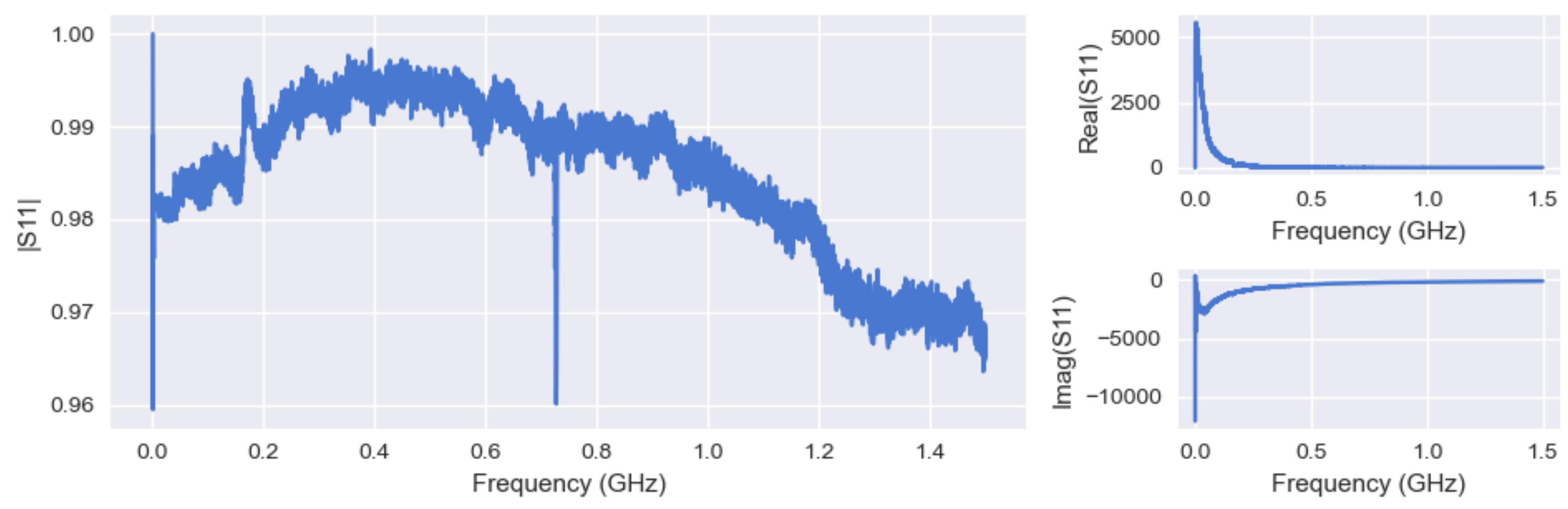
In [ ]:
```python
# Always run this block first whenever you change the bvd_library.py file, otherwise the change won't be updated to this notebook.
import importlib
import bvd_library
importlib.reload(bvd_library)
from bvd_library import *

# add %matplotlib widget to make plots interactive
%matplotlib widget
mpl.style.use('seaborn')
mpl.style.use('seaborn-muted')
```

First, we create a BVD_Model instance and load file. Then we plot |S11| and identify resonance.

In [ ]:
```python
bvd_model = BVD_Model()
bvd_model.load_file('B0.s2p')
bvd_model.plot_s11(figscale=0.8)
```

Figure



From the |S11| plot, we can identify the background frequency range with [fbs, fbe], and the resonance region with [frs, fre]. We first do background fitting to obtain Rs, Ls, Rp, Cp; then we do resonance fitting to obtain Rm, Cm, Lm. Procedure for fitting S11 with BVD model is:

from S11 data, we compute raw impedance using $Z_0 = Z_{char} \times \dfrac{1+S11}{1-S11}$, where characteristic impedance Z_{char} is taken to be 50 ohm.

We slice two flat regions out of raw impedance and concatenate them to obtain background impledance data Z_{bg}, which can be fitted using equation:

$$Z_{bg,fit} = R_s + jwL_s + \frac{1}{\frac{1}{R_p} + jwC_p}$$

We fit Z_{bg} data with $Z_{bg,fit}$ equation using bvd_model.fit_BVD_model_background method and obtain [Rs, Ls, Rp, Cp]; Using these values, we can de-background admittance data:

$$Y_a = \frac{1}{Z_0 - Rs - jwL_s} - (\frac{1}{Rp} + jwC_p)$$

The admittance data here can be modeled with the RLC circuit, which can be fitted with equation:

$$Y_{a,fit} = \frac{1}{R_m + jwL_m - j/wC_m}$$

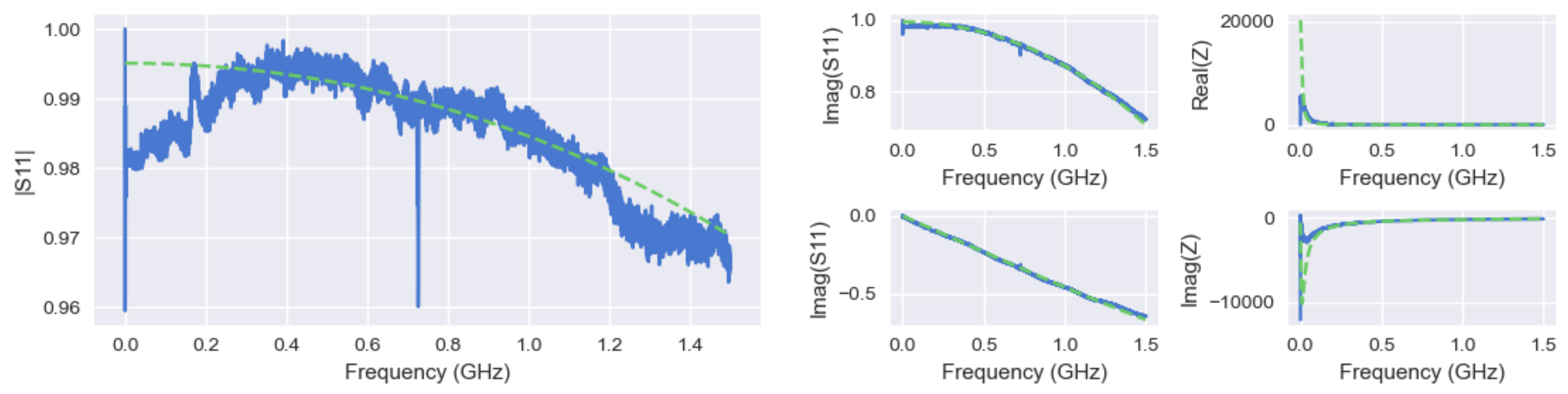
We fit Y_a data with $Y_{a,fit}$ using bvd_model.fit_BVD_model_resonances method, and obtain [Rm, Lm, Cm].

In [ ]:
```python
fbs, frs, fre, fbe = 0.3e9, 0.71e9, 0.74e9, 14e9
_, _, BG_fit_params = bvd_model.fit_BVD_model_background(start1=fbs, end1=frs, start2=fre, end2=fbe,
                                                          plot_fit=True, figscale=0.8
                                                         )
[Ls, Rs, Rp, Cp] = BG_fit_params
print('Ls =', Ls, 'nH;', 'Rs =', Rs, 'Ohm;', 'Rp =', Rp, 'Ohm;', 'Cp =', Cp, 'nF')
s11_fit, Z_total, popt, Y_list = bvd_model.fit_BVD_model_resonances(start1=fbs, end1=frs, start2=fre, end2=fbe,
                                                                      bg_params=BG_fit_params, rs_manual_fitting_params=None,
                                                                      plot_fit=True, figscale=0.8
                                                                     )
[Lm, Cm, Rm] = popt
print('Lm =', Lm, 'nH;', 'Cm =', Cm, 'nF;', 'Rm =', Rm, 'Ohm')
```

```
Ls = 1.83377519411899 nH; Rs = 4.542031472646649 Ohm; Rp = 20274.471182348385 Ohm; Cp = 0.0007394531193844545 nF
Lm = 301155.6550772744 nH; Cm = 1.58985277318971e-07 nF; Rm = 3403.2052449051102 Ohm
```

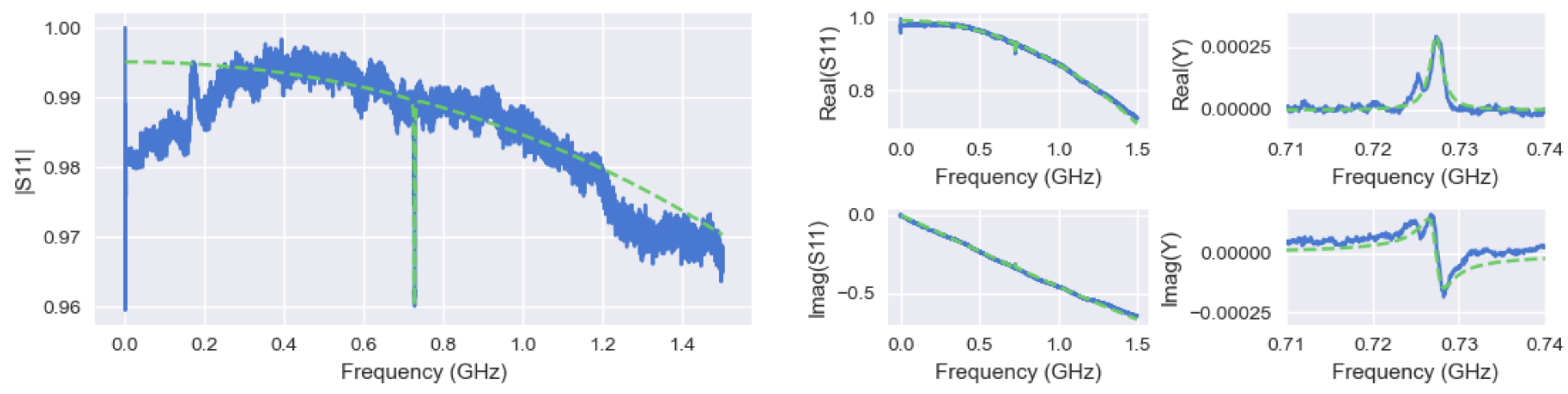
Figure

Background Fitting



Figure

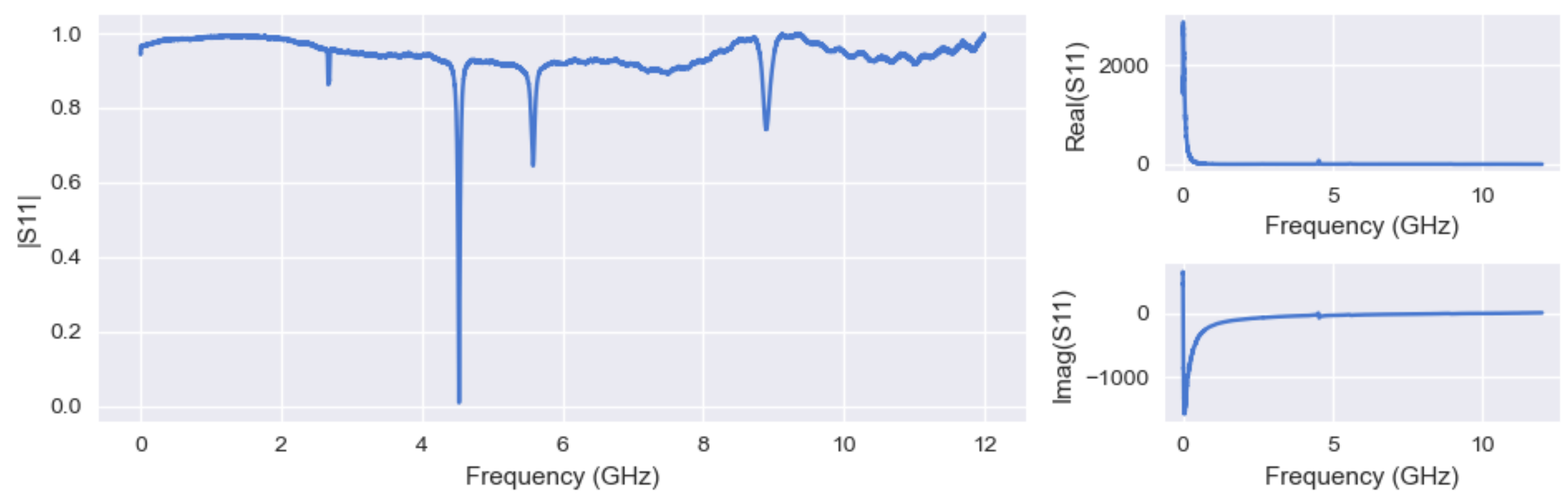
Resonance Fitting



We can also load .prn file. However, this will likely prompt you to install a 'matlabengin' libray, which can be done with "python pip install matlabengine" cammand.

In [ ]:
```python
bvd_model.load_file('4K-2-S11-pol.prn')
bvd_model.plot_s11(figscale=0.8)
```

Figure



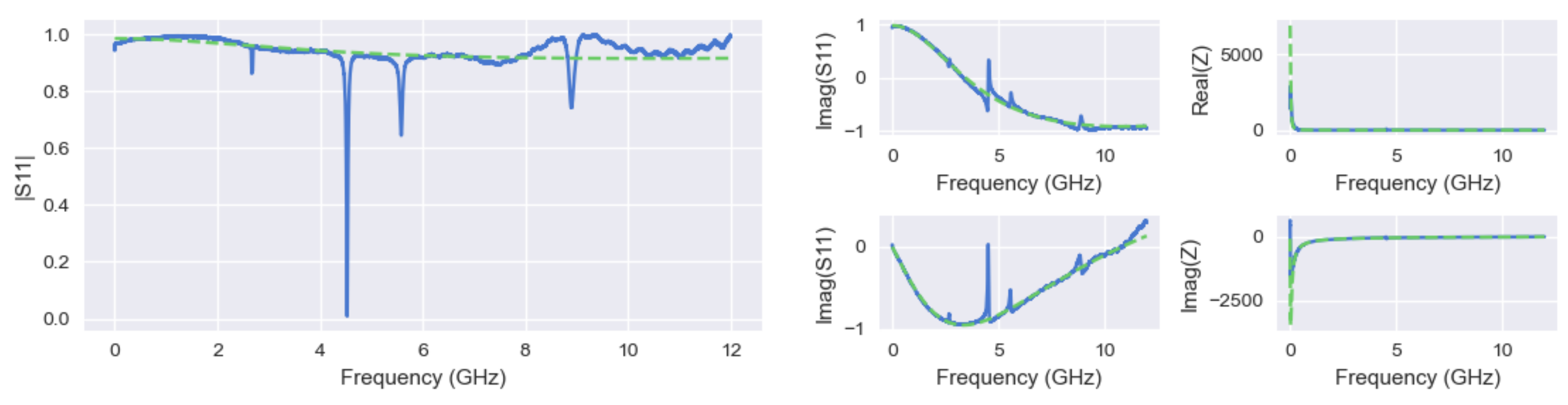
From the |S11| plot, we set [fbs, frs, fre, fbe] as before.

In [ ]:
```python
fbs, frs, fre, fbe = 2e9, 4.3e9, 4.7e9, 10e9
_, _, BG_fit_params = bvd_model.fit_BVD_model_background(start1=fbs, end1=frs, start2=fre, end2=fbe,
                                                          plot_fit=True, figscale=0.8
                                                         )
[Ls, Rs, Rp, Cp] = BG_fit_params
print('Ls =', Ls, 'nH;', 'Rs =', Rs, 'Ohm;', 'Rp =', Rp, 'Ohm;', 'Cp =', Cp, 'nF')
s11_fit, Z_total, popt, Y_list = bvd_model.fit_BVD_model_resonances(start1=fbs, end1=frs, start2=fre, end2=fbe,
                                                                      bg_params=BG_fit_params, rs_manual_fitting_params=None,
                                                                      plot_fit=True, figscale=0.8
                                                                     )
[Lm, Cm, Rm] = popt
print('Lm =', Lm, 'nH;', 'Cm =', Cm, 'nF;', 'Rm =', Rm, 'Ohm')
```

```
Ls = 0.2609268708011194 nH; Rs = 2.164788916496168 Ohm; Rp = 6893.990305717491 Ohm; Cp = 0.00082667040865783 nF
Lm = 145.48679520207793 nH; Cm = 8.535864579693588e-06 nF; Rm = 24.32589033819095 Ohm
```

Figure

Background Fitting



Figure

Resonance Fitting