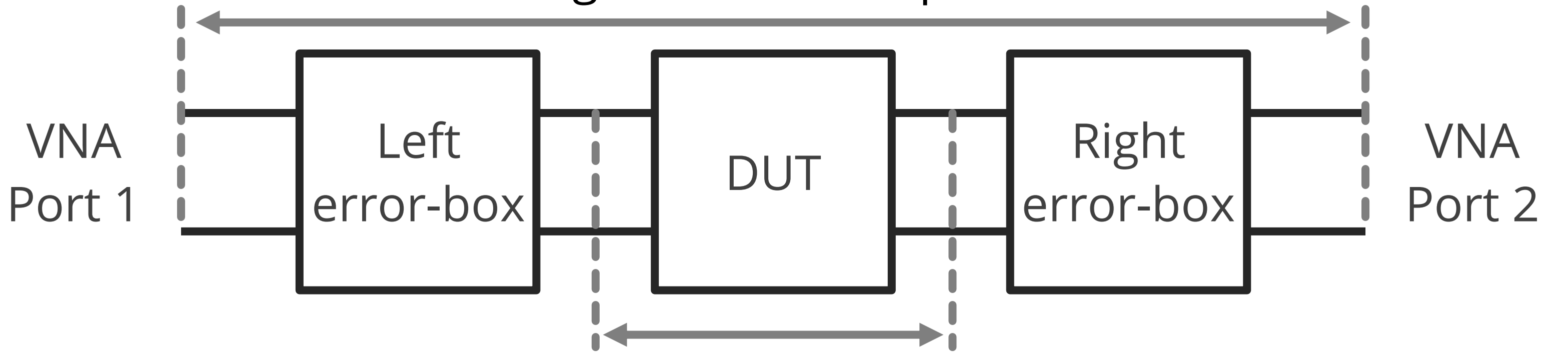
Improving the Reliability of the Multiline TRL Calibration Algorithm

Ziad Hatab[#*], Michael Gadringer[#], Wolfgang Bosch[#*]

[#]Graz University of Technology, Graz, Austria
*Christian Doppler Laboratory – TONI, Graz, Austria

ARFTG-98th Las Vegas, Nevada, USA
January 17th - 18th, 2022

VNA Error-box Model



- Measurement is modeled using the cascade-parameters (T-parameters):

$$\boldsymbol{M}_{DUT} = \underbrace{k_a \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & 1 \end{bmatrix}}_{\text{left error-box}} \boldsymbol{T}_{DUT} \underbrace{\begin{bmatrix} b_{11} & b_{12} \\ b_{21} & 1 \end{bmatrix} k_b}_{\text{right error-box}}$$

- Combine k_a and k_b to form the 7-term model (6 in $\boldsymbol{A}$ and $\boldsymbol{B}$, and k is the 7$^{\text{th}}$):

$$\boldsymbol{M}_{DUT} = k\boldsymbol{A}\boldsymbol{T}_{DUT}\boldsymbol{B}$$

Thru-Reflect-Line (TRL) Calibration

- **Thru** standard, fully known:

$$\boldsymbol{M}_{Thru} = k\boldsymbol{A}\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\boldsymbol{B}$$

- **Line** standard, only length known:

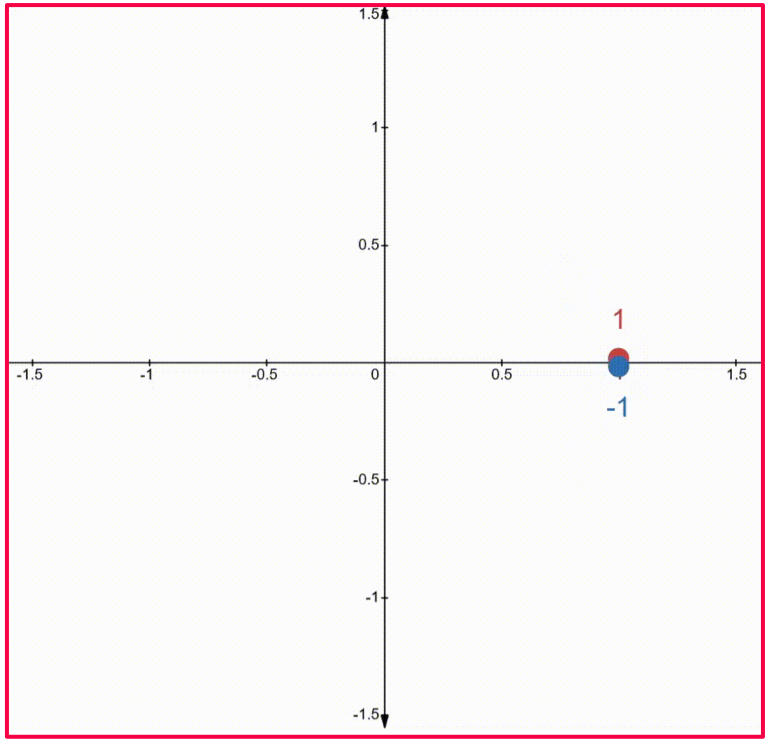
$$\boldsymbol{M}_{Line} = k\boldsymbol{A}\begin{bmatrix}e^{-\gamma l} & 0\\ 0 & e^{\gamma l}\end{bmatrix}\boldsymbol{B}$$

- The **eigenvalue** problem:

$$\boldsymbol{M}_{Line}\boldsymbol{M}_{Thru}^{-1} = \boldsymbol{A}\begin{bmatrix}e^{-\gamma l} & 0\\ 0 & e^{\gamma l}\end{bmatrix}\boldsymbol{A}^{-1}$$

When $e^{-\gamma l} = e^{\gamma l}$, the calibration becomes **unsolvable**.

- **Reflect** standard, unknown, but symmetric. Used with the **Thru** standard to finalize the calibration.



Extending TRL to Multiline TRL

- Use $N > 2$ lines. Results in $\frac{N(N-1)}{2}$ pairs:

$$\boldsymbol{M}_i\boldsymbol{M}_j^{-1} = \boldsymbol{A}\begin{bmatrix} e^{-\gamma(l_i-l_j)} & 0 \\ 0 & e^{\gamma(l_i-l_j)} \end{bmatrix}\boldsymbol{A}^{-1}$$

- **Question:** How to combine the solutions (eigenvectors) of all pairs?
- **Idea:** [R. B. Marks, MTT, 39, (1991)]

 Combine the eigenvectors using the Gauss-Markov theorem (weighted sum).

- Constraints:
  1) Error propagation through the eigenvectors (error-boxes) are assumed linear.
  2) Only $N-1$ pairs allowed to be used, where one line is common among all pairs.
     - ➢ allows for an invertible covariance matrix for the Gauss-Markov theorem.

Can we do better?

Questions:

Should we enforce linearization?

Must we use Gauss-Markov method to get the best results?

To overcome this, we need to **re-develop** multiline TRL.

Kronecker Product "⊗" and Matrix Vectorization

- Given matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$, their Kronecker Product is defined as:

$$\boldsymbol{X} \otimes \boldsymbol{Y} = \begin{bmatrix} x_{11}\boldsymbol{Y} & x_{12}\boldsymbol{Y} & \cdots \\ x_{21}\boldsymbol{Y} & x_{22}\boldsymbol{Y} & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix}$$

- Vectorization of a matrix $\boldsymbol{X}$ is defined by stacking its columns:

$$\boldsymbol{X} = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{bmatrix} \xrightarrow{\text{vec}()} \text{vec}(\boldsymbol{X}) = \begin{bmatrix} x_{11} \\ x_{21} \\ x_{12} \\ x_{22} \end{bmatrix}$$

- **Vectorization of matrix product:** given $\boldsymbol{X}$, $\boldsymbol{Y}$ and $\boldsymbol{Z}$, then:

$$\text{vec}(\boldsymbol{XYZ}) = (\boldsymbol{Z}^T \otimes \boldsymbol{X})\text{vec}(\boldsymbol{Y})$$

Reformulating mTRL Error-box Model (1/2)

- Measurement of i-th line standard:

$$\boldsymbol{M}_i = k\boldsymbol{A}\boldsymbol{L}_i\boldsymbol{B}$$

- Vectorization of $\boldsymbol{M}_i$:

$$\mathrm{vec}(\boldsymbol{M}_i) = k(\boldsymbol{B}^T \otimes \boldsymbol{A})\mathrm{vec}(\boldsymbol{L}_i)$$

- Including all N lines:

$$\underbrace{[\mathrm{vec}(\boldsymbol{M}_1) \quad \cdots \quad \mathrm{vec}(\boldsymbol{M}_N)]}_{\boldsymbol{M}} = k(\boldsymbol{B}^T \otimes \boldsymbol{A})\underbrace{[\mathrm{vec}(\boldsymbol{L}_1) \quad \cdots \quad \mathrm{vec}(\boldsymbol{L}_N)]}_{\boldsymbol{L}}$$

$$\boldsymbol{M} = k(\boldsymbol{B}^T \otimes \boldsymbol{A})\boldsymbol{L}$$

$4 \times N$ $\quad$ 4×4 $\quad$ $4 \times N$

Eq. 1

Reformulating mTRL Error-box Model (2/2)

- Inverse measurement of i-th line standard:

$$\boldsymbol{M}_i^{-1} = \frac{1}{k}\boldsymbol{B}^{-1}\boldsymbol{L}_i^{-1}\boldsymbol{A}^{-1}$$

- Vectorization of $\boldsymbol{M}_i^{-1}$:

$$\mathrm{vec}\left(\boldsymbol{M}_i^{-1}\right) = \frac{1}{k}(\boldsymbol{A}^{-T} \otimes \boldsymbol{B}^{-1})\mathrm{vec}(\boldsymbol{L}_i^{-1})$$

- Including all N lines, and applying **some tricks** (see our paper!):

$$\underbrace{\boldsymbol{D}^{-1}\boldsymbol{M}^T\boldsymbol{P}\boldsymbol{Q}}_{N\times 4} = \frac{1}{k}\underbrace{\boldsymbol{L}^T\boldsymbol{P}\boldsymbol{Q}}_{N\times 4}\underbrace{(\boldsymbol{B}^T \otimes \boldsymbol{A})^{-1}}_{4\times 4}$$

Eq. 2

$$\boldsymbol{P} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; \quad \boldsymbol{Q} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}; \quad \boldsymbol{D} = \mathrm{diag}([\det(\boldsymbol{M}_1) \quad \cdots \quad \det(\boldsymbol{M}_N)])$$

Solving for the Propagation Constant γ

- Simplifying notation by letting $\boldsymbol{X} = (\boldsymbol{B}^T \otimes \boldsymbol{A})$

$$\boldsymbol{M} = k\boldsymbol{X}\boldsymbol{L} \quad (1)$$

$$\boldsymbol{D}^{-1}\boldsymbol{M}^T\boldsymbol{P}\boldsymbol{Q} = \frac{1}{k}\boldsymbol{L}^T\boldsymbol{P}\boldsymbol{Q}\boldsymbol{X}^{-1} \quad (2)$$

- Multiplying Eq. (1) to the **right** of Eq. (2):

$$\underbrace{\boldsymbol{D}^{-1}\boldsymbol{M}^T\boldsymbol{P}\boldsymbol{Q}\boldsymbol{M}}_{\text{Measurement}} = \underbrace{\boldsymbol{L}^T\boldsymbol{P}\boldsymbol{Q}\boldsymbol{L}}_{\text{Line Model}}$$

- Find γ through optimization:

$$\gamma_{opt} = \min_{\gamma}\left\|\boldsymbol{D}^{-1}\boldsymbol{M}^T\boldsymbol{P}\boldsymbol{Q}\boldsymbol{M} - \boldsymbol{L}_{(\gamma)}^T\boldsymbol{P}\boldsymbol{Q}\boldsymbol{L}_{(\gamma)}\right\|_F^2$$

Formulating the Eigenvalue Problem (1/2)

- We multiply an $N \times N$ matrix $\boldsymbol{W}$ to Eq. (1):

$$\boldsymbol{M}\boldsymbol{W} = k\boldsymbol{X}\boldsymbol{L}\boldsymbol{W} \quad (1)$$

$$\boldsymbol{D}^{-1}\boldsymbol{M}^T\boldsymbol{P}\boldsymbol{Q} = \frac{1}{k}\boldsymbol{L}^T\boldsymbol{P}\boldsymbol{Q}\boldsymbol{X}^{-1} \quad (2)$$

- Then, we multiply Eq. (1) to the **left** of Eq. (2):

$$\underbrace{\boldsymbol{M}\boldsymbol{W}\boldsymbol{D}^{-1}\boldsymbol{M}^T\boldsymbol{P}\boldsymbol{Q}}_{\boldsymbol{F}} = \boldsymbol{X}\underbrace{\boldsymbol{L}\boldsymbol{W}\boldsymbol{L}^T\boldsymbol{P}\boldsymbol{Q}}_{\boldsymbol{H}}\boldsymbol{X}^{-1}$$

- We end up with a similarity equation:

$$\boldsymbol{F} = \boldsymbol{X}\boldsymbol{H}\boldsymbol{X}^{-1}$$

- If $\boldsymbol{H}$ is diagonal, then we have an eigenvalue problem.

Formulating the Eigenvalue Problem (2/2)

- It turns out that if $\boldsymbol{W}$ is skew-symmetric, then $\boldsymbol{H}$ becomes diagonal:

$$\boldsymbol{W} = \begin{bmatrix} 0 & w_{12} & w_{13} & w_{14} & \cdots \\ -w_{12} & 0 & w_{23} & w_{24} & \ddots \\ -w_{13} & -w_{23} & 0 & w_{34} & \ddots \\ -w_{14} & -w_{24} & -w_{34} & 0 & \ddots \\ \vdots & \ddots & \ddots & \ddots & \ddots \end{bmatrix} \xrightarrow{\text{yields}} \boldsymbol{H} = \begin{bmatrix} -\lambda & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda \end{bmatrix}$$

$$\lambda = \sum_{\substack{i=1 \\ i<j\le N}}^{N-1} w_{ij} \cdot \left(e^{\gamma(l_i - l_j)} - e^{-\gamma(l_i - l_j)}\right)$$

- The eigenvalue problem is only solvable if $|\lambda| > 0$. This is enforced if:

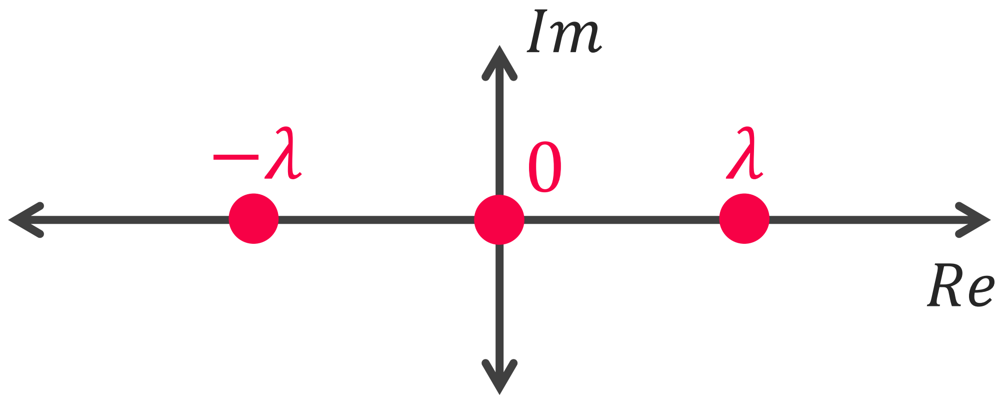
$$w_{ij} = \operatorname{conj}\left(e^{\gamma(l_i - l_j)} - e^{-\gamma(l_i - l_j)}\right)$$

Solving the Calibration Problem

- The eigenvalue problem:

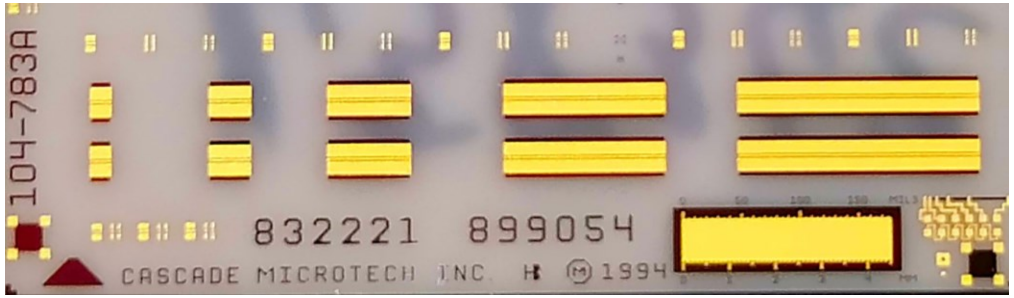
$$\boldsymbol{F} = \boldsymbol{X}\begin{bmatrix} -\lambda & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda \end{bmatrix}\boldsymbol{X}^{-1}$$

$$\lambda = \sum_{\substack{i=1 \\ i<j\leq N}}^{N-1} \left| e^{\gamma(l_i - l_j)} - e^{-\gamma(l_i - l_j)} \right|^2$$



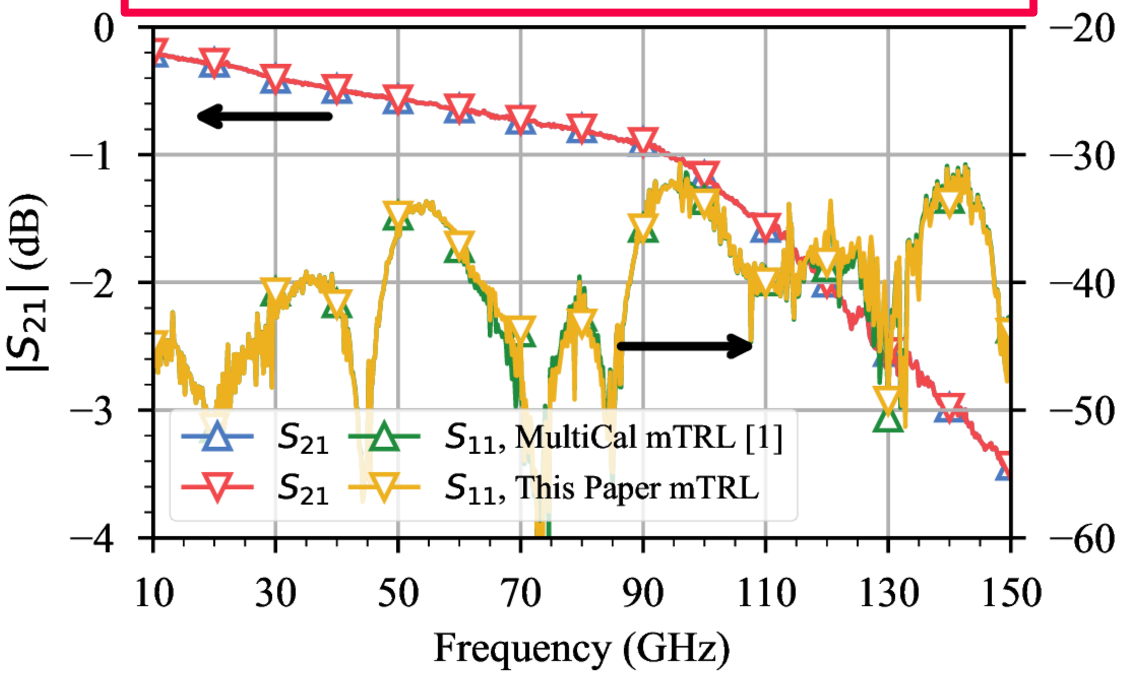
- The eigenvectors of $\boldsymbol{F}$ are the columns of $\boldsymbol{X}$ (up to a scalar factor).
- The rest of the calibration is solved using the **Reflect** and **Thru** standards.

On-wafer Measurements

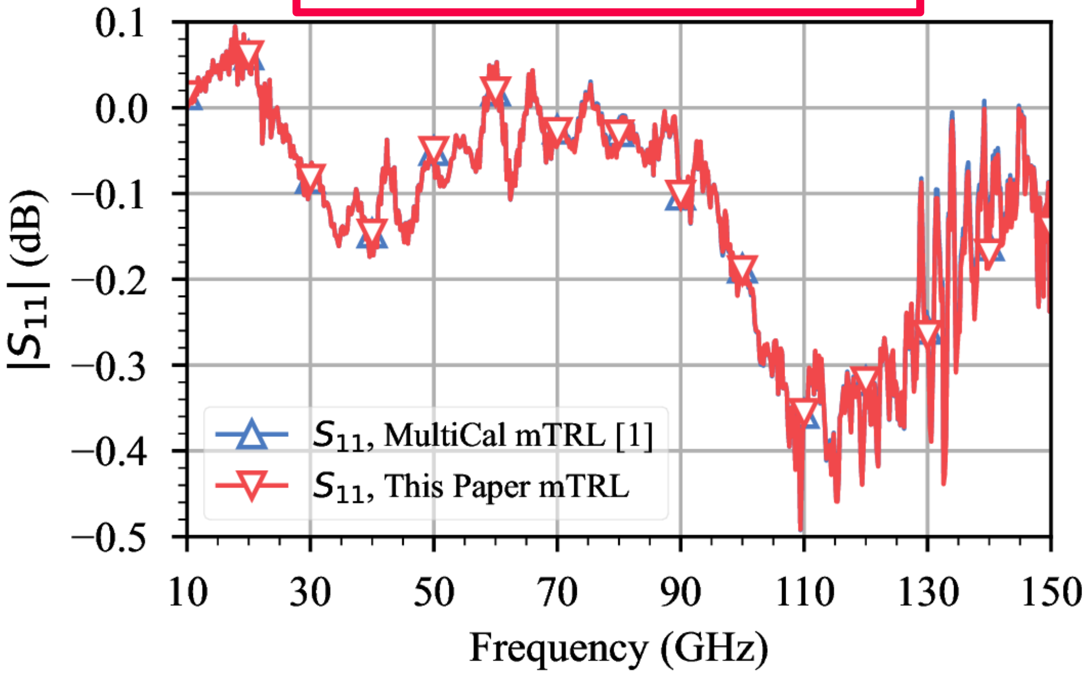


- **Calibration standards:** 4 CPWs lines {0.2, 0.45, 0.9, 1.8}mm and a Short.
- 1 CPW line 3.5mm used for verification (not part of the calibration).

Calibrated verification line: 3.5mm



Calibrated short

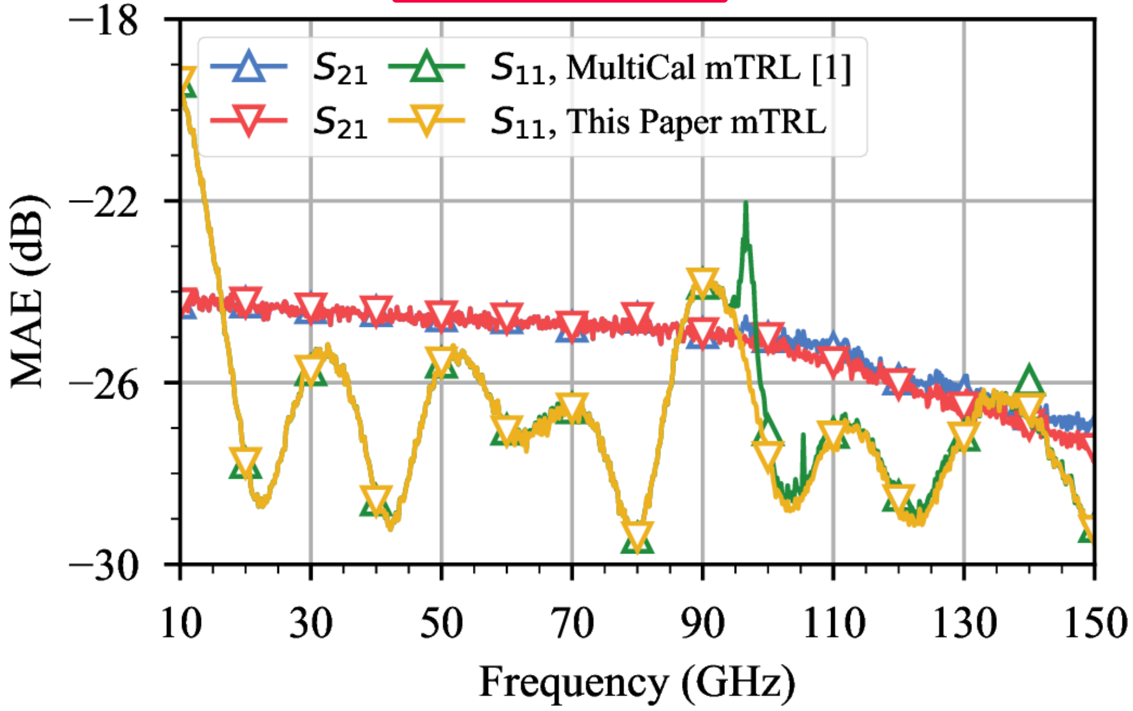


[1] D. DeGroot, J. Jargon, and R. Marks, “Multiline trl revealed,” Washington, DC, USA: IEEE, 2002, pp. 131–155.

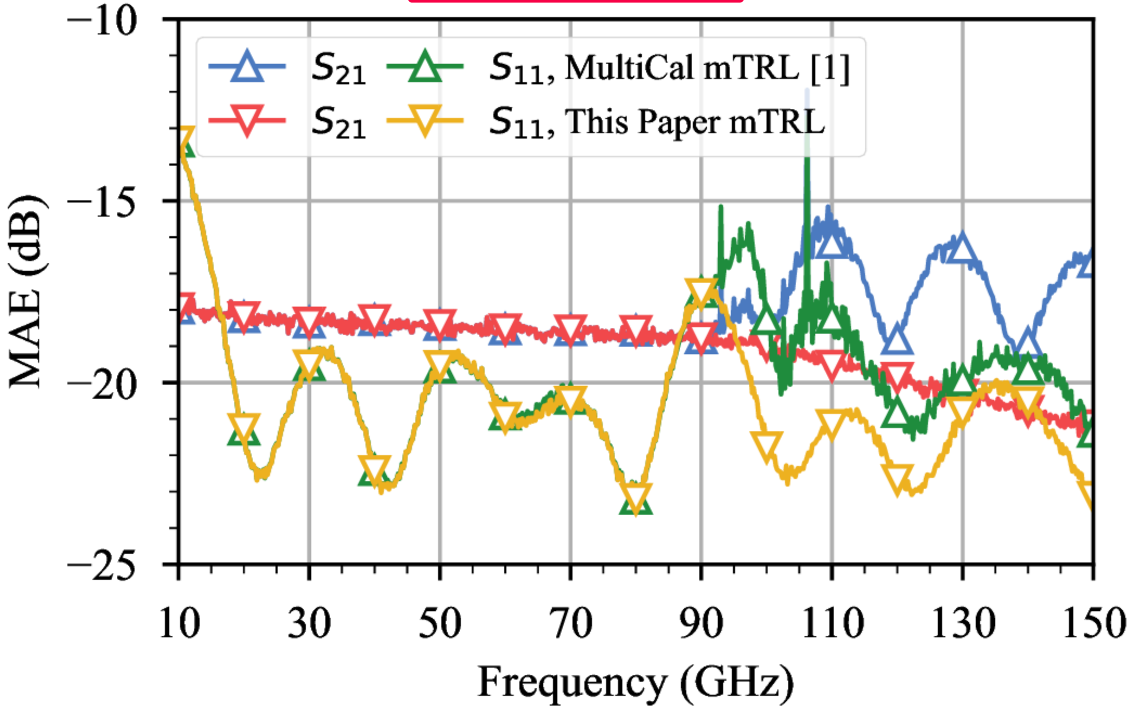
Monte Carlo Experiment – Additive Noise

- Add noise to the measurements of the standards: $S_{ij}^{New} = S_{ij}^{Meas} + Noise$
- Perform mTRL using distorted measurements at each trial ($M = 2000$ trials).
- Calibrate a DUT and compute its **Mean-Absolute-Error**: $\mathrm{MAE}(S_{ij}) = \frac{1}{M}\sum_{m=1}^{M}\left|S_{ij,m}^{MC} - S_{ij}\right|$

$\sigma = 0.05$



$\sigma = 0.1$

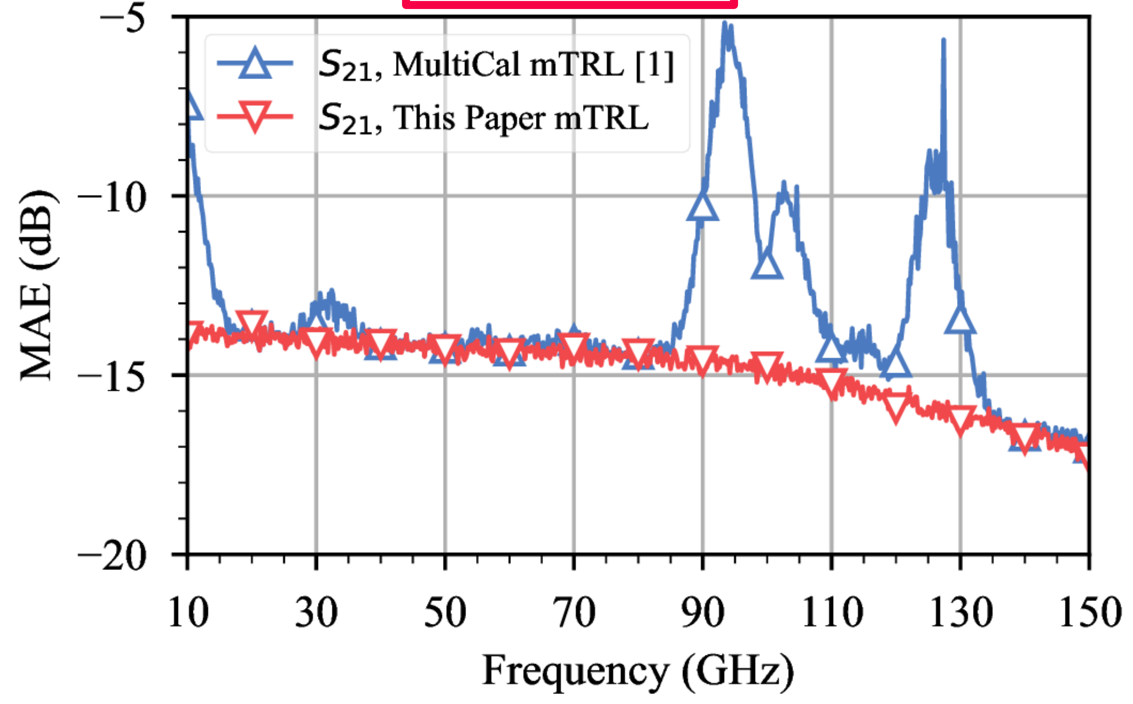


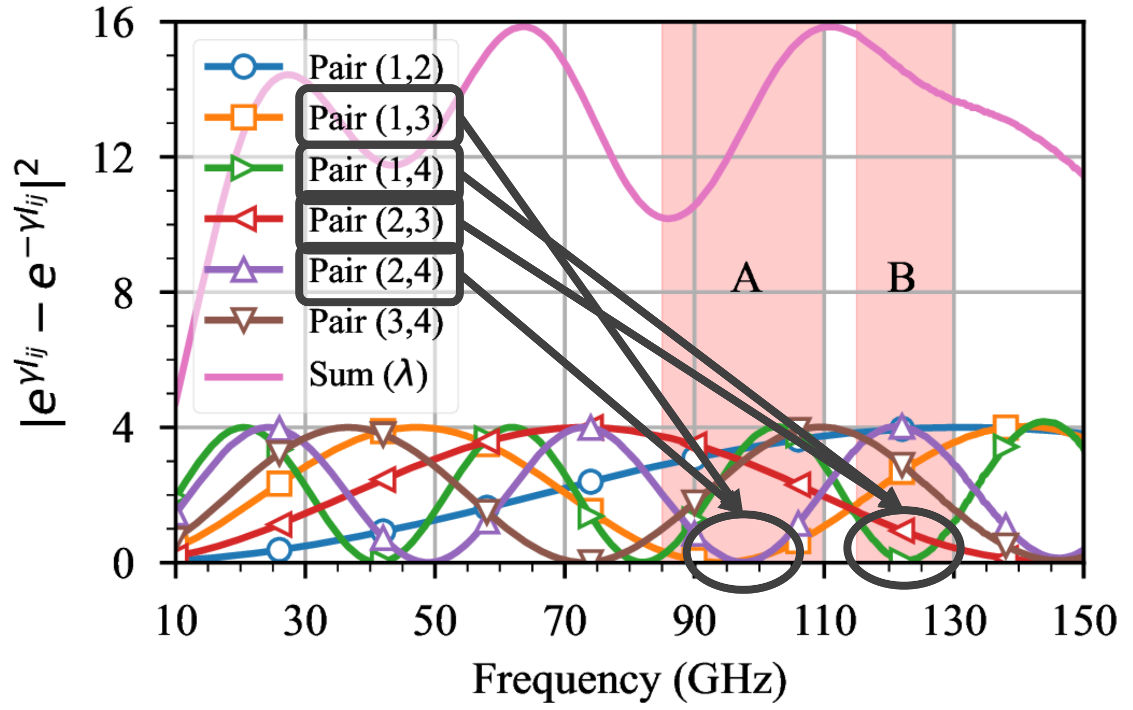
[1] D. DeGroot, J. Jargon, and R. Marks, “Multiline trl revealed,” Washington, DC, USA: IEEE, 2002, pp. 131–155.

Monte Carlo Experiment – Phase Sensitivity

- Distort the phase of the measurements of the standards: $\arg(S_{ij})^{New} = \arg(S_{ij})^{Meas} + Noise$
- Perform mTRL using distorted measurements at each trial ($M = 2000$ trials).
- Calibrate a DUT and compute its **Mean-Absolute-Error**: $\mathrm{MAE}(S_{ij}) = \frac{1}{M}\sum_{m=1}^{M}\left|S_{ij,m}^{MC} - S_{ij}\right|$

$\sigma = 15^{\circ}$





[1] D. DeGroot, J. Jargon, and R. Marks, “Multiline trl revealed,” Washington, DC, USA: IEEE, 2002, pp. 131–155.

Summery

- VNA error-box model can be simplified with the help of Kronecker product and matrix vectorization.
- Combining all line measurement with a self-derived weighting matrix.
- No matrix inverse. No covariance matrices.
- Solving a single eigenvalue problem.
- Better statistical performance.
- **Scalability**. For example, if you have **1000 Lines**, you will still solve a single 4×4 eigenvalue problem. Imagine if you do that the old way!!

https://github.com/ZiadHatab/multiline-trl-calibration

Check my github repository and try the algorithm yourself.
Feedbacks are very, very welcomed!!!

Contact: z.hatab@tugraz.at or zi.hatab@gmail.com
You can also reach me at Twitter: @ziad_hatab