

Cavities Assignment Code

March 15, 2022

```
[1]: import numpy as np
from scipy.special import jn_zeros, jnp_zeros

d = 78.5e-3 #m
l = 20e-3 #m
c = 3e8 # m

def res_freq(m, n, p, mode="TM", Nzeros=5):

    jmn = jn_zeros(m,Nzeros)[n-1] if mode == "TM" else jnp_zeros(m,Nzeros)[n-1]

    return np.sqrt((c * jmn / np.pi)**2. + (c * p * d / (2 * l))**2) / d

[2]: # ten lowest eigenmodes, listing (m,n,p,mode)
modes = [(0,1,0,"TM"), (1,1,0,"TM"), (2,1,0,"TM"), (0,2,0,"TM"),(1,1,1,"TE"),
        ↵
        ↪(3,1,0,"TM"),(0,1,1,"TM"),(2,1,1,"TE"),(1,2,0,"TM"),(1,1,1,"TM"),(0,1,1,"TE")]

freqs = []

for mode in modes:
    freq = res_freq(*mode)
    freqs.append(freq)

    print("m,n,p: ({0}, {1}, {2}), mode:{3}, freq:{4:.4f}GHz".format(*mode,↵
    ↪freq*1e-9))
```

```
m,n,p: (0, 1, 0), mode:TM, freq:2.9254GHz
m,n,p: (1, 1, 0), mode:TM, freq:4.6612GHz
m,n,p: (2, 1, 0), mode:TM, freq:6.2473GHz
m,n,p: (0, 2, 0), mode:TM, freq:6.7150GHz
m,n,p: (1, 1, 1), mode:TE, freq:7.8273GHz
m,n,p: (3, 1, 0), mode:TM, freq:7.7613GHz
m,n,p: (0, 1, 1), mode:TM, freq:8.0503GHz
m,n,p: (2, 1, 1), mode:TE, freq:8.3698GHz
m,n,p: (1, 2, 0), mode:TM, freq:8.5343GHz
m,n,p: (1, 1, 1), mode:TM, freq:8.8304GHz
m,n,p: (0, 1, 1), mode:TE, freq:8.8304GHz
```