In [1]:
```
%reload_ext autoreload
%autoreload 2
from openscad3 import *
set_printoptions(suppress=True)
```

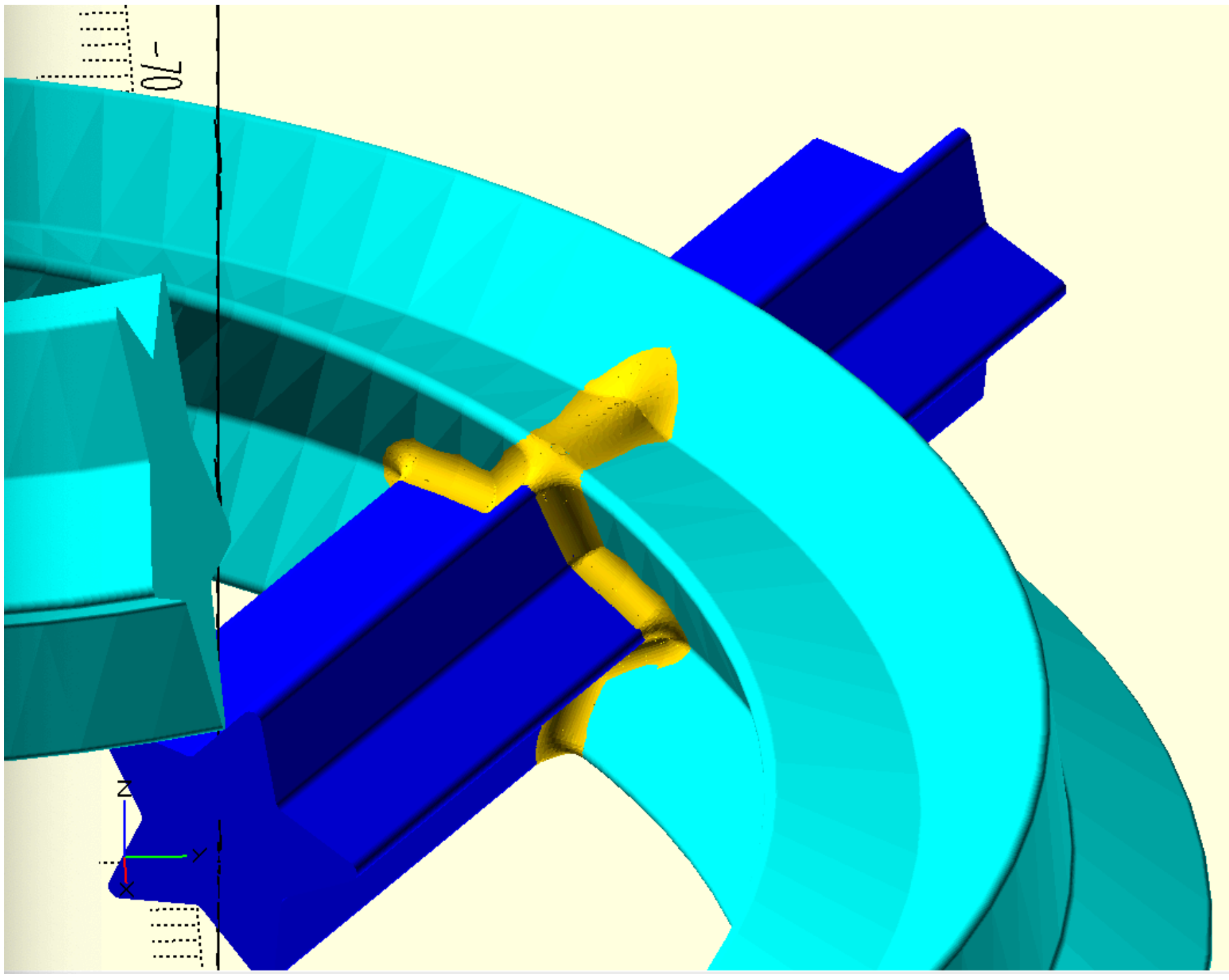
Finding intersection points

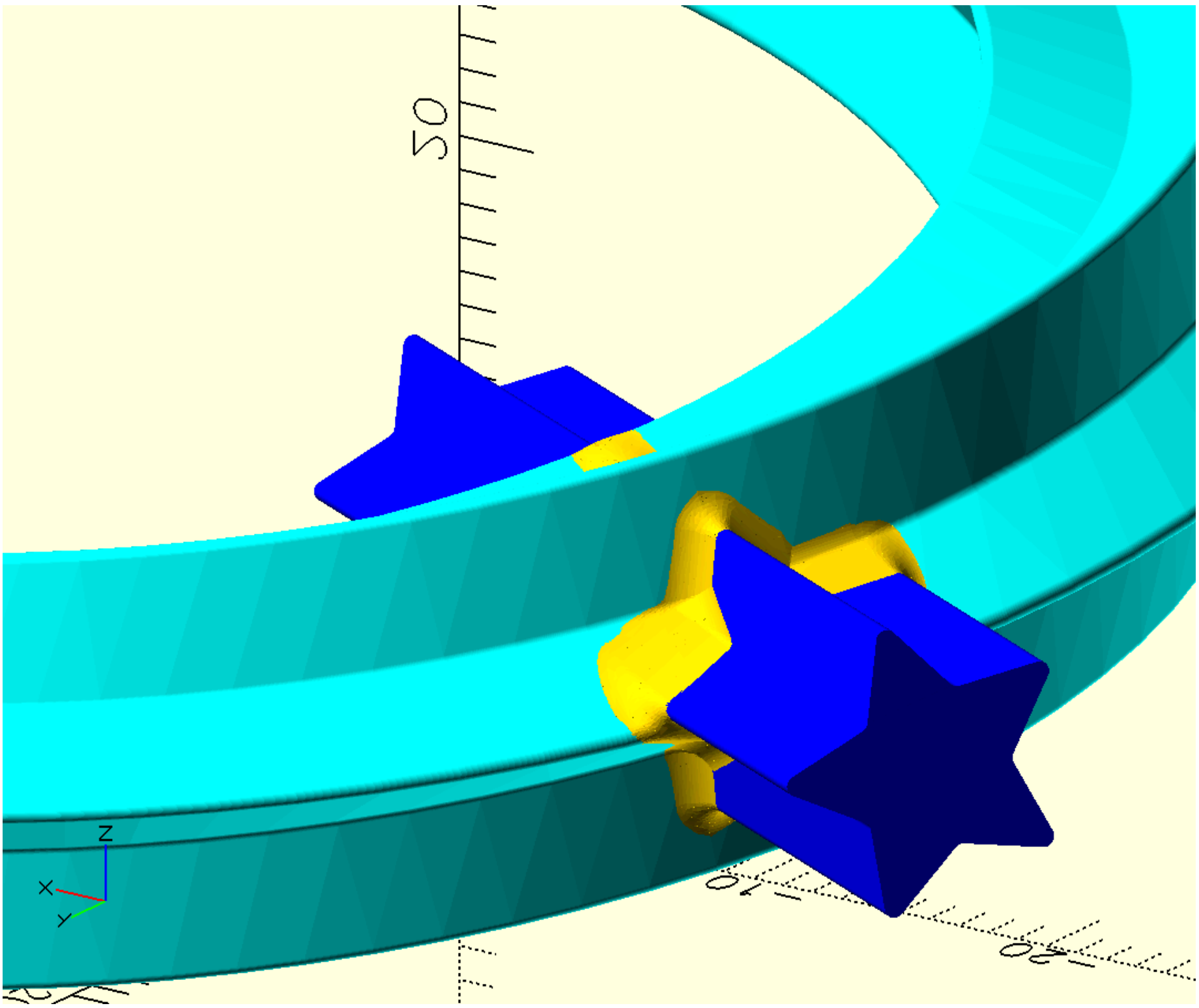
In [2]:
```
t0=time.time()
sec1=circle(10,s=6)
pent1=circle(7,s=6)
pent2=c3t2(rot(f'z{360/5/2}',circle(3.5,s=6)))
sec2=concatenate(cpo([pent1]+[pent2])).tolist()
sec2=corner_radius(array(c2t3(sec2))+[0,0,.3],20)
sec3=concatenate(cpo([pent1]+[pent2])).tolist()
sec3=offset(sec3,-1)
sec3=corner_radius(array(c2t3(sec3))+[0,0,.3],20)
path1=helix(20,30,1,5)
path2=[[0,0,10],[-30,20,13]]
sol=path_extrude_open(sec2,path1)
sol1=path_extrude_open(sec3,path2)
sol2=sol[20:40]

a=cr2dt([[1,0],[-1,0,1],[0,1]],30)
b=[ path_extrude_open(m_points1(offset(sec3,x),20,.1),path2) for (x,y) in a]
c=[ path_extrude_open(m_points1(offset(sec2,y),20,.1),path1[25:35],1) for (x,y) in a]

ipx=[ip_unordered(c[i],b[i]) for i in range(len(a))]
n1=[i_p_n(ipx[i],c[i])*-1 for i in range(len(a))]
ipx=concatenate(ipx)
n1=concatenate(n1)

t1=time.time()
t1-t0
```

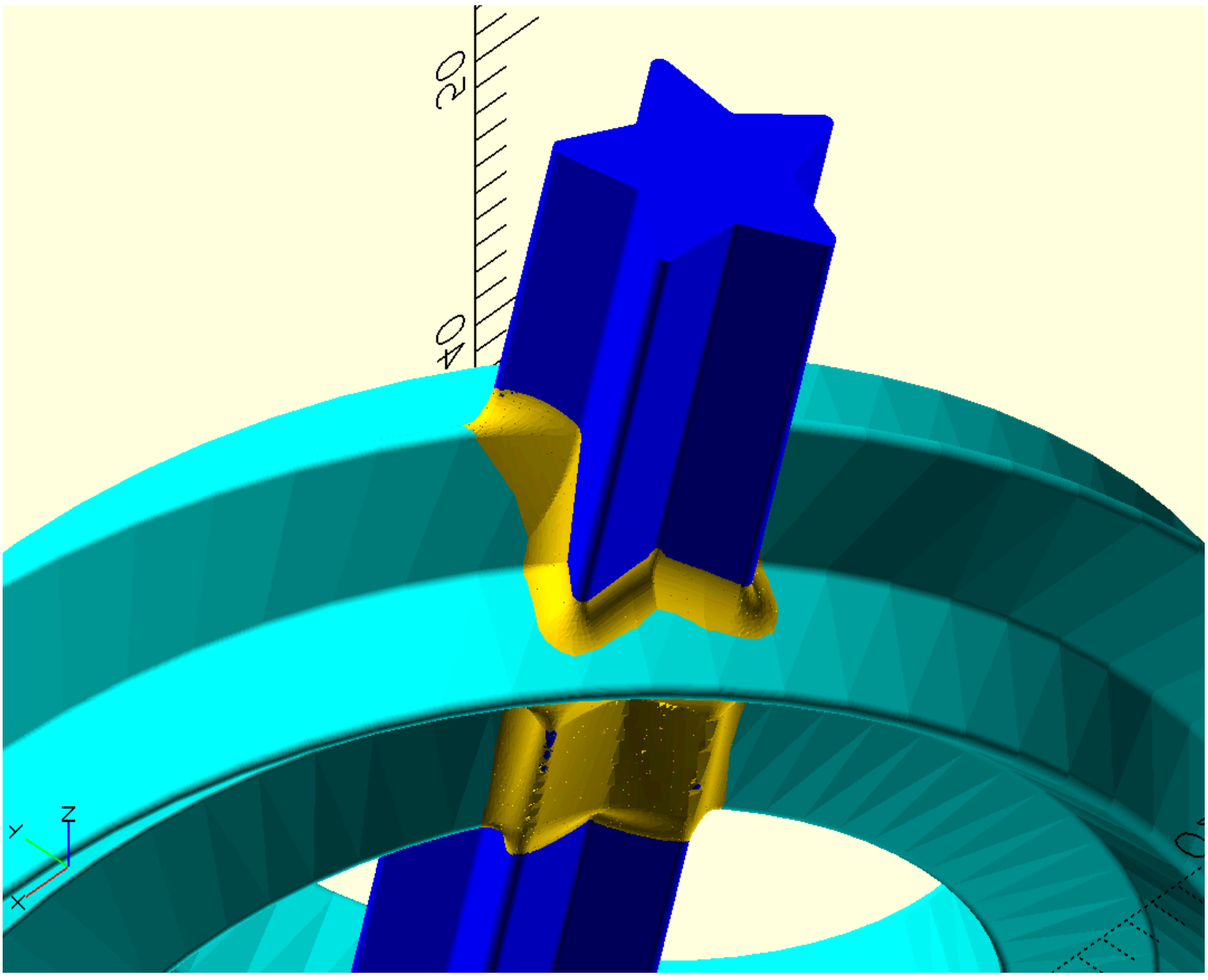
Out[2]:
```
172.5001871585846
```

ball pivoting mesh generation

In [4]:
```
tri=bp_mesh_from_points(ipx,n1,f=5)
cp=ipx.mean(0)
tri_1=[l_([cp,cp,cp])]*len(tri)
sol2=l_(a_([tri,tri_1]).transpose(1,0,2,3))
fo(f'''
for(p={sol2})swp(p);
color("cyan"){swp(sol)}
color("blue"){swp(sol1)}
''')
```

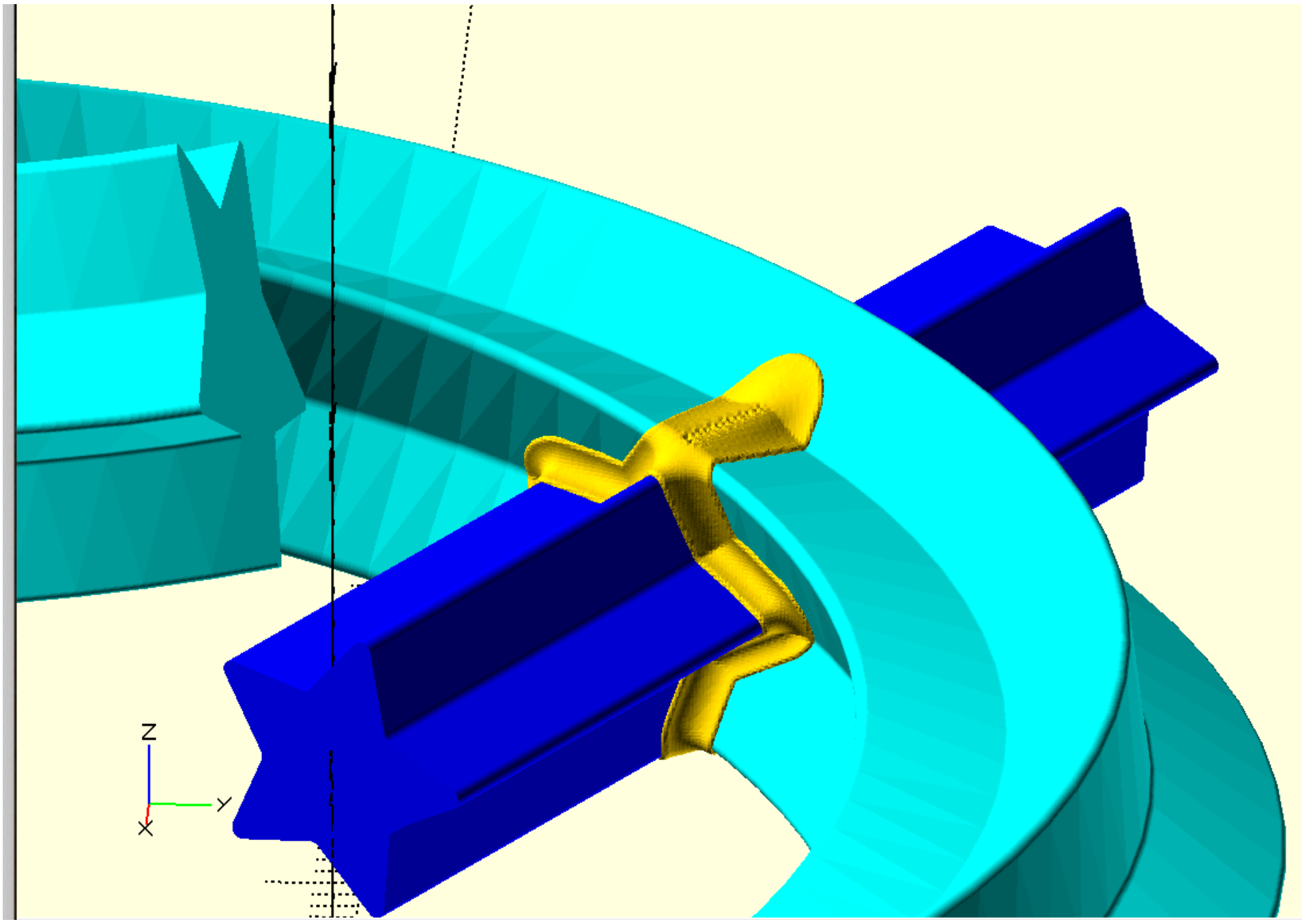


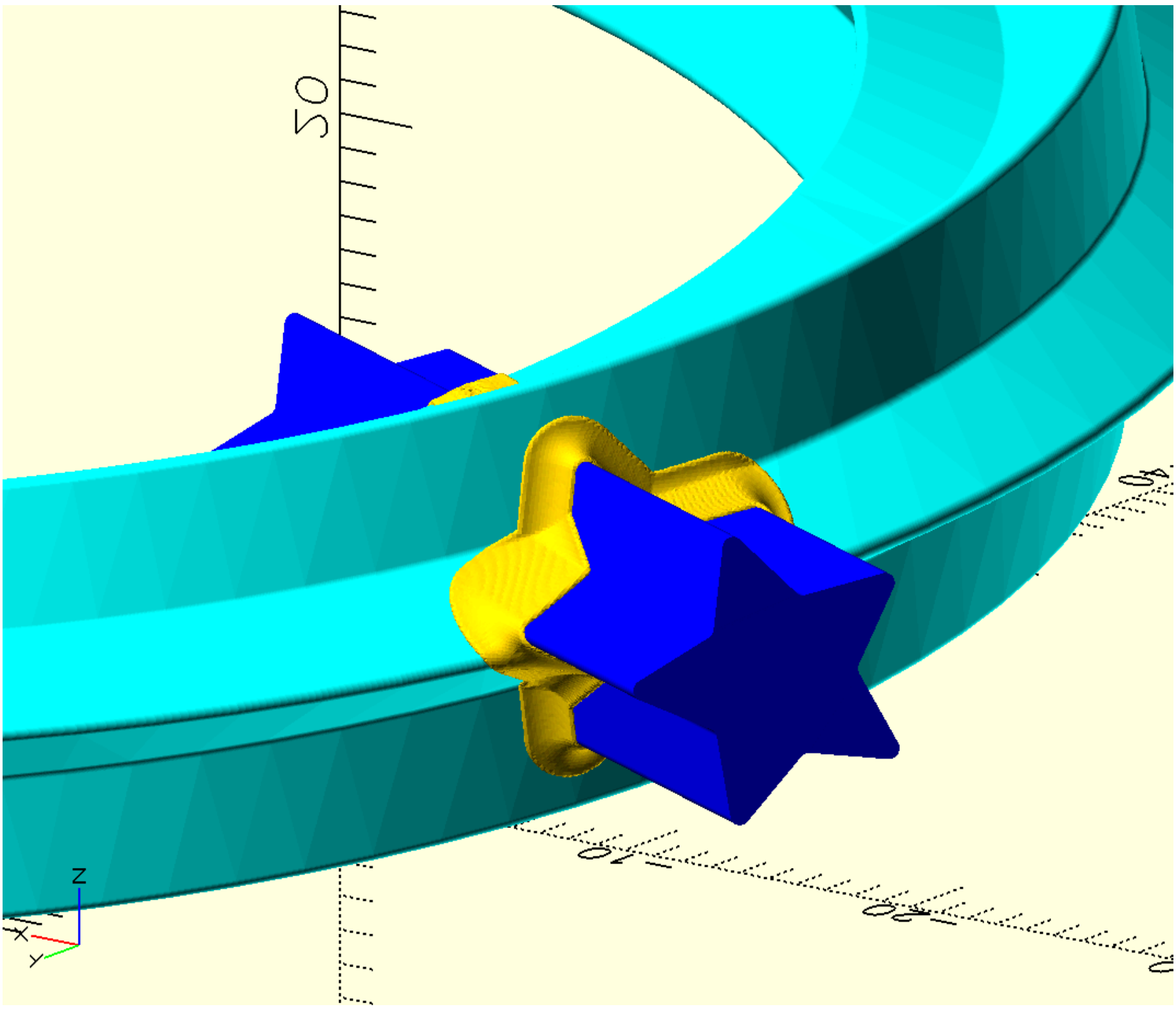

50
Z
X
Y

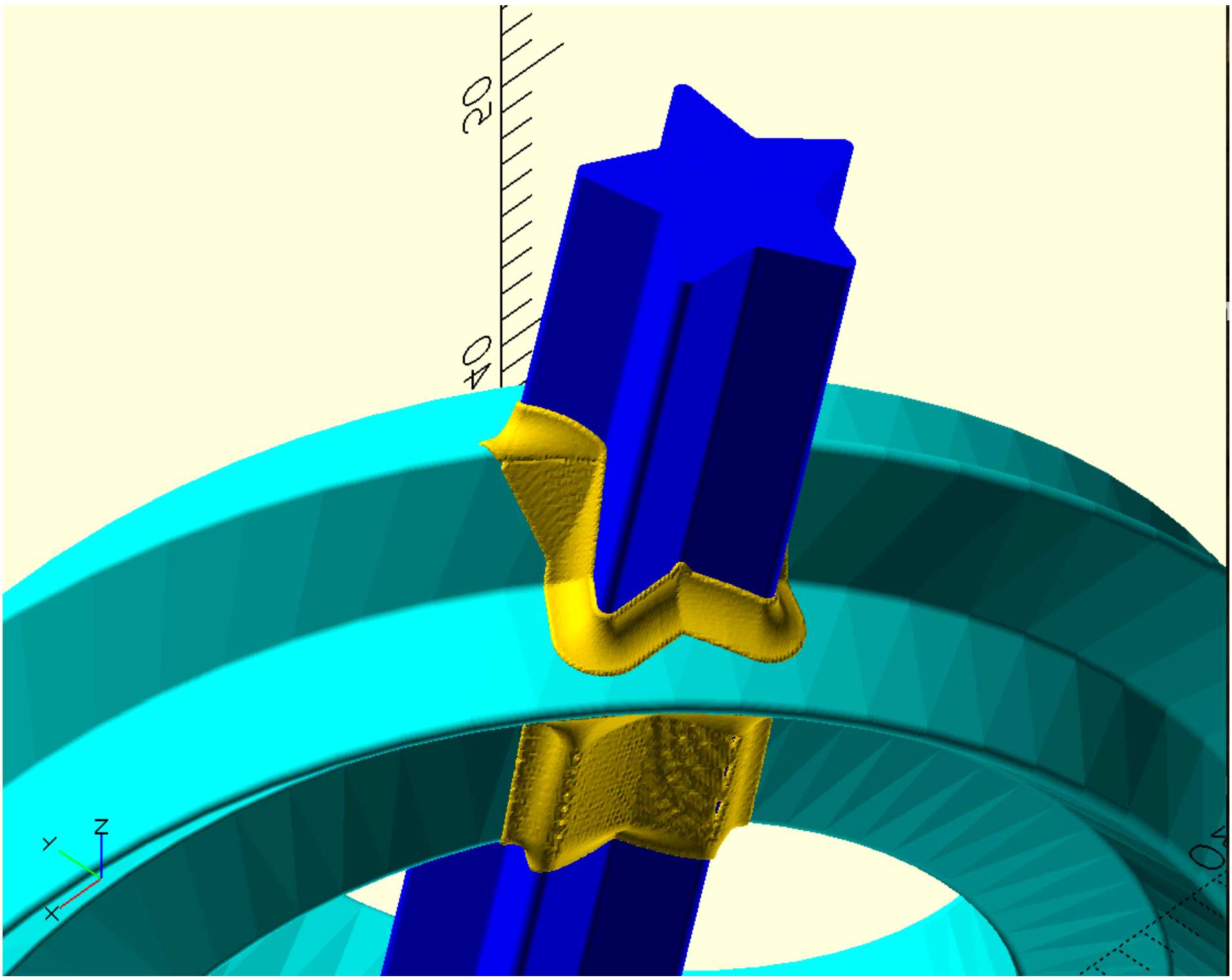

20
40
Z
Y
X

Marching cubes method

In [5]:
```
fo(f'''
{points_to_meshes(l_(ipx))}
color("cyan"){swp(sol)}
color("blue"){swp(sol1)}
''')
```




50
Z
X
Y


20
40
Z
Y
X