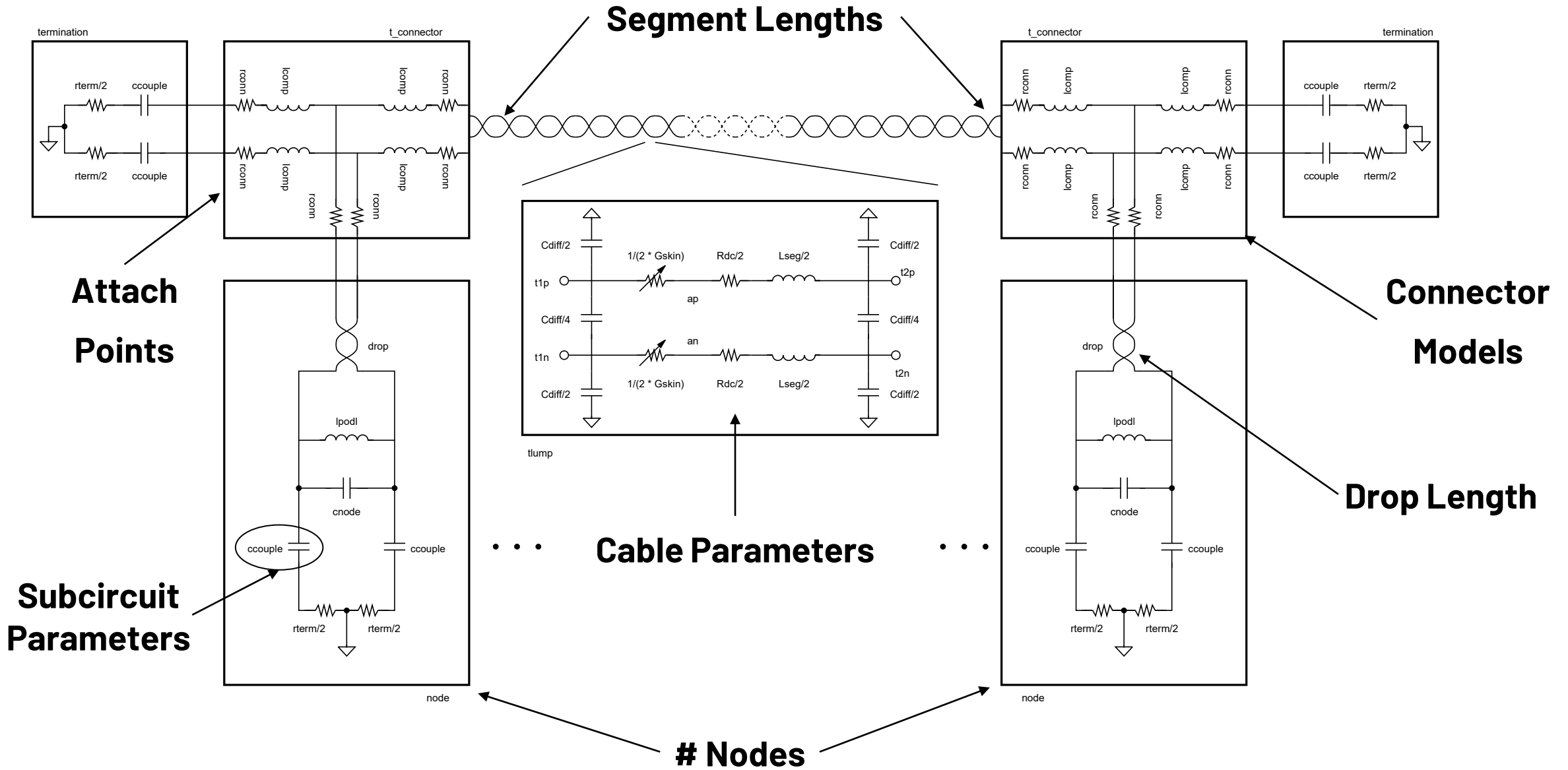


Consensus Model Update

Michael Paul

Multi-drop System : Algorithmic Model Assembly



Consensus Model

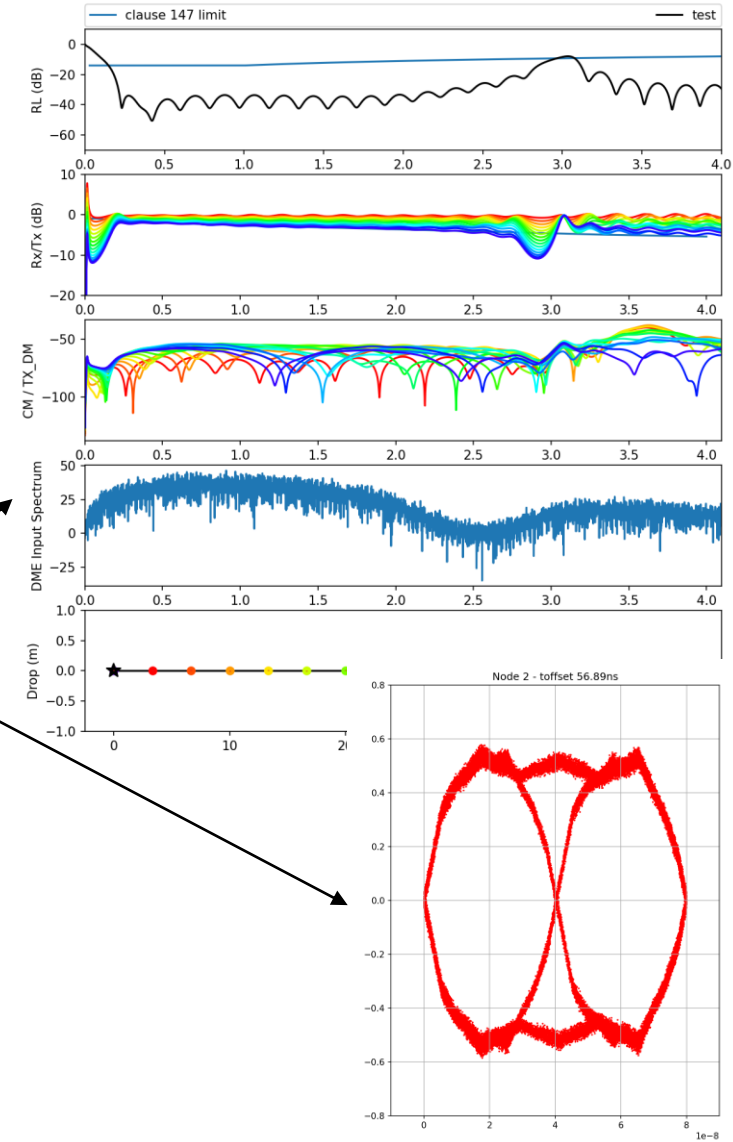
Configure System (.json)
e.g. your_test.json

`python cmodel.py -json=your_test.json`

Build netlist

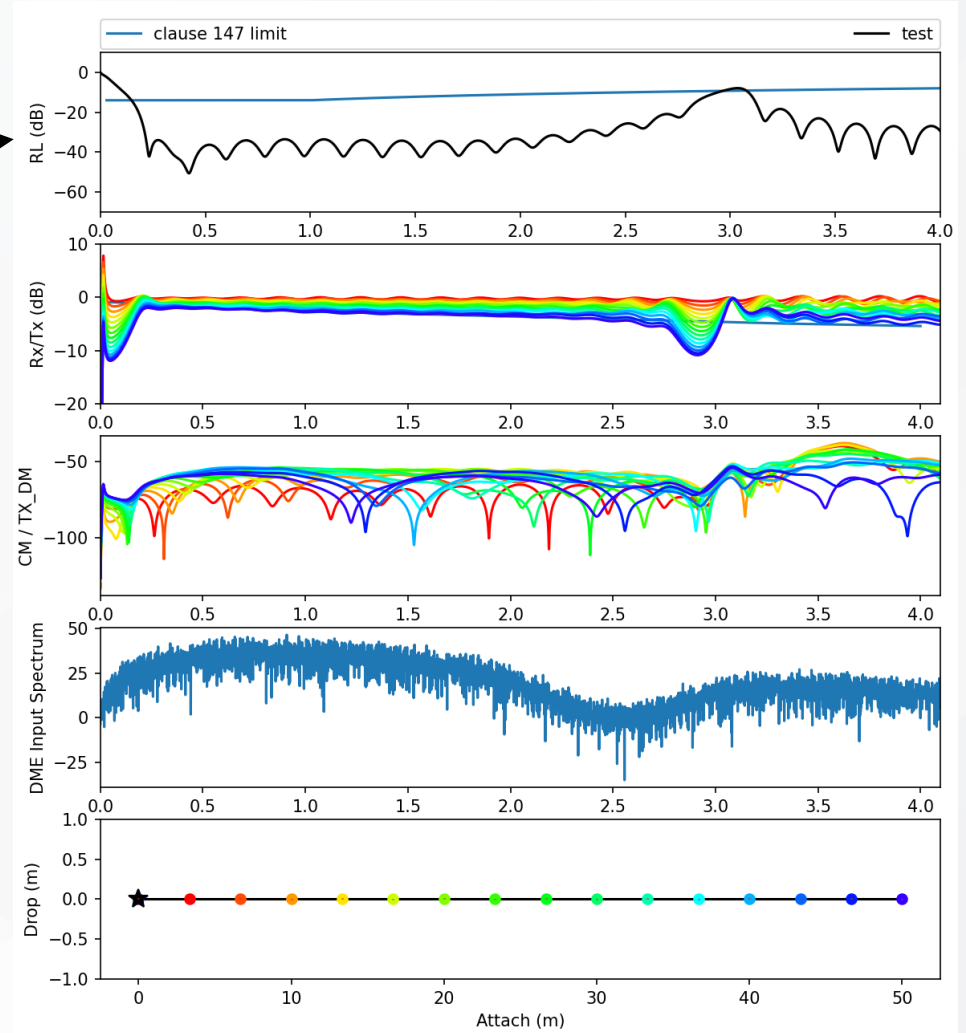
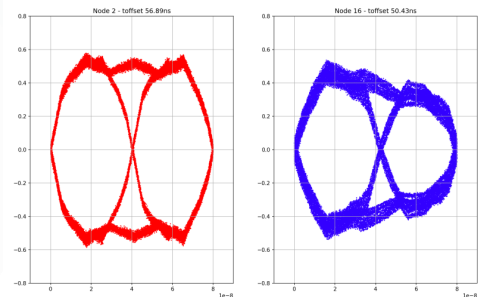
Run LTSpice

Extract + Process Data



Model Outputs

- ▶ Return Loss
- ▶ Insertion Loss
- ▶ Common Mode Conversion
- ▶ Input Power Spectral Density
- ▶ Simple network model
- ▶ Eye Diagrams



Starting the model

```
defaults.json  
"defaults_comment": "This are the default configuration parameters. These can be overridden",  
"includes": [  
  "include lump2.p",  
  "include node.p"  
],  
"analysis": "ac",  
"nodes": 16,  
"length": 50,  
"random_attach": false,  
"start_attach": 0,  
"end_attach": 0,  
"start_pad": 0,  
"end_pad": 0,  
"separation_min": 1.0,  
"segments_per_meter": 20,  
"drop_max": 0.02,  
"random_drop": false,  
"seed": -1,  
"tx_node": 1,  
"attach_error": 0,  
"attach_points": null,  
"autoscale": true,  
"noplot": false,  
"plot_png_filename": "cable.png",  
"eye_adjust": [0, 0],  
"default_termination": {  
  "comment": "cmatch declares how well the terminations match (in pct), 0 means perfect match. -1 means c",  
  "cmatch": 100,  
  "rctrl": 0,  
  "ccouple": 220e-9,  
  "cmatch": 0.00  
},  
"default_node": {  
  "drop_length": 0.01,  
  "random_drop": false,  
  "node": "50-12",  
  "lcomp": "50a-9",  
  "lpad": "80a-6",  
  "mode": "10a3",  
  "lcomp_match": 1,  
  "spice_model": "node",  
  "drop_model": null  
},  
"segment": {  
  "default_1kg":
```

- ▶ Each model run outputs a file: **last_run.json**
- ▶ **last_run.json** contains an expanded description of the system
 - Fully instantiated cable segments and nodes. etc
- ▶ Edit last_run.json to make new experiments with heterogeneous cable segments and nodes
- ▶ Refer to description.json for help configuring parameters

python cmodel.py -json=<your_config>.json

Build netlist

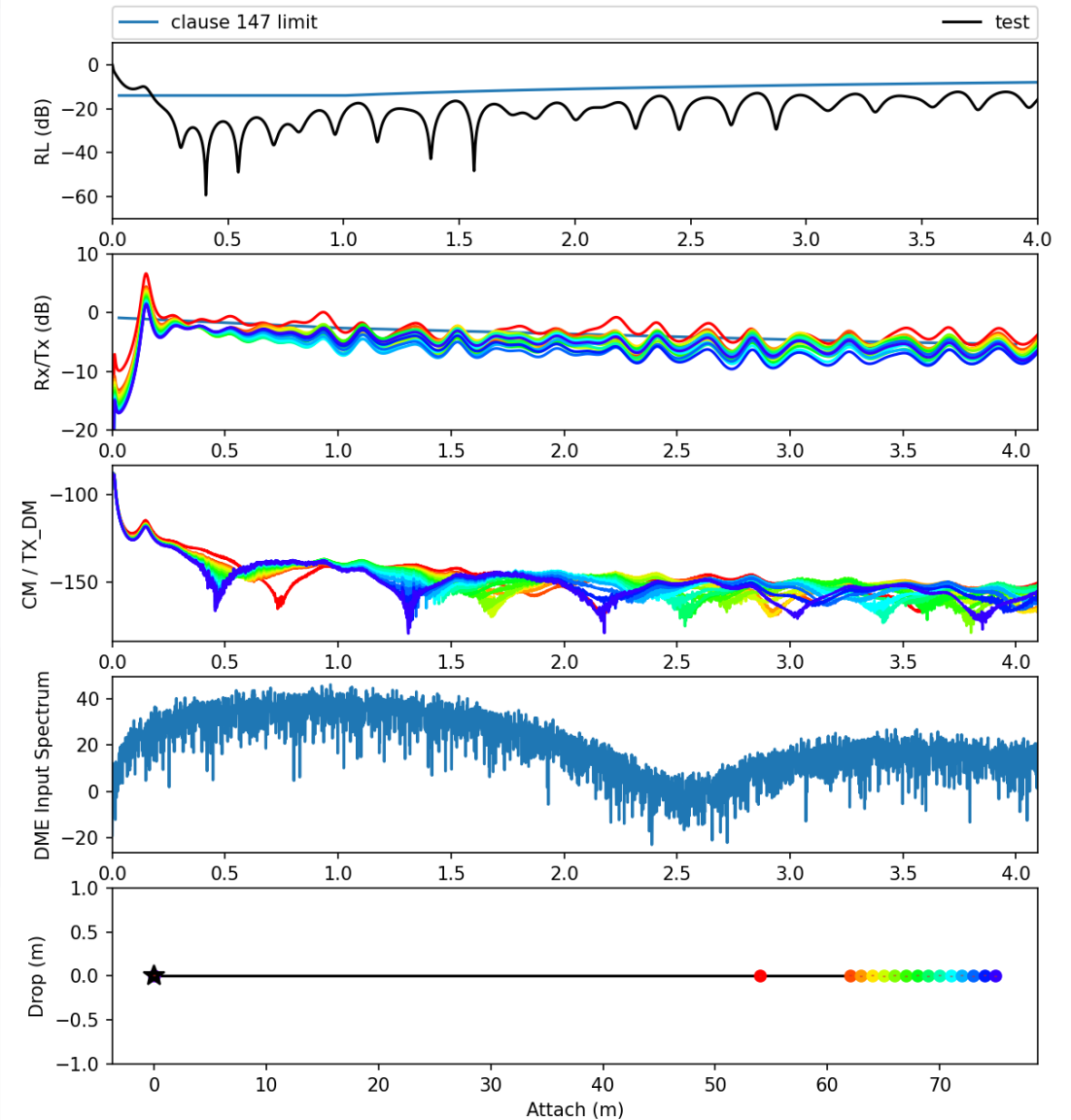
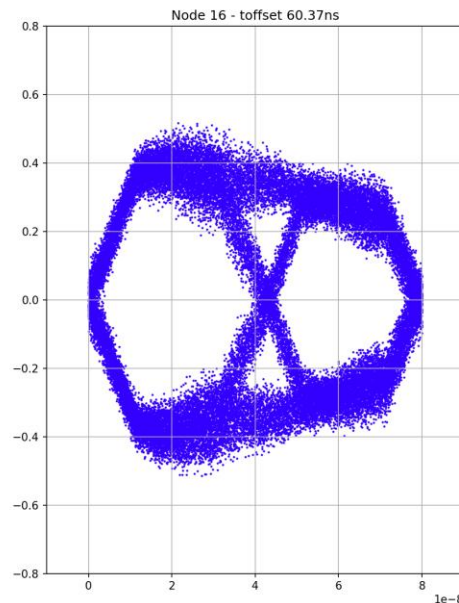
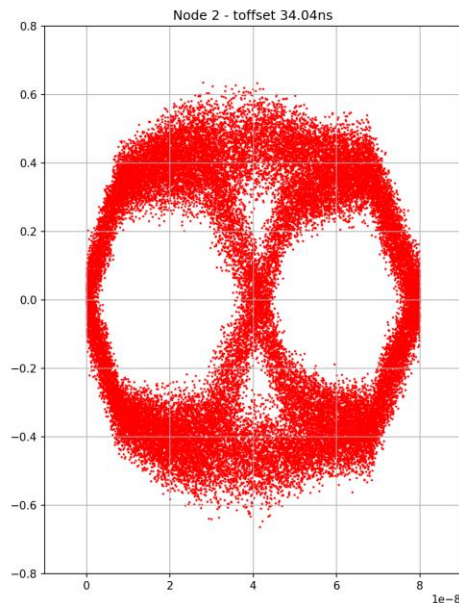
Run LTSpice

Extract + Process Data

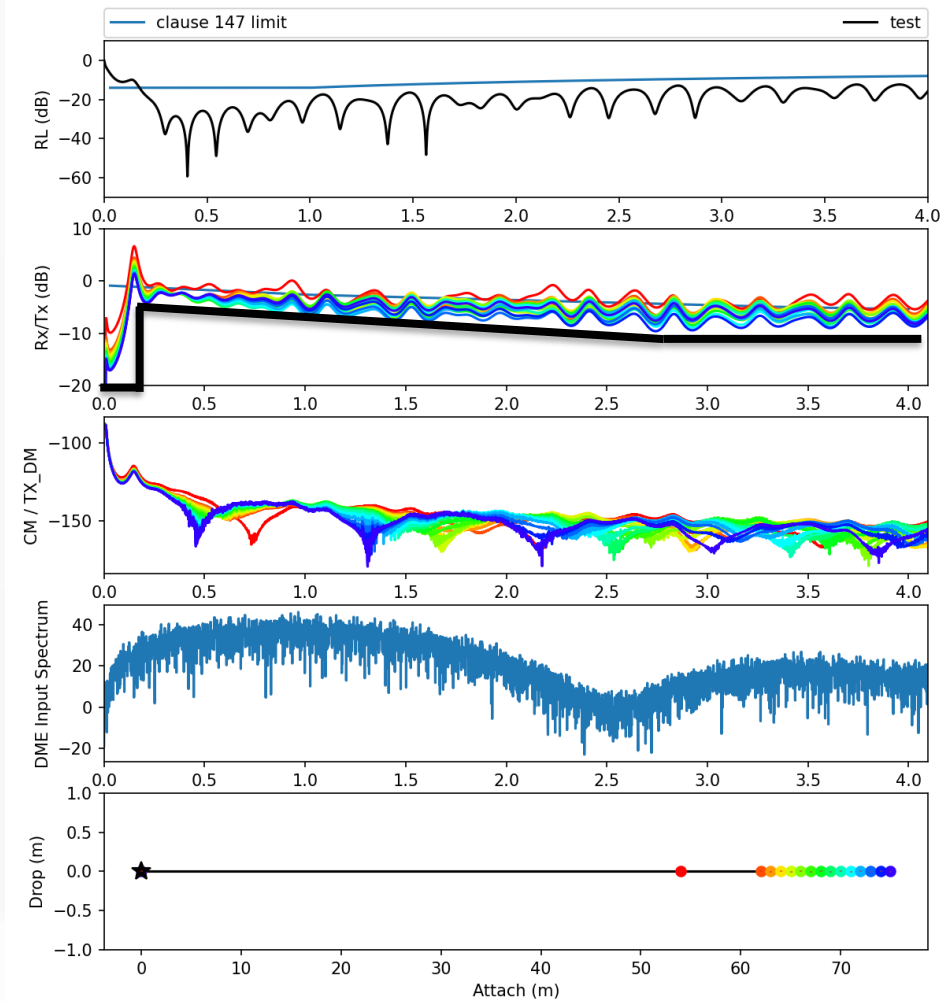
```
last_run.json  
"defaults_comment": "This are the default configuration parameters. These can be overridden",  
"includes": [  
  "include lump2.p",  
  "include node.p"  
],  
"analysis": "ac",  
"nodes": 16,  
"length": 50,  
"random_attach": false,  
"start_attach": 0,  
"end_attach": 0,  
"start_pad": 0,  
"end_pad": 0,  
"separation_min": 1.0,  
"segments_per_meter": 20,  
"drop_max": 0.02,  
"random_drop": false,  
"seed": -1,  
"tx_node": 1,  
"attach_error": 0,  
"attach_points": null,  
"autoscale": true,  
"noplot": false,  
"plot_png_filename": "cable.png",  
"eye_adjust": [0, 0],  
"default_termination": {  
  "comment": "cmatch declares how well the terminations match (in pct), 0 means perfect match. -1 means c",  
  "cmatch": 100,  
  "rctrl": 0,  
  "ccouple": 220e-9,  
  "cmatch": 0.00  
},  
"default_node": {  
  "drop_length": 0.01,  
  "random_drop": false,  
  "node": "50-12",  
  "lcomp": "50a-9",  
  "lpad": "80a-6",  
  "mode": "10a3",  
  "lcomp_match": 1,  
  "spice_model": "node",  
  "drop_model": null  
},  
"segment": {  
  "default_1kg":
```

Limit Lines Need Adjustment

- ▶ Eyes are open but insertion loss exceeds limits
- ▶ Need to adjust RL limits
- ▶ What is minimum required eye opening?



- ▶ Need to address RL / IL limit lines for standard to proceed
 - Automate multi-run data extraction
 - Form new limit lines based on data
 - Provide cutouts for power coupling inductance
 - Correlate new limit lines to eye openings



New IL Limit Line —

Thank You

GitHub Repository:

https://github.com/SPE-MD/SPMD-Simulations/ADI_Model/