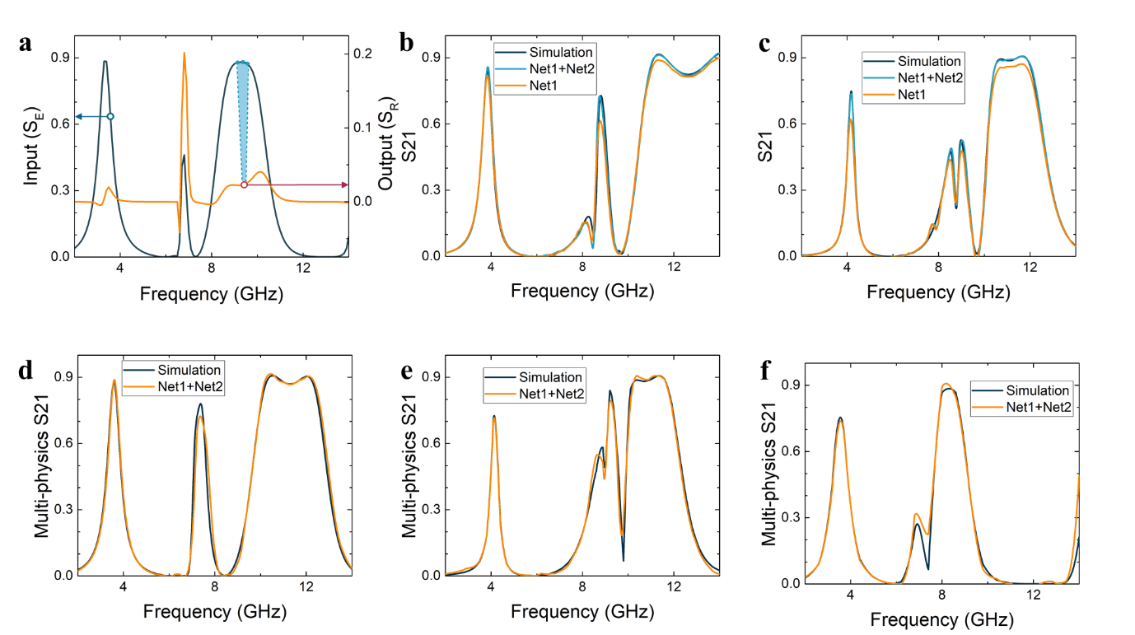
Multi-physical metasurface NN

**Configurations:**

**Suggested Environment**: Python 3.8.18, tensorflow 2.13.0, deepxde 1.10.2

**Dataset: All datasets for training and testing are contained in the subdirectory.**

**EM response by multi-fidelity DeepOnet:**

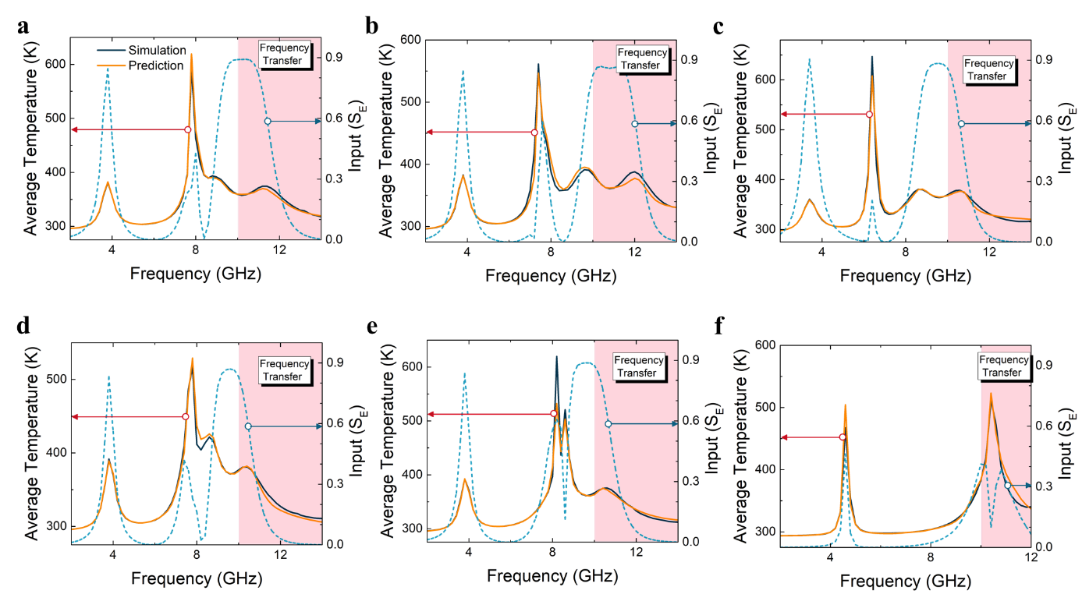


**There are 3 steps in total in this shared code：**

**Step 1. Test the EM predictor with forward\_code /electro\_predictor.py. Validate the model; The average relative error in validation set (l1) is printed.**

**Step 2. Test the Multiphysics predictor with** **forward\_code/ electro\_thermal\_predictor.py. Validate the model; The average relative error in validation set (l2) is printed.**

**Average temperature by latent dynamics networks:**



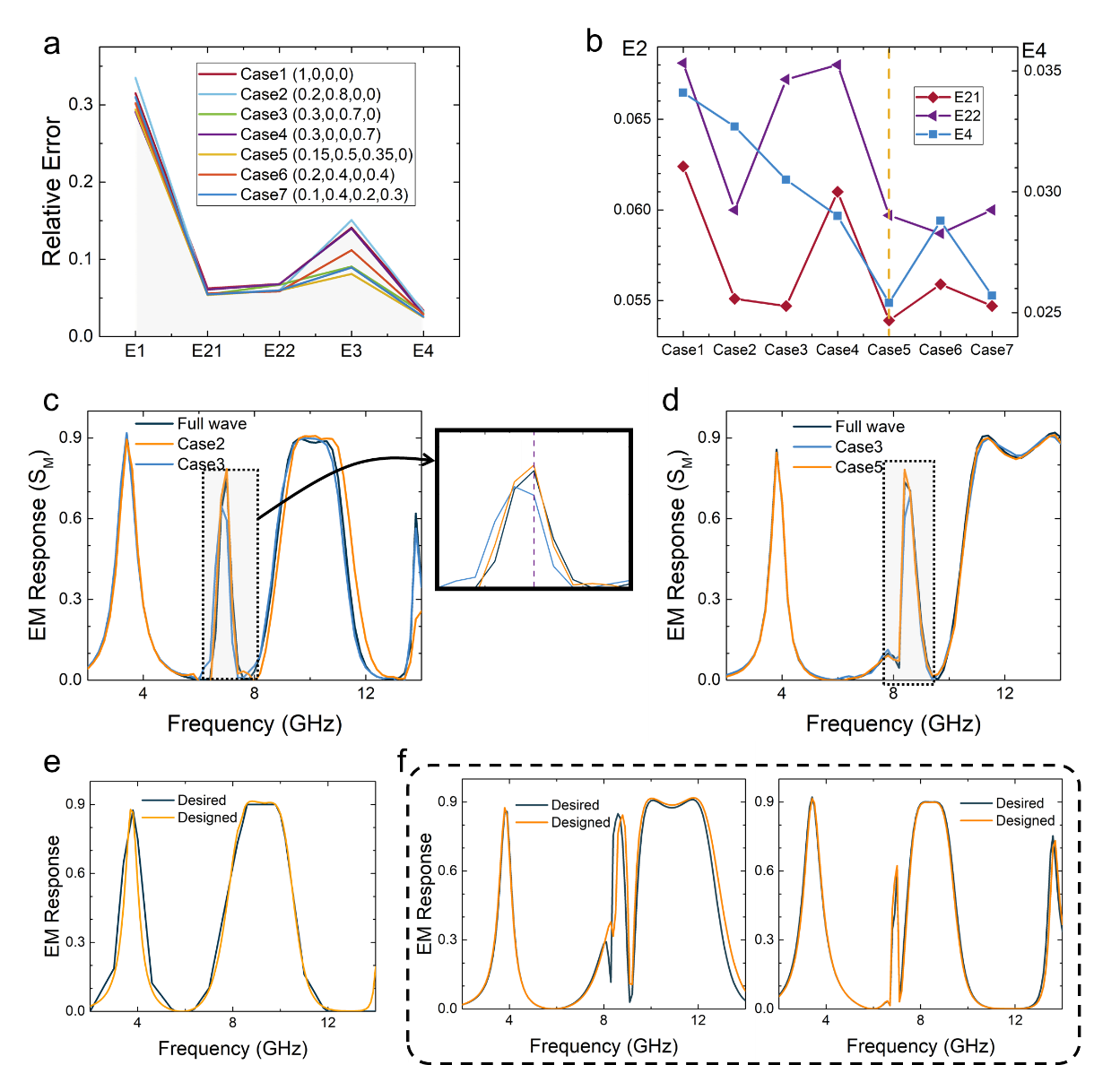
**There are 2 steps in total in this shared code：**

**Step 1. Train/test the Multiphysics predictor with**

**forward\_code /temperature\_predictor.py.**

**Step 2. Validate the model with forward\_code/temperature\_predictor.py: he results are in variable *yy* and the ground truth are in variable *y1*. The average relative error in training frequency range is e1, and that of frequency transfer is e2.**

**Inverse design by data-analytical driven networks:**



**In the development of inverse design network, the weights in (4) are important to direct the training. To test the model, run inverse\_design\_code/inverse\_design.py with the trained model, the average relative error in validation set (from E1 to E4) are printed. Model without input\_T is tested in**

**inverse\_design\_code/** **inverse\_design\_without\_input\_T.py**