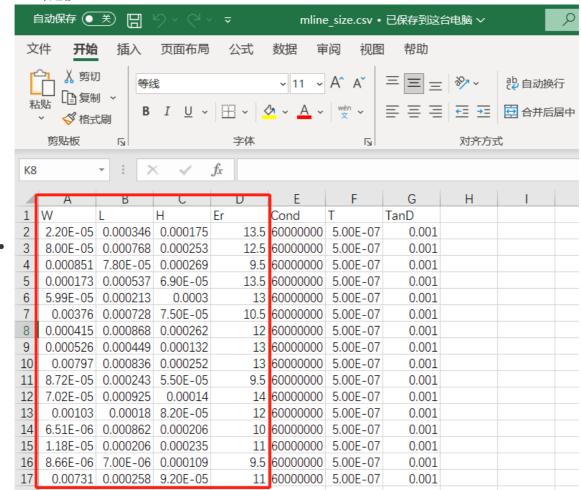
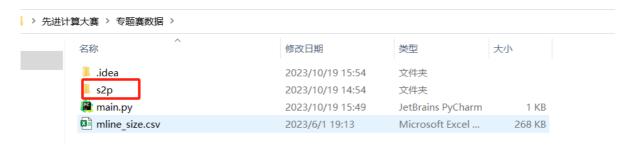
B2-AI微带线建模

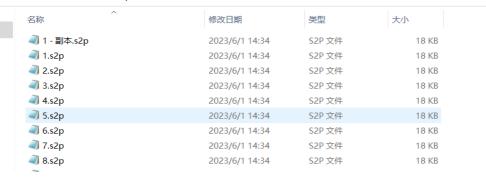
问题介绍:

- 输入W、L、H、Er四个参数,自动输出一个s2p文件,包含20行,一行对应一个频段。
- 一个频段要求给出四个参数, S11, S12, S21, S22。但由于对角元素相同,实际只要得到S11, S21即足够



s2p文件的介绍:





用txt方式打开s2p文件,可知:

<u>■</u> 2.s2p - 记事本							- 0	×
文件(E) 编辑(E) 格式(C	查看(Y) 帮助(H)							
# Hz S RI R 50.00								^
100000000.0	0.0032519491295	0.00148298111982	0.996738883033	-0.00468704228247	0.996738883033	-0.0046870422824	7 0.0032519491295	0
200000000.0	0.00329302889542	0.00296556360454	0.996679357437	-0.00937373304386	0.996679357437	-0.0093737330438	86 0.0032930288954	2
300000000.0	0.00334237111953	0.00444771482247	0.996601289928	-0.0140600648037	0.996601289928	-0.0140600648037	0.00334237111953	0
400000000.0	0.00340278107458	0.00592933321506	0.996501873312	-0.0187459509743	0.996501873312	-0.0187459509743	0.00340278107458	0
500000000.0	0.00347533498278	0.00741030162558	0.996380029045	-0.0234312841563	0.996380029045	-0.0234312841563	0.00347533498278	0
600000000.0	0.0035605776754	0.0088904962309	0.996235209611	-0.0281159468242	0.996235209611	-0.0281159468242	0.0035605776754	0.0
700000000.0	0.00365882694518	0.0103697897285	0.996067094201	-0.0327998154125	0.996067094201	-0.0327998154125	0.00365882694518	0.
800000000.0	0.00377028458582	0.0118480528064	0.995875477696	-0.037482762327	0.995875477696	-0.037482762327	0.00377028458582	0.0
900000000.0	0.00389508581433	0.0133251549296	0.995660221269	-0.0421646570898	0.995660221269	-0.0421646570898	0.00389508581433	0.1
1000000000.0	0.00403332431316	0.0148009648054	0.995421227352	-0.0468453670574	0.995421227352	-0.0468453670574	0.00403332431316	0
1100000000.0	0.00418506614605	0.0162753506788	0.995158425744	-0.0515247579049	0.995158425744	-0.0515247579049	0.00418506614604	0
1200000000.0	0.00435035804748	0.0177481805324	0.994871765328	-0.0562026939694	0.994871765328	-0.0562026939694	0.00435035804748	0
1300000000.0	0.0045292326369	0.0192193222269	0.994561208868	-0.0608790385052	0.994561208868	-0.0608790385052	0.0045292326369	0.0
1400000000.0	0.00472171184547	0.0206886436049	0.994226729597	-0.0655536538794	0.994226729597	-0.0655536538794	0.00472171184547	0
1500000000.0	0.00492780925123	0.0221560125695	0.993868308887	-0.0702264017269	0.993868308887	-0.0702264017269	0.00492780925123	0
1600000000.0	0.00514753171951	0.0236212971463	0.99348593462	-0.0748971430762	0.99348593462	-0.0748971430762	0.00514753171951	0.0
1700000000.0	0.00538088058572	0.0250843655332	0.993079600012	-0.0795657384529	0.993079600012	-0.0795657384529	0.00538088058572	0
1800000000.0	0.00562785252751	0.0265450861417	0.992649302746	-0.0842320479679	0.992649302746	-0.0842320479679	0.00562785252751	0
1900000000.0	0.00588844022079	0.0280033276319	0.992195044326	-0.0888959313927	0.992195044326	-0.0888959313927	0.00588844022079	0
2000000000.0	0.00616263284188	0.0294589589428	0.991716829579	-0.0935572482245	0.991716829579	-0.0935572482245	0.00616263284188	0 ~
<								>
					第1名	T, 第1列 100% Wi	ndows (CRLF) UTF-8	

含义介绍:

什么是s2p文件?我用txt方式打开了s2p文件,请你帮我分析一下是啥意思# Hz S RI R 50.00 200000000.0 0.00329302889542 0.00296556360454 0.996679357437 -0.00937373304386 0.996679357437 -0.00937373304386 300000000.0 -0.0140600648037 400000000.0 0.00340278107458 0.00592933321506 0.996501873312 -0.0187459509743 0.996501873312 -0.0187459509743 5000000000 -0.0234312841563



2023/10/19 10:40:42

S2P文件是一种用于描述射频器件 (例如电缆、滤波器、放大器等) 的参数文件。它包含了频率点和相关参数,通常用于射频系统设计和仿真。

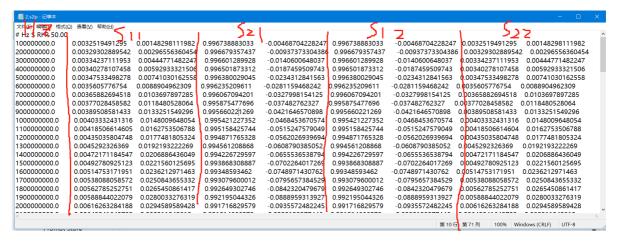
在给出的示例中,每行表示一个频率点及其参数。文件的开头"# Hz S RI R 50.00"表示该文件使用的频率单位是Hz,参数类型为S参数,数据类型为实数(RI),参考阻抗为50.00欧姆。

接下来的每行都表示一个频率点和对应的参数值。以第一行为例解释,频率为100MHz(即100000000.0Hz),S参数有8个值,分别是0.0032519491295、0.00148298111982、0.996738883033、-0.00468704228247、等等。

具体来说,这些值通常表示复数形式的参数,对应于网络理论中的散射参数(S-parameters)。其中,前两个值(0.0032519491295和0.00148298111982)分别表示S11参数的实部和虚部,接下来两个值(0.996738883033和-0.00468704228247)表示S21参数的实部和虚部,以此类推。

注意实数和虚数的命名:

• 例如,S11部分,第一列是实数,为S11_realPart,第二列为S11_imagePart



区分单频点和多频点数据

假设有一个射频电路, 您需要对其进行S参数测量, 并得到以下数据:

• 单频点: 在频率f=1 GHz上测量的S参数为:

```
[[0.3 + 0.4j, 0.2 - 0.1j],
[0.2 - 0.1j, 0.3 + 0.2j]]
```

这是一个2x2的矩阵,表示该射频电路在频率为1 GHz时输入和输出端口之间的传输函数。

多频点:在频率范围[1 GHz, 2 GHz]内均匀采样n个频率点的S参数数据。例如,当n=3时,您得到以下S参数数据:

```
[
[[0.3 + 0.4j, 0.2 - 0.1j], # 在f=1 GHz处的S参数矩阵
[0.2 - 0.1j, 0.3 + 0.2j]],

[[0.2 + 0.3j, 0.1 - 0.2j], # 在f=1.5 GHz处的S参数矩阵
[0.1 - 0.2j, 0.4 + 0.3j]],

[[0.1 + 0.2j, 0.3 - 0.1j], # 在f=2 GHz处的S参数矩阵
[0.3 - 0.1j, 0.2 + 0.1j]]
]
```

这是一个3维数组,其中每个元素都是一个2x2的矩阵,分别对应于不同频率下的S参数。例如,第一个元素表示在1 GHz下的S参数矩阵,第二个元素表示在1.5 GHz下的S参数矩阵,以此类推。

当您处理单频点S参数时,可以直接使用矩阵操作来分析电路的特性,例如计算反射损耗或传输损耗等。 而在处理多频点S参数时,则需要针对每个频率点分别进行分析,例如绘制频率响应曲线、计算带宽等。 毕竟多频点S参数提供了更丰富的频率信息,可以更好地描述电路的行为。

Python读取s2p文件并进行打印输出:

```
import skrf

# 读取S2P文件

# 此处替换为绝对路径

filename = 'D:\\桌面\\先进计算大赛\\专题赛数据\\s2p\\1.s2p'
network = skrf.Network(filename)

# 获取S参数的数据
s_params = network.s

print(s_params)
```

多频点数据输出展示:

```
D:\Anaconda\python.exe D:/桌面/先进计算大赛/专题赛数据/main.py
[[[0.00531001+0.00117779j 0.99468719-0.00235004j]
        [0.99468719-0.00235004j 0.00531001+0.00117779j]]

[[0.00535293+0.00235531j 0.99464111-0.00469982j]
        [0.99464111-0.00469982j 0.00535293+0.00235531j]]

[[0.00539204+0.00353263j 0.99459745-0.00704941j]
        [0.99459745-0.00704941j 0.00539204+0.00353263j]]

[[0.00543195+0.00470975j 0.99455161-0.00939883j]
        [0.99455161-0.00939883j 0.00543195+0.00470975j]]
```

批量读取s2p文件并转化为csv文件:

```
import os
import numpy as np
import pandas as pd
import skrf
# 输入文件夹路径和输出文件夹路径
input_folder = 'D:\\桌面\\先进计算大赛\\专题赛数据\\s2p'
output_folder = 'D:\\桌面\\先进计算大赛\\专题赛数据\\s2p_To_csv'
# 获取所有s2p文件的路径
file_list = os.listdir(input_folder)
s2p_files = [os.path.join(input_folder, file) for file in file_list if
file.endswith('.s2p')]
# 逐个处理s2p文件
for i, s2p_file in enumerate(s2p_files):
   # 读取S参数数据
   network = skrf.Network(s2p_file)
   s_params = network.s
   freqs = network.f
   # 将S参数数据转换为DataFrame格式
```

```
columns = ["freq (GHz)", "S11_real", "S11_image", "S21_real", "S21_image",
"S12_real", "S12_image", "S22_real",
              "S22_image"]
   data = np.column_stack((freqs, np.abs(s_params[:, 0, 0]),
np.angle(s_params[:, 0, 0]),
                           np.abs(s_params[:, 1, 0]), np.angle(s_params[:, 1,
0]),
                           np.abs(s_params[:, 0, 1]), np.angle(s_params[:, 0,
1]),
                           np.abs(s_params[:, 1, 1]), np.angle(s_params[:, 1,
1])))
   df = pd.DataFrame(data=data, columns=columns)
   # 构造输出文件路径,保持与输入文件相同的文件名
   base_name = os.path.basename(s2p_file)
   output_file = os.path.join(output_folder, base_name.replace('.s2p', '.csv'))
   # 保存为CSV文件
   df.to_csv(output_file, index=False)
   # 打印结果
   print(f"转换成功: {s2p_file} -> {output_file}")
```