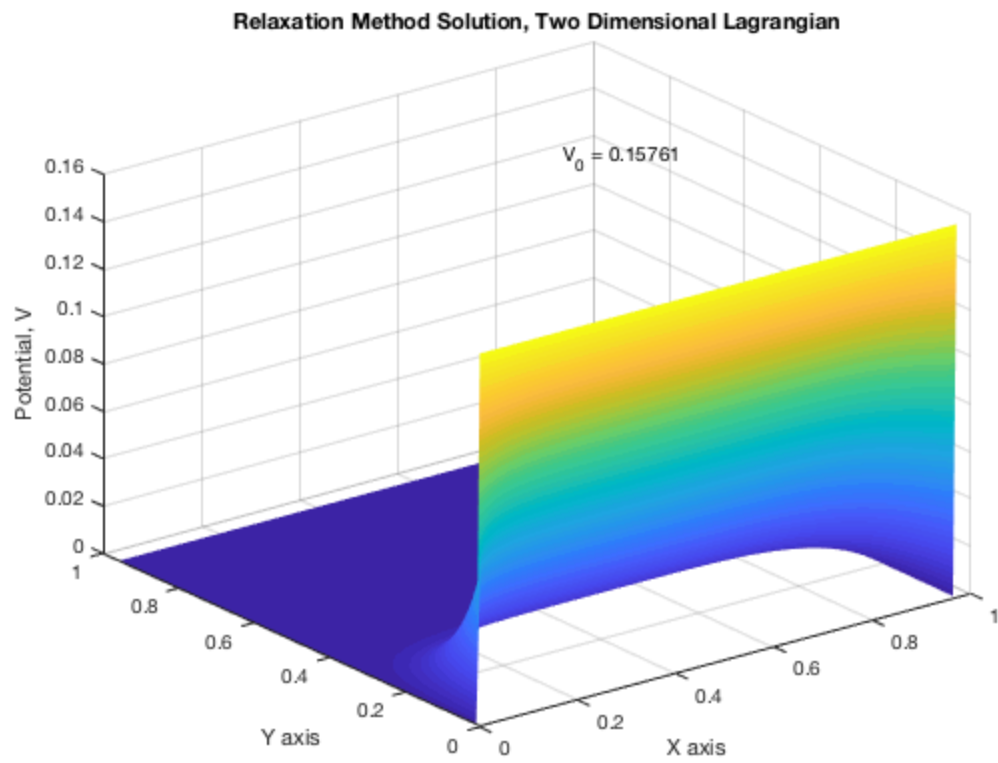


---

```
xsize = 100; %size of matrix
ysize = 100;
temp = zeros(xsize,ysize); %for storing and assigning updated values
v = zeros(xsize,ysize); %final potential solution
v0 = rand; %the potential v naught, randomly generated between 0 & 1
xmax = rand; %x coord of boundary (width of shape/area)
ymax = rand; %y coord of boundary (height of shape/area)
n = 500; %number of iterations
dx = xmax/(xsize-1); %array spacing
dy = ymax/(ysize-1);
x = 0:dx:xmax;
y = 0:dy:ymax;
temp(:,1) = 0; %implement boundary conditions
temp(:,ysize) = 0;
temp(1,:) = v0;
temp(xsize,:) = temp(xsize-1,:);
i = 2:xsize-1; %define indexing integers for iteration
j = 2:ysize-1;
for it=1:n
    v = temp;
    temp(i,j) = (v(i+1,j) + v(i-1,j) + v(i,j+1) + v(i,j-1))/4;
    %average neighbors
    temp(:,1) = 0; %re-implement boundary conditions
    temp(:,ysize) = 0;
    temp(1,:) = v0;
    temp(xsize,:) = temp(xsize-1,:);
end

surf(x,y,v,'EdgeColor','none'); %plotting as a surface w/ interpolated
    mapping
shading interp
xlabel('X axis')
ylabel('Y axis')
zlabel('Potential, V')
title('Relaxation Method Solution, Two Dimensional Lagrangian');
txt = [' V_0 = ' num2str(v0)];

text(xmax/1.5,ymax/1.5,v0,txt);
```



*Published with MATLAB® R2018a*