Solenoid Spiral

Table of Contents

X = solenoid_spiral(N,r0,h,phir,phi0,RES,x0,y0,z0,phix,phiy,phiz,view)

Author: JCCopyrights Summer 2019 This function generates a solenoid circular spiral geometry to be used as a coil. The coil will be generated with center in (0,0,0) in XY plane. It can be moved using the x0,..,phix... parameters The turns are flat until phir angle, then until the end of the turn it starts linearly increasing in height.

Parameters

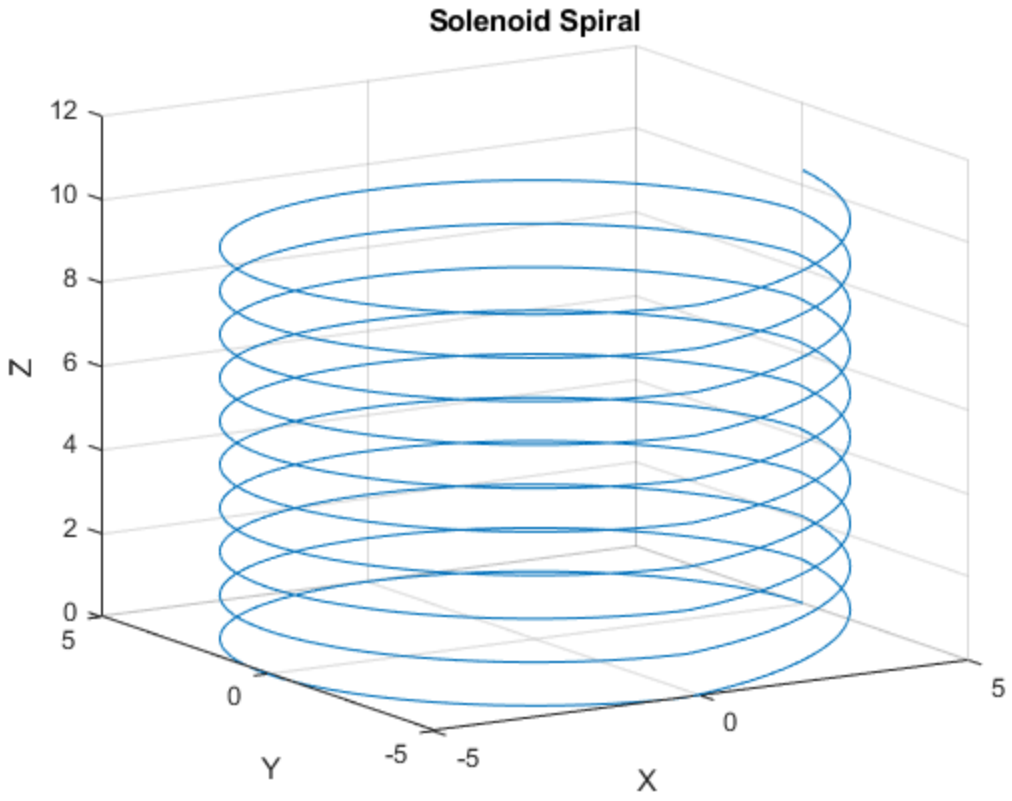
- @param **N** Number of Turns
- @param **r0** External radius of the coil
- @param **h** Height between turns
- @param **phir** Angle at which the turn ends
- @param **phi0** Angle at which the turns start
- @param **RES** Number of nodes of the Geometry (Discretization)
- @param **x0** Center position X
- @param **y0** Center position Y
- @param **z0** Center position Z
- @param **phix** Turn respect X axis
- @param **phiy** Turn respect Y axis
- @param **phiz** Turn respect Z axis
- @param **view** Optional parameter, if true generates figure with geometry
- @retval **X** Geometry nodes

Code

```
function X =
 solenoid_spiral(N,r0,h,phir,phi0,RES,x0,y0,z0,phix,phiy,phiz,view)
 % The next turns begins at phir, and continous increasing until 2pi
+phi0
 Rx=[1,0,0;0,cos(phix),-sin(phix);0,sin(phix),cos(phix)];
 Ry=[cos(phiy),0,sin(phiy);0,1,0;-sin(phiy),0,cos(phiy)];
 Rz=[cos(phiz),-sin(phiz),0;sin(phiz),cos(phiz),0;0,0,1];
 omega =.010;
 t = linspace(0,2*pi*N/omega,RES);
```

```
 X=ones(3,size(t,2));
 phi = omega*t+phi0;
 r = r0;
 X(1,:) = r .* cos(phi);
 X(2,:) = r .* sin(phi);
 X(3,1)=0;
 counter=phir;
 counter2=(2*pi+phi0);
 phi_RES=omega*2*pi*N/(omega*(RES-1));
 for i=2:size(t,2)
  if phi(i)>=counter
   %Starts Increasing for next Turn.
   X(3,i)=X(3,i-1)+h/((2*pi+phi0-phir)/phi_RES);
   if phi(i)>=counter2
    %Starts new Turn.
    counter=counter+2*pi;
    counter2=counter2+2*pi;
   end
  else
   X(3,i)=X(3,i-1);
  end
 end
 %The max height of the Helix is N*h
 for i=1:size(t,2)
  X(:,i)=transpose(Rx*[X(1,i);X(2,i);X(3,i)]);
  X(:,i)=transpose(Ry*[X(1,i);X(2,i);X(3,i)]);
  X(:,i)=transpose(Rz*[X(1,i);X(2,i);X(3,i)]);
  X(:,i)=X(:,i)+[x0;y0;z0];
 end
 if nargin>12
  if view
   plot3(X(1,:),X(2,:),X(3,:))
   grid on
   xlabel('X')
   ylabel('Y')
   zlabel('Z')
   title('Solenoid Spiral');
  end
 end
end
```

Geometry



Published with MATLAB® R2019a