计算代码



for i in range(v3.shape[0]):

theta\_13\_t=np.arccos(np.dot(v1[i],v3[i])/np.sqrt((v1[i]\*\*2).sum()+(v3[i]\*\*2).sum()))

theta\_13.append(theta\_13\_t)

v1\_temp=-v1[i]#v1反方向向量

v1\_temp=v1\_temp/np.sqrt((v1\_temp\*\*2).sum()) #v1\_temp是v1反向量的单位化

v2\_temp = norm\_v(v2[i])#v2单位化

v3\_temp=v3[i]/np.sqrt((v3[i]\*\*2).sum())#v3单位化

if (v1[i][0]>=0 and v1[i][1]>=0):

#if(v3[i][0]<=0 and v3[i][1]>=0):# =是否能取到，何时取到，先凭感觉的，都取并没有影响

if -1<=v3\_temp[0]<=v2\_temp[0] and v3\_temp[1]>=0:

theta\_w2 = theta\_c[i]-theta\_13

if -1<=v3\_temp[0]<=v1\_temp[0] and v3\_temp[1]<=0:

theta\_w2 = theta\_c[i]-theta\_13

if v1\_temp[0]<=v3\_temp[0]<=1 and v3\_temp[1]<=0:

theta\_w2 = theta\_c[i]+theta\_13

if -v1\_temp[0]<=v3\_temp[0]<=1 and v3\_temp[1]>=0:

theta\_w2 = theta\_c[i]+theta\_13

if v2\_temp[0]<=v3\_temp[0]<=-v1\_temp[0] and v3\_temp[1]>=0:

theta\_w2 = theta\_c[i]-theta\_13

if v1[i][0]>=0 and v1[i][1]<=0:

if -1<=v3\_temp[0]<=-v1\_temp[0] and v3\_temp[1]<=0:

theta\_w2 = theta\_c[i]+theta\_13

if -1<=v3\_temp[0]<=v1\_temp[0] and v3\_temp[1]>=0:

theta\_w2 = theta\_c[i]+theta\_13

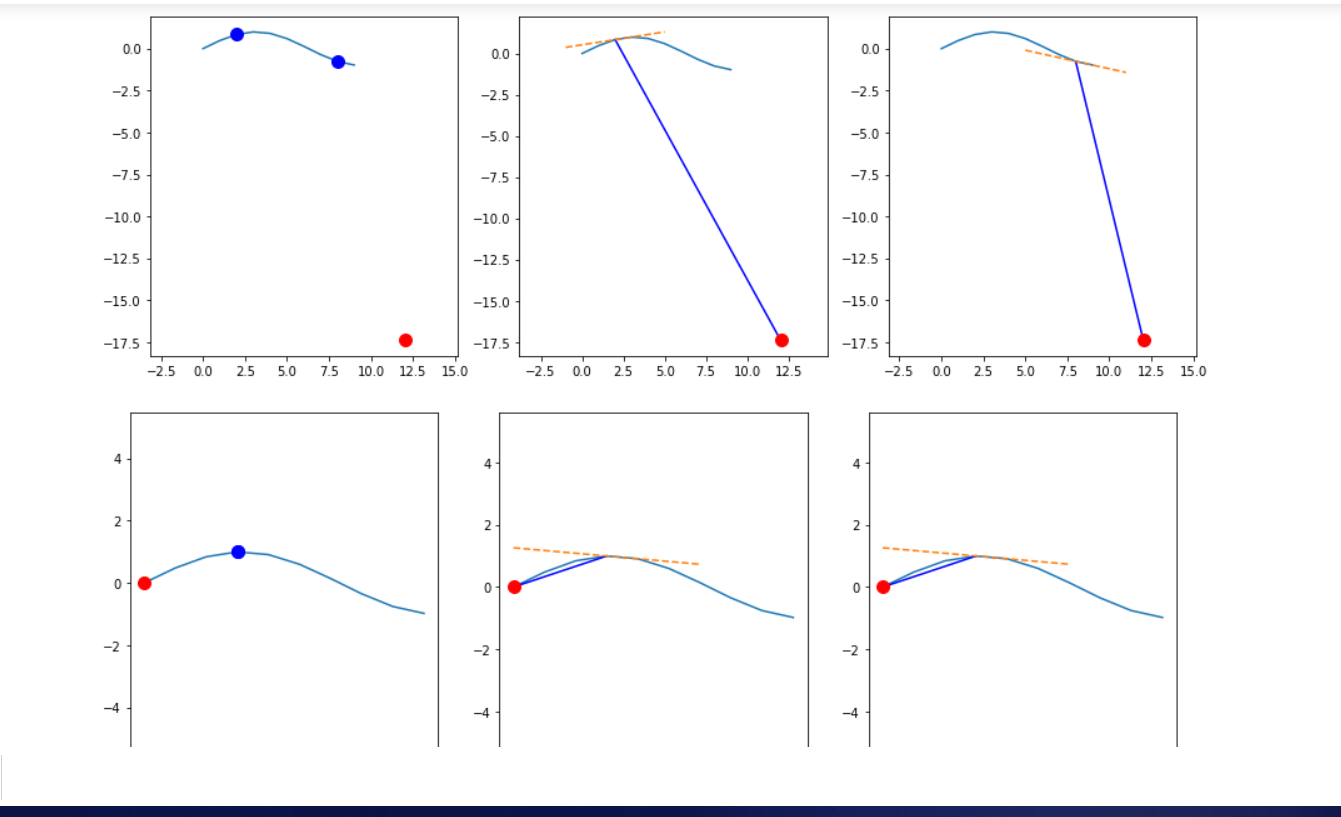
if v1\_temp[0]<=v3\_temp[0]<=1 and v3\_temp[1]>=0:

theta\_w2 = theta\_c[i]-theta\_13

if -v1\_temp[0]<=v3\_temp[0]<=1 and v3\_temp[1]<=0:

theta\_w2 = theta\_c[i]-theta\_13

绘图结果



条件：水流v3在第一象限且与x轴成45°。

错误情况

第一组图应该是正确的。

第二组图应该是错误的。

所以应当检查v3在第一象限而v1如上图橙色线所示情况的theta\_w计算代码。

其他思路：

1，可以将入射线一并画出来并采用text(x,y,angel)来标注角度信息。

2，在放入实际海岸线之前，可以一个情况一个情况来推测代码是否正确。