

---

```

% Shivam_Swarnakar_184106011_ME704
% MATLAB code for Stream-function vorticity method-based solution of
  2D incompressible flow

clc;
clear;

imax = 51;
jmax = 51;

nu = 0.01;

si(1:jmax,1:imax) = 0;
omega(1:jmax,1:imax) = 0;
u(1:jmax,1:imax) = 0;
v(1:jmax,1:imax) = 0;

si_old = si;
omega_old = omega;
u_old = u;
v_old = v;

u_0 = 1;
u(jmax,:) = u_0;
L = 1;
delx = L/(imax-1);
dely = L/(jmax-1);

eps = 0.001;
err = 1;

si_relax = 1;
omega_relax = 1;

n = 0;
while(abs(err) >= eps)

    for j = 2:jmax-1
        for i = 2:imax-1
            u(j,i) = (si(j+1,i)-si(j-1,i))/(2*dely);
            v(j,i) = -(si(j,i+1)-si(j,i-1))/(2*delx);

            si(j,i) = 0.25*(si(j+1,i)+si(j-1,i)+si(j,i
+1)+si(j,i-1)+(delx^2)*omega(j,i));
            si(j,i) = si_old(j,i) + si_relax*(si(j,i)-si_old(j,i));

            omega(jmax,:) = -(2*u_0*dely + 2*si(jmax-1,:))/
(dely^2); %TOP BC
            omega(1,:) = -2*(si(2,:)/(dely^2)); %BOTTOM BC
            omega(:,imax) = -(2*si(:,imax-1))/(delx^2); %LEFT BC
            omega(:,1) = -(2*si(:,2))/(delx^2); %RIGHT BC

```

---

---

```

        aE = 1 - ((min(u(j,i),0)*delx)/nu);
        aW = 1 + ((max(u(j,i),0)*delx)/nu);
        aN = 1 - ((min(v(j,i),0)*dely)/nu);
        aS = 1 + ((max(v(j,i),0)*dely)/nu);
        aP = aE+aW+aN+aS;

        omega(j,i) = (aE*omega(j,i+1) + aW*omega(j,i-1) +
aN*omega(j+1,i) + aS*omega(j-1,i))/aP;
        omega(j,i) = omega_old(j,i) + omega_relax*(omega(j,i)-
omega_old(j,i));
    end
end

error_si = si -si_old;
err_si = max(max(error_si));

error_omega = omega -omega_old;
err_omega = max(max(error_omega));

error_u = u -u_old;
err_u = max(max(error_u));

error_v = v -v_old;
err_v = max(max(error_v));

u_old = u;
v_old = v;
si_old = si;
omega_old = omega;

err1 = max(err_u,err_v);
err2 = max(err_omega,err_si);
err = max(err1,err2);
n = n+1;

end

file_name_u = ['u_',num2str(imax),'.mat'];
file_name_v = ['v_',num2str(imax),'.mat'];

save(file_name_u,'u');
save(file_name_v,'v');

var = cell(1,4);
var{1,1} = u;
var{1,2} = v;
var{1,3} = si;
var{1,4} = omega;

var_name = ["u","v","si","omega"];

x = linspace(0,1,imax);
y = linspace(0,1,jmax);
for i= 1:4

```

---

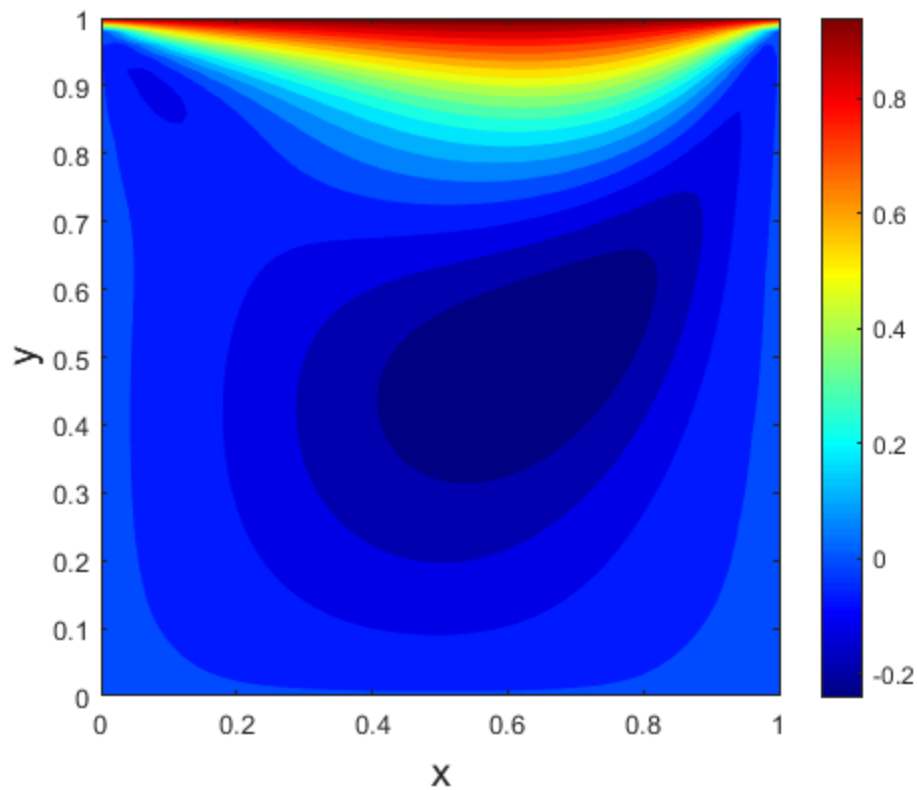
---

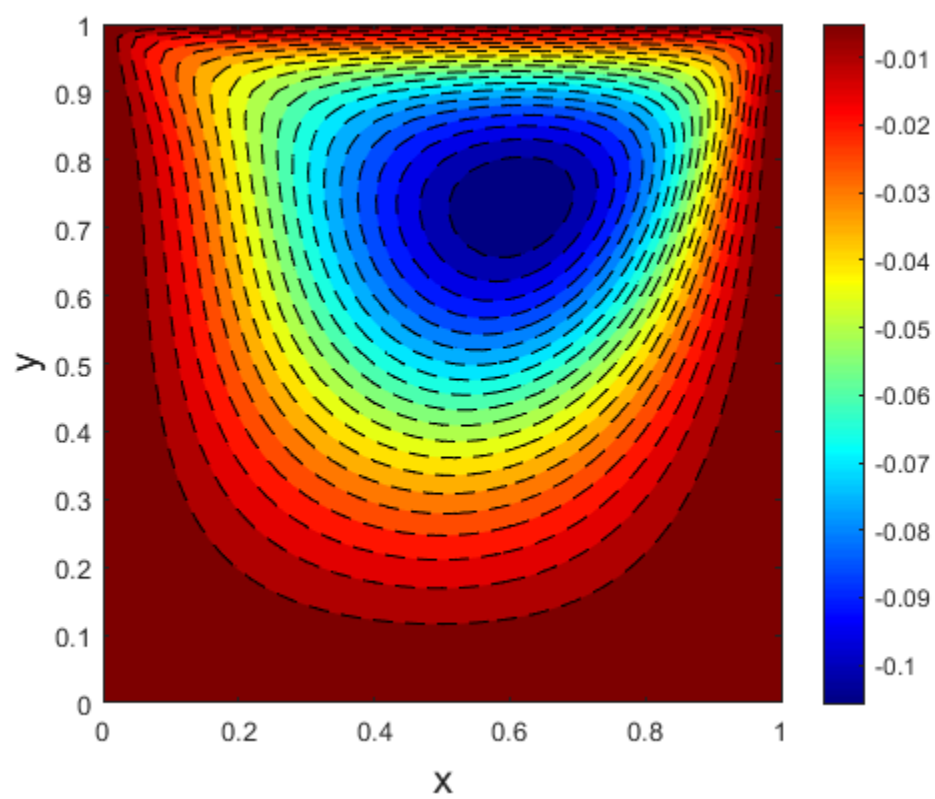
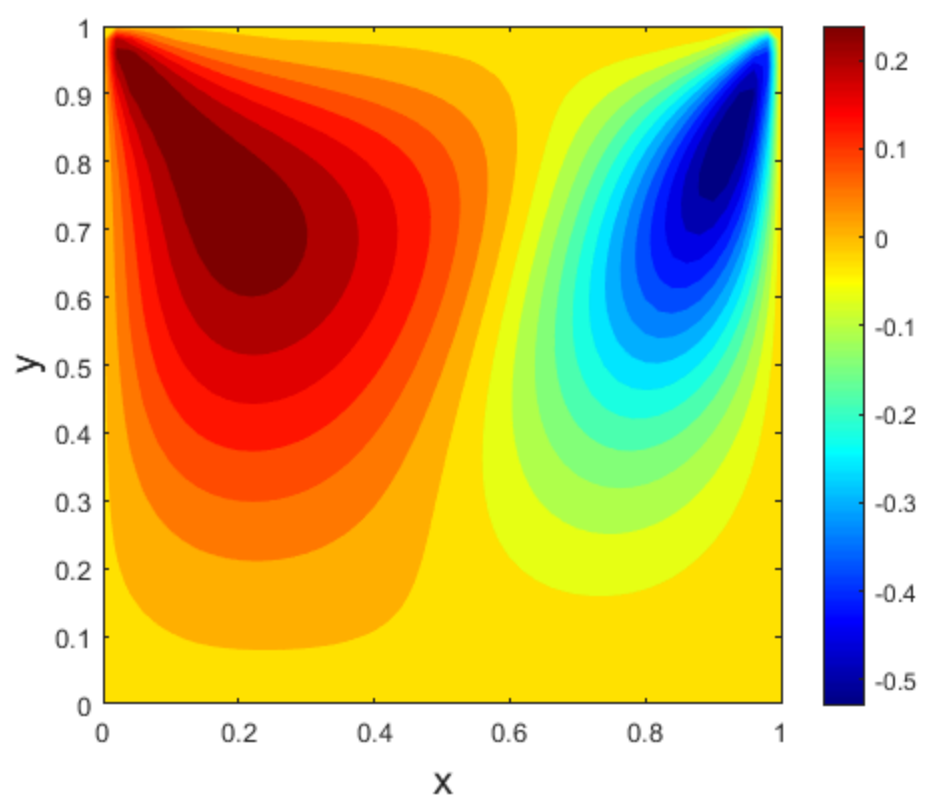
```

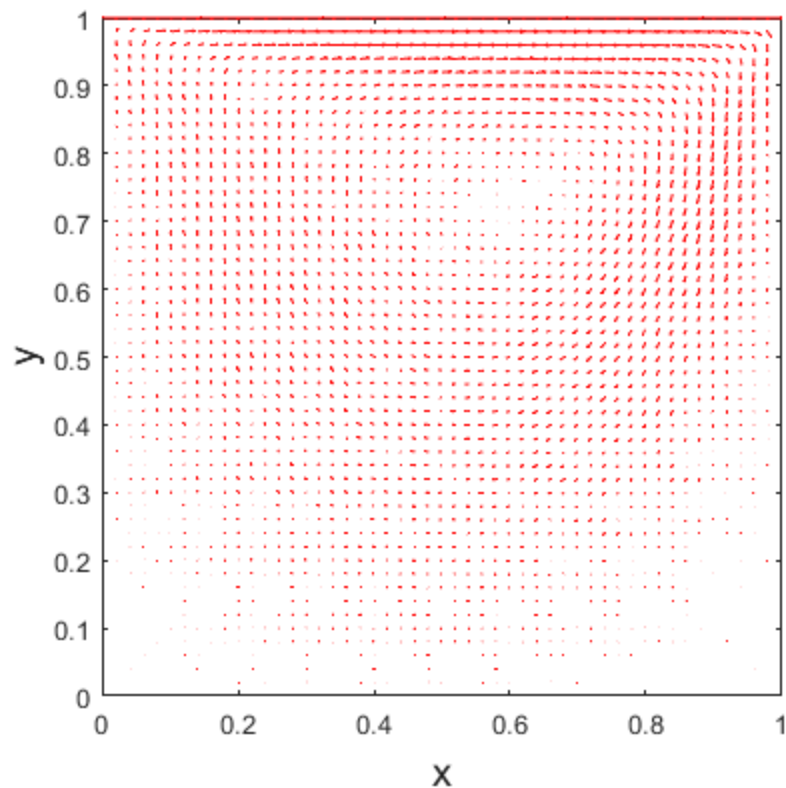
figure;
if i == 3
    contourf(x,y,var{1,i}(:,,:),20,'LineStyle','--');
else
    contourf(x,y,var{1,i}(:,,:),20,'LineStyle','none');
end
colormap('jet');
colorbar;
xlabel('x','FontSize',15);
ylabel('y','FontSize',15);
axis equal;
name = (var_name(i)+'_'+num2str(imax)+'_contour_plot.png');
saveas(gcf,name);
end

quiver(x,y,u,v,'LineWidth',1,'Color','r');
axis equal;
xlabel('x','FontSize',15);
ylabel('y','FontSize',15);
xlim([0,1]);
ylim([0,1]);
% name_vec_plt = ('Vector_'+num2str(imax)+'_plot.png');
% saveas(gcf,name_vec_plt);

```







## Grid test

```
load ghia_data_YU_Re100.dat.txt
load ghia_data_XV_Re100.dat.txt

data_u = cell(1,4);
data_u1 = load('u_11.mat');
data_u{1,1} = data_u1.u(:,6);
data_u2 = load('u_31.mat');
data_u{1,2} = data_u2.u(:,15);
data_u3 = load('u_51.mat');
data_u{1,3} = data_u3.u(:,25);
data_u{1,4} = ghia_data_YU_Re100_dat(:,2);

data_v = cell(1,4);
data_v1 = load('v_11.mat');
data_v{1,1} = data_v1.v(6,:);
data_v2 = load('v_31.mat');
data_v{1,2} = data_v2.v(15,:);
data_v3 = load('v_51.mat');
data_v{1,3} = data_v3.v(25,:);
data_v{1,4} = ghia_data_XV_Re100_dat(:,2);

data_L = cell(1,3);
data_L{1,1} = linspace(0,1,11);
```

---

```

data_L{1,2} = linspace(0,1,31);
data_L{1,3} = linspace(0,1,51);
data_L{1,4} = ghia_data_YU_Re100_dat(:,1);

l_color = ["-r","-b","-g","-ok"];

for i =1:4

    plot(data_u{1,i}(),data_L{1,i},l_color(i),'LineWidth',2);
    hold on;
    axis equal;
    xlabel('U','FontSize',15);
    ylabel('y','FontSize',15);
    ylim([0,1]);
    axis equal;
    set(gca,'FontSize',15);
    legend('11\times11 grid','31\times31 grid','51\times
51grid','Benchmark result','Position',[0.7,0.55,0,0]);
    legend box off;
    saveas(gcf,'grid_test_YU.png');
end
figure;
data_L{1,4} = ghia_data_XV_Re100_dat(:,1);

for i =1:4

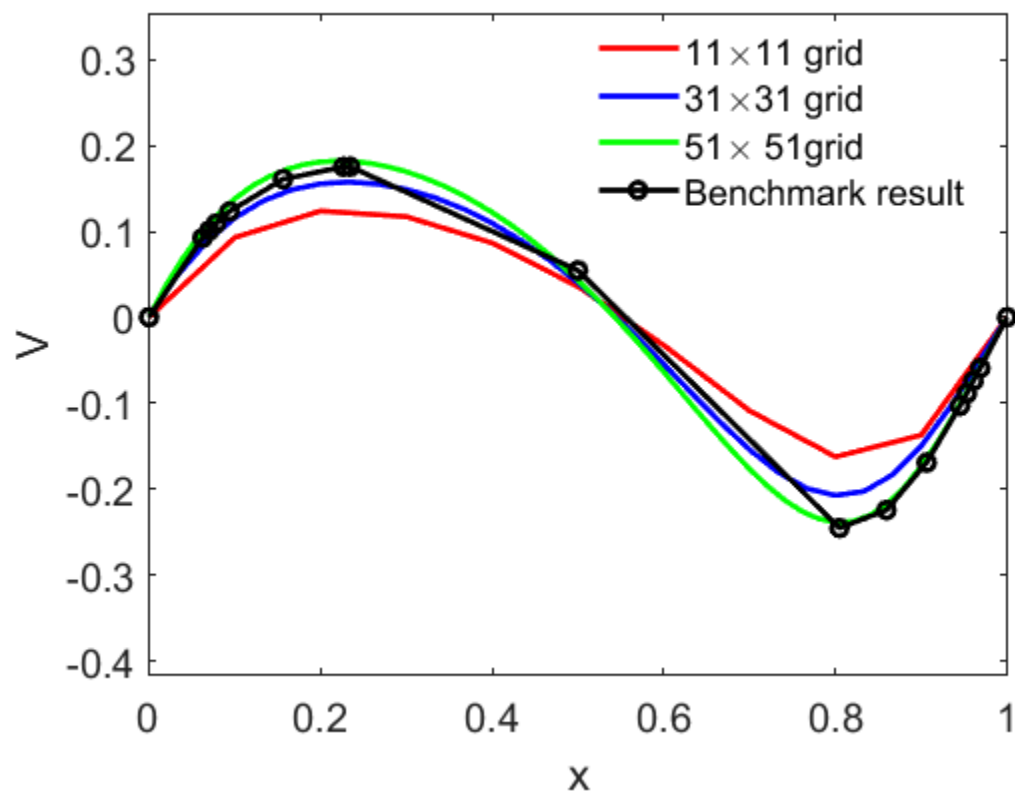
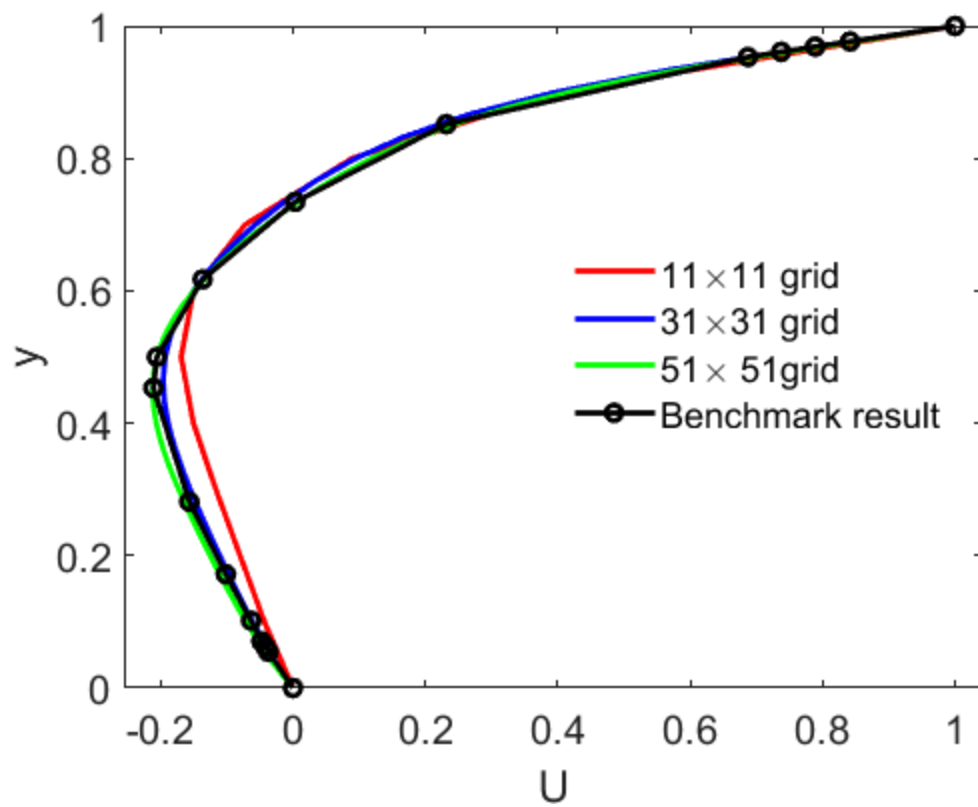
    plot(data_L{1,i},data_v{1,i}(),l_color(i),'LineWidth',2);
    hold on;
    axis equal;
    xlabel('x','FontSize',15);
    ylabel('V','FontSize',15);
    xlim([0,1]);
    axis equal;
    set(gca,'FontSize',15);
    legend('11\times11 grid','31\times31 grid','51\times
51grid','Benchmark result','Position',[0.7,0.8,0,0]);
    legend box off;
    saveas(gcf,'grid_test_XV.png');
end

```

```

Warning: Ignoring extra legend entries.
Warning: Ignoring extra legend entries.
Warning: Ignoring extra legend entries.
Warning: Ignoring extra legend entries.
Warning: Ignoring extra legend entries.
Warning: Ignoring extra legend entries.

```



---

*Published with MATLAB® R2020b*