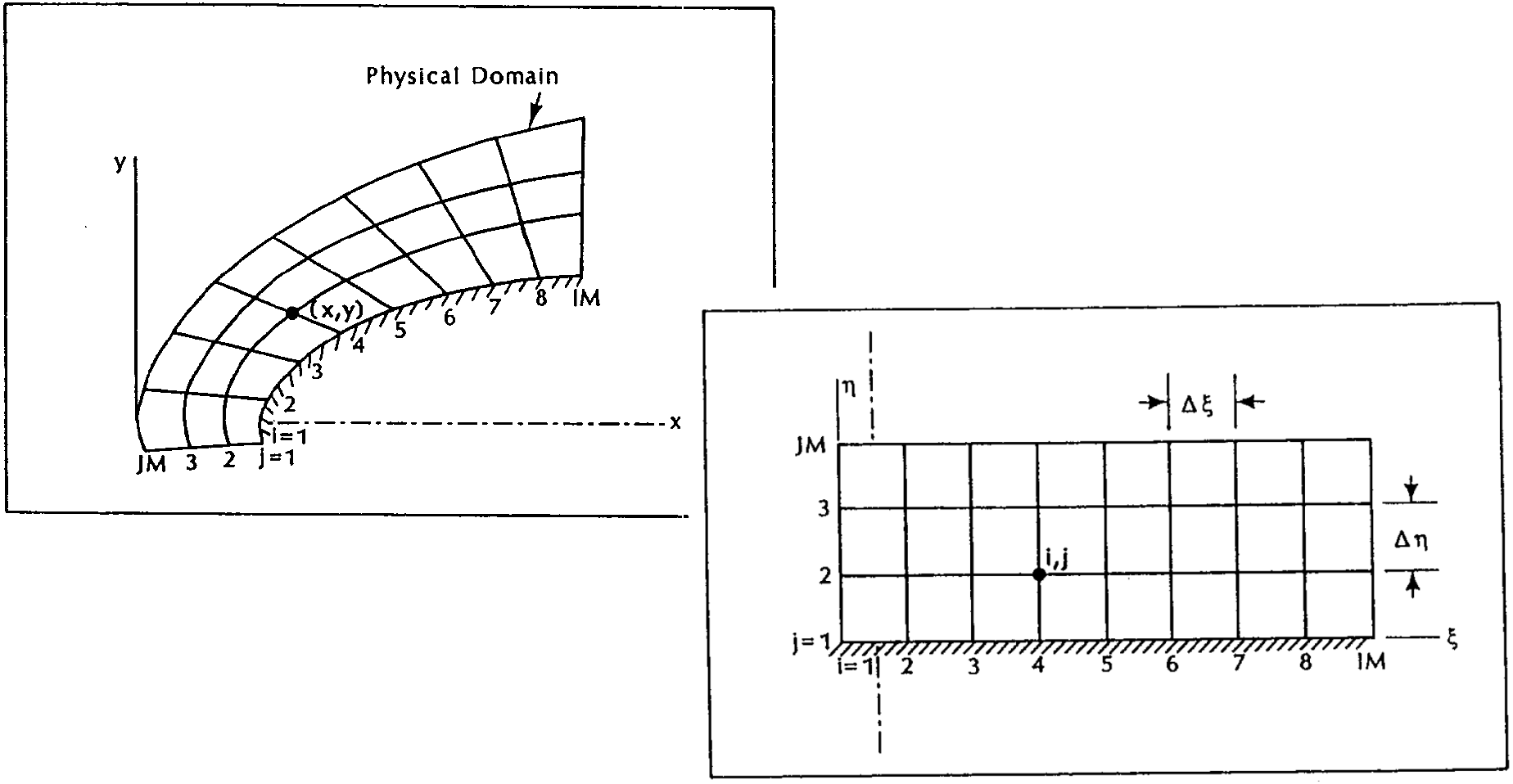
Streamlines and isotherms for Ra = 10 and Q = 0 (steady state).



Streamlines and isotherms for Ra = 10 and Q = 1 (steady state).

Curvilinear boundaries

Domain transformation



Computational domain with constant stepsizes $\Delta\xi$ and $\Delta\eta$.

$$\xi = \xi(x, y)$$

$$\eta = \eta(x, y)$$

The chain rule for partial differentiation yields the following expression:

$$\frac{\partial}{\partial x} = \frac{\partial \xi}{\partial x}\frac{\partial}{\partial \xi} + \frac{\partial \eta}{\partial x}\frac{\partial}{\partial \eta}$$

or

$$\frac{\partial}{\partial x} = \xi_x \frac{\partial}{\partial \xi} + \eta_x \frac{\partial}{\partial \eta}$$

$$\frac{\partial}{\partial y} = \xi_y \frac{\partial}{\partial \xi} + \eta_y \frac{\partial}{\partial \eta}$$

Now consider a model PDE, such as

$$\frac{\partial u}{\partial x} + a\frac{\partial u}{\partial y} = 0$$

This equation may be transformed from physical space to computational

$$\xi_x \frac{\partial u}{\partial \xi} + \eta_x \frac{\partial u}{\partial \eta} + a\left(\xi_y \frac{\partial u}{\partial \xi} + \eta_y \frac{\partial u}{\partial \eta}\right) = 0$$

which may be rearranged as

$$(\xi_x + a\xi_y)\frac{\partial u}{\partial \xi} + (\eta_x + a\eta_y)\frac{\partial u}{\partial \eta} = 0$$

The obtained equation for the computational domain is more complicated!

Boundary discretization

COMMUNICATIONS IN MATH SCI
Vol. 1, No. 1, pp. 181–197

A FAST FINITE DIFFERENCE METHOD FOR SOLVING NAVIER-STOKES EQUATIONS ON IRREGULAR DOMAINS*

ZHILIN LI† AND CHENG WANG‡

4.1. A Review of Local Vorticity Boundary Formula in a Rectangular Domain Let us assume that Ω is a rectangle for the moment. The Dirichlet boundary condition $\psi = 0$ on $\partial\Omega$ is used to solve the stream-function via the vorticity obtained from (2.10). Yet the normal boundary condition, $\frac{\partial \psi}{\partial \boldsymbol{n}} = 0$, cannot be enforced directly. The way to overcome this difficulty is to convert it into the boundary condition for the vorticity. We use $\psi|_{\partial\Omega} = 0$ and $\frac{\partial \psi}{\partial \boldsymbol{n}} = 0$ to approximate the vorticity on the boundary. Take the bottom part of $\partial\Omega$, where the subscript j is zero, for example, we use the central finite difference scheme to approximate the Laplacian, which is simply $D_y^2\psi$ since $D_x^2\psi = 0$ from the boundary condition $\psi = 0$.

The second-order finite difference approximation yields

$$\omega_{i,0} = D_y^2 \psi_{i,0} = \frac{\psi_{i,1} + \psi_{i,-1} - 2\psi_{i,0}}{h^2} = \frac{2\psi_{i,1}}{h^2} - \frac{2}{h}\frac{\psi_{i,1} - \psi_{i,-1}}{2h}, \qquad (4.1)$$

where $\psi_{i,0} = 0$ and $(i, -1)$ refers to a "ghost" grid point outside of the computational domain. Since

$$\frac{\psi_{i,1} - \psi_{i,-1}}{2h} \approx \frac{\partial \psi}{\partial \mathbf{n}} + O(h^2) = O(h^2),$$

we have $\psi_{i,-1} = \psi_{i,1} + O(h^3)$ which leads to **Thom's formula**

$$\omega_{i,0} = \frac{2\psi_{i,1}}{h^2}\,. \qquad (4.2)$$

The vorticity on the boundary can also be determined by other approximations to $\psi_{i,-1}$. For example, using a third-order one-sided finite difference scheme to approximate the normal boundary condition $\frac{\partial \psi}{\partial \boldsymbol{n}} = 0$, we can write

$$\begin{aligned} (\partial_y \psi)_{i,0} &= \frac{-\psi_{i,-1} + 3\psi_{i,1} - \frac{1}{2}\psi_{i,2}}{3h} = 0 + O(h^3)\,, \quad \text{which leads to} \\ \psi_{i,-1} &= 3\psi_{i,1} - \tfrac{1}{2}\psi_{i,2} + O(h^4). \end{aligned} \qquad (4.3)$$

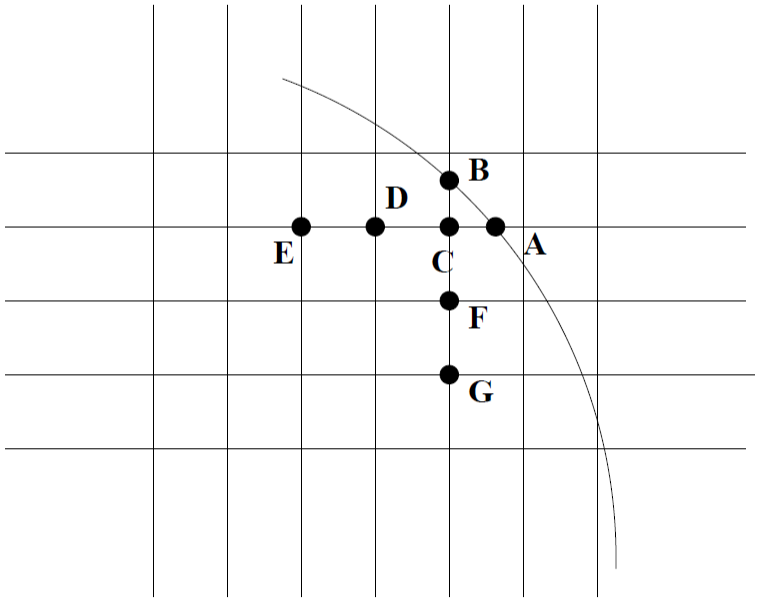
Plugging this back to the difference vorticity formula $\omega_{i,0} = \frac{1}{h^2}(\psi_{i,1} + \psi_{i,-1})$ in (4.1), we have **Wilkes-Pearson's formula**

$$\omega_{i,0} = \frac{1}{h^2}(4\psi_{i,1} - \frac{1}{2}\psi_{i,2})\,. \tag{4.4}$$

This formula is second-order accurate for the vorticity on the boundary.

4.2. The Extension to a Curved Domain The extension of the above methodology to a domain with a curved boundary is similar but a little more complicated.

Fig. 4.1



We use FIG. 4.1 as an illustration. FIG. 4.1 shows several grid points near the boundary $\partial\Omega$. In FIG. 4.1, D, E, F, G are regular grid points, AB is an arc section on the boundary. Special attention is needed at the grid points close to the boundary, such as the point C. We denote $a = |AC|/h$, $b = |BC|/h$. Note that $0 < a, b < 1$.

The combination of the Dirichlet boundary condition $\psi\,|_{\partial\Omega} = 0$ and the Neumann boundary condition $\frac{\partial\psi}{\partial \boldsymbol{n}} = 0$ implies that

$$\nabla\psi = 0\,, \quad \text{at} \quad A\,. \tag{4.5}$$

In other words, the partial derivative of the stream function along any direction is zero on the boundary. This can also be seen by the fact that both $u = -\partial_y\psi$ and $v = \partial_x\psi$ vanish on the boundary.

The local Taylor expansion at the boundary point A gives

$$\psi_C = \frac{a^2h^2}{2}(\partial_x^2\psi_A) - \frac{a^3h^3}{6}(\partial_x^3\psi_A) + O(h^4)\,, \tag{4.6}$$

$$\psi_D = \frac{(1+a)^2h^2}{2}(\partial_x^2\psi_A) - \frac{(1+a)^3h^3}{6}(\partial_x^3\psi_A) + O(h^4)\,, \tag{4.7}$$

where the information of $\psi = 0$ and $\partial_x\psi = 0$ at point A was used in the derivation for (4.6) and (4.7).

The Taylor expansion (4.6) gives a first order approximation to $\partial_x^2\psi$ at the boundary point A

$$(\partial_x^2\psi)_A = \frac{2}{a^2h^2}\psi_C + O(h)\,, \tag{4.8}$$

which corresponds to the Thom's formula (4.2). Or the combination of (4.6) and (4.7) gives a second-order approximation to $\partial_x^2\psi$ at point A

$$(\partial_x^2\psi)_A = \frac{2}{a^2h^2}\left((1+a)\psi_C - \frac{a^3}{(1+a)^2}\psi_D\right) + O(h^2)\,, \tag{4.9}$$

which corresponds to the Wilkes' formula (4.4).