DEMO_0001_TPMS_Linear_Gradient

Table of Contents

This is a demo for:

- Building geometry for linear volume fraction or cell size gradiet TPMS lattices

Name

Author: *Mahtab Vafaee*, mahtab.vafaee@gmail.com

```
Change log:
2023/11/15 MV Created
2024/01/29 MV Sorted for publishing
-------------------------------------------------------------------------
```

```
clear; close all; clc;
```

Plot settings

```
cMap=parula(250);
faceAlpha1=1;
faceAlpha2=0.5;
edgeColor1='none';
edgeColor2='none';
fontSize=15;
```

Control parameters

Creating control parameters depending on the type of demo the user wants to run

```
gradType= 'levelSet'; % choose between cellSize or levelSet
levelset= 1; %Isosurface level, corresponding to volume fraction

switch gradType

    case 'levelSet' % PAPER, figure 1
```

```
        inputStruct.L=[3 1 1]; % characteristic length
        inputStruct.Ns=100; % number of sampling points, resolution
        inputStruct.surfaceCase='g'; %Surface type
        inputStruct.numPeriods=[9 3 3]; %Number of periods in each direction
 [6 2 2]

    case 'cellSize' % Coarse version
        inputStruct.L=[3 1 1]; % characteristic length
        inputStruct.Ns=100; % number of sampling points, resolution
        inputStruct.surfaceCase='g'; %Surface type
        inputStruct.numPeriods=[6 2 2]; %Number of periods in each direction
 [6 2 2]
end
```

Create triply periodic minimal surface

```
%Get parameters from input structure
L = inputStruct.L; % characteristic length
Ns = inputStruct.Ns; % number of sampling points
k = inputStruct.numPeriods;

%Create coordinates
xMin=0; xMax= 2*pi*k(1,1);
yMin=0; yMax= 2*pi*k(1,2);
zMin=0; zMax= 2*pi*k(1,3);

xRange=linspace(xMin,xMax,Ns);
yRange=linspace(yMin,yMax,Ns);
zRange=linspace(zMin,zMax,Ns);
[X,Y,Z]=meshgrid(xRange,yRange,zRange);

switch gradType

    case 'levelSet' % volume fraction gradient, Figure 1 (a,b)

        %Calculate 3D image data
        S=(sin(X).*cos(Y))+(sin(Y).*cos(Z))+(cos(X).*sin(Z));
        S=reshape(S,size(X));

        % levelset gradient
        GF=X; % Use x-dir for now
        GF=GF-min(GF(:)); % 0-...
        GF=GF./max(GF(:)); % 0-1

        GF=GF*((1/0.3)-(1/1.2)); % 0-2.5
        GF=GF + (1/1.2); % 0.8333-3.3333

        S=S.*GF;

    case 'cellSize' % cell size gradient, Figure 1 (d,e)

        % Calculate gradient frequency
        m=3;
```

```
        K1= (m-1)/(xMax-xMin);
        C1= (xMin*K1)+1;
        C0= 0.5*K1*(xMin)^2;

        a = K1/2*X+C1+C0/X;
        b = K1*X+C1;
        c = K1*X+C1;

        %Calculate 3D image data
        S=(sin(a.*(X-1/4*pi)).*cos(b.*(Y-1/4*pi)))+(sin(b.*(Y-1/4*pi)).*...
            cos(c.*(Z-1/4*pi)))+(cos(a.*(X-1/4*pi)).*sin(c.*(Z-1/4*pi)));
        S=reshape(S,size(X));

end
```

Scaling coordinates

```
switch length(L)
    case 1
        X=((X./abs(xMax-xMin)).*L);
        Y=((Y./abs(yMax-yMin)).*L);
        Z=((Z./abs(zMax-zMin)).*L);

    case 3
        X=((X./max(X(:))).*L(1,1));
        Y=((Y./max(Y(:))).*L(1,2));
        Z=((Z./max(Z(:))).*L(1,3));
end
```

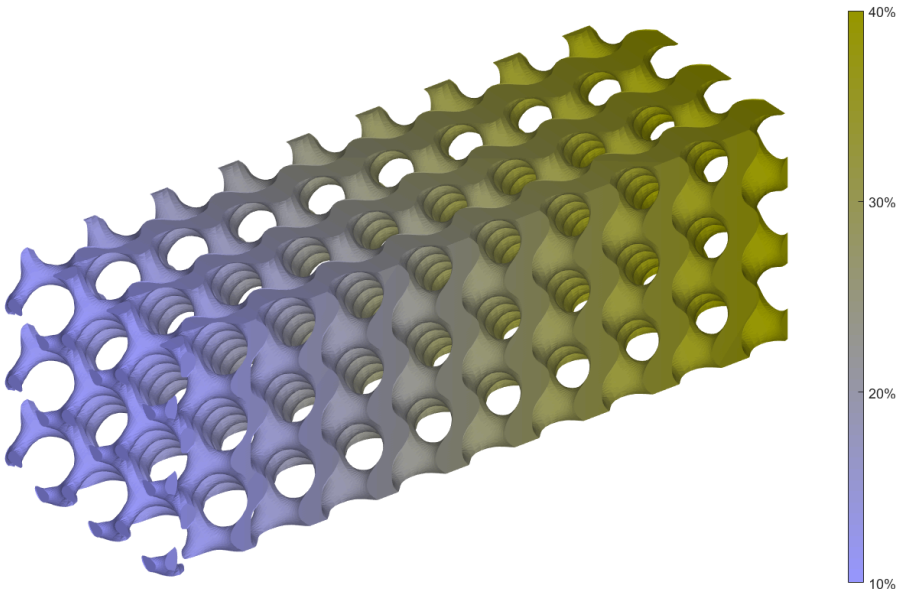
Isosurface

```
[F,V] = isosurface(X,Y,Z,S,levelset);
C=zeros(size(F,1),1);

%Capping ends
[fc,vc]=isocaps(X,Y,Z,S,levelset, 'above');
nc=patchNormal(fc,vc);
cc=zeros(size(fc,1),1);
cc(nc(:,1)<-0.5)=1;
cc(nc(:,1)>0.5)=2;
cc(nc(:,2)<-0.5)=3;
cc(nc(:,2)>0.5)=4;
cc(nc(:,3)<-0.5)=5;
cc(nc(:,3)>0.5)=6;

%Join sets
[f,v,c]=joinElementSets({F,fc},{V,vc},{C,cc});
[f,v]=mergeVertices(f,v); %Merge nodes

%Check for unique faces
[~,indUni,~]=unique(sort(f,2),'rows');
f=f(indUni,:); %Keep unique faces
```

```
c=c(indUni);

%Remove collapsed faces
[f,logicKeep]=patchRemoveCollapsed(f);
c=c(logicKeep);

%Remove unused points
[f,v]=patchCleanUnused(f,v);

f=fliplr(f); %Invert faces
```

Visualize surface

```
cFigure; hold on;

switch gradType

    case 'levelSet'
        [Vm] = patchCentre(f,v); % gradiant color
        gpatch(f,v,Vm(:,1),'none', 1);

        map=[0.6*ones(256,2), linspace(1, 0, 256)'];
        cmap = colormap(map) ; %Create Colormap
        cbh = colorbar;  %Create Colorbar
        cbh.Ticks = linspace(0, 3, 4); %Create 4 ticks 10%-40%
        cbh.TickLabelInterpreter = 'tex';
        cbh.TickLabels = {'10%','20%','30%','40%'} ;    %Replace the labels
        % of these 8 ticks with the numbers 1 to 8

    case 'cellSize'
        gpatch(f,v,[0.75, 0.75, 0],'none', 1);

end

axis off;
axisGeom; camlight headlight;
gdrawnow;
```



Published with MATLAB® R2021b