Notice for UVNA-63 Application Notes

Application Notes for the UVNA-63 are provided as a guide to help our customers obtain a better understanding of the theory and math behind building a VNA. Each Application Note will provide examples and experiments that describe the functionality and calibration methods to make accurate measurements using a VNA.

Not all technical details are covered in these notes; however customers can find these details in the references mentioned at the end of each Application Note. These notes are not intended to replace technical textbooks.

Each Application Note will state functions that our customers can code using Python/MATLAB with the guidance of the same Application Note. Further details or function descriptions can be found at the following links:
Python: https://www.minicircuits.com/uvnadocs/Python/index.html
MATLAB: https://www.minicircuits.com/uvnadocs/MATLAB/index.html
Also, API commands are available in our Software manual:
https://www.minicircuits.com/pdfs/UVNA-63_Software_and_Programming_Guide.pdf

There may be some prerequisites or knowledge required in order to understand the Application Note. We recommend learning the prerequisites mentioned in the Application Notes before proceeding.

Learning Objectives will be mentioned clearly for particular Application Notes.

UVNA-63 Application Note

Error Correction

by Nikola Janjušević
Applications Department

VNA Kit Functions:

`get1PortResponseModel()`, `get2PortResponseModel()`, `correctResponse()`,
`get1PortModel()`, `correct1Port()`,
`get12TermModel()`, `correct12Term()`,
`get8TermModel()`, `correct8Term()`, `ab2S_SwitchCorrect()`

Module Prerequisites: Introduction to Network Analysis: What is a VNA?

Learning Objectives

- To understand systematic error in the context of a network analyzer.
- To understand the models used to describe systematic error in a network analyzer and what assumptions the models make.
- To be able to program correction models for a network analyzer.

Introduction

In any measurement setup, the tools used to perform the measurement will bring repeatable, systematic errors. In a network analyzer, there are a multitude of RF components between the sensor and the desired measurement plane. The components that aid in measuring and connecting to the device under test (DUT) bring their own frequency response to the measurement and distort the desired signals. As a result, the measured parameters are not those of the DUT, but of the entire measurement apparatus. Error correction methods attempt to model and remove the effect of the measurement apparatus, moving the measurement plane from the sensor to the connections of the DUT.

1 Reflectometers

In the following sections, we will be referring to the basic functional unit capable of making a measurement as a reflectometer. For a vector network analyzer, this most generally includes a directional coupler, IF converter (local oscillator and mixer), receiver, analog to digital converter, and some digital signal processing. Even more generally, a reflectometer may be described as the directional coupler plus the sensor. A notational simplification presented in Figure 1 is used throughout the entirety of this application note.

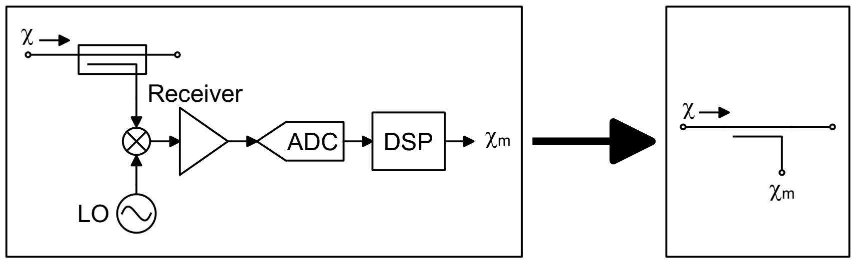


Figure 1: Notational simplification of a reflectometer

When $\chi_m = \chi$, we will refer to the reflectometer as a **perfect reflectometer**, which recovers the desired phasor with no error.

2 Sources of Systemic Error in a Network Analyzer

In the case of RF network measurements, simply having a cable that connects your sensor to the port of your DUT will add a time (and thus phase) shift to your measurements due to the finite speed of light. This is a *systematic* measurement error, as it is *repeatable* and is a *result of the measurement setup*. The following list shows the different types of systematic error present in a network analyzer and their main contributers [1]:

Types of Systematic Errors	**Main Contributers**
Directivity Error	Finite directivity of directional couplers
Reflection Tracking Error	Imperfect reflectometers
Transmission Tracking Error	Imperfect reflectometers (esp. mixer non-linearity)
Port-Match Error	Impedance mismatch of connections
Leakage Error	Local oscillators, switches, signal path proximity

Table 1: Systematic Errors and Thier Contributers

3 Frequency Response Correction

Frequency Response Calibration (a.k.a. Response Calibration) models the systematic error of a network analyzer as a filter response $H(\omega)$. Measured S-parameters are thus modeled as the actual S-parameters of the DUT passed through the network analyzer's channel, as described in *Equation* 1, as follows:

$$S_{ij_M}(\omega) = H(\omega) S_{ij}(\omega) \tag{1}$$

To obtain the actual S-parameters of the DUT, the inverse channel $H^{-1}(\omega)$ must be obtained. This error model only accounts for the reflection and transmission tracking terms of systematic error. If your network analyzer is only capable of measuring magnitude, this form of error correction may be performed only with each term in *Equation* 1 replaced with their respective magnitude. Such a correction, using only the magnitude of the channel, is often called Scalar Error Correction and is used most often by Scalar Network Analyzers (SNAs).

3.1 Exploration: Response Correction as "Normalization"

Response correction is not very accurate as it only accounts for 'tracking' error terms. It is most often performed to simply remove the most obvious effects of the measurement setup, such as reflectometer paths having noticeably different attenuations. In the following measurement set up (see Figure 2), the Reflect and Reference paths on each port have a total attenuation of around -10dB(+ mainline loss of couplers + cable losses) and -16dB(+ cable losses) respectively. This means an uncalibrated S-parameter measurement may have a noticeable amount of fictitious 'gain' from a passive device as the b wave measurement naturally has a lower path attenuation than the a wave measurement.

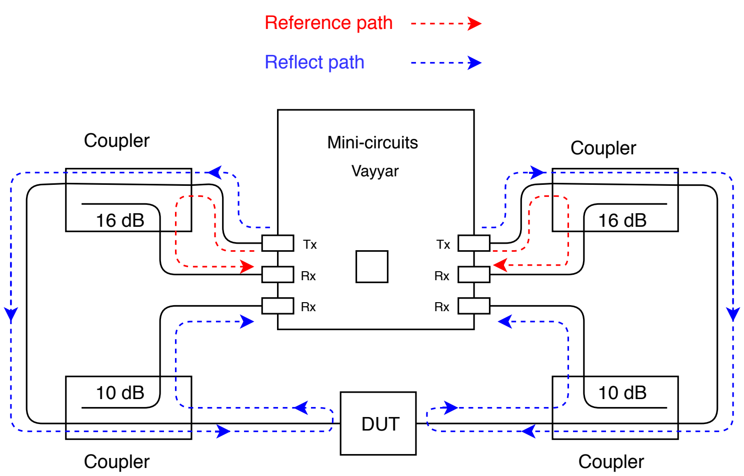


Figure 2: VNA Kit measurement setup with Reference (Red) and Reflect (Blue) paths shown

Figure 3 shows the results of performing a crude frequency response calibration with the UVNA-63 Kit in the measurement set up shown by Figure 2, with a 2.5 GHz bandpass filter as the DUT. The calibration was 'crude' in the sense that, to compute the error terms, open ended cables were used instead of an open standard from the calibration kit, and an unknown thru was used instead of an ideal known thru standard.

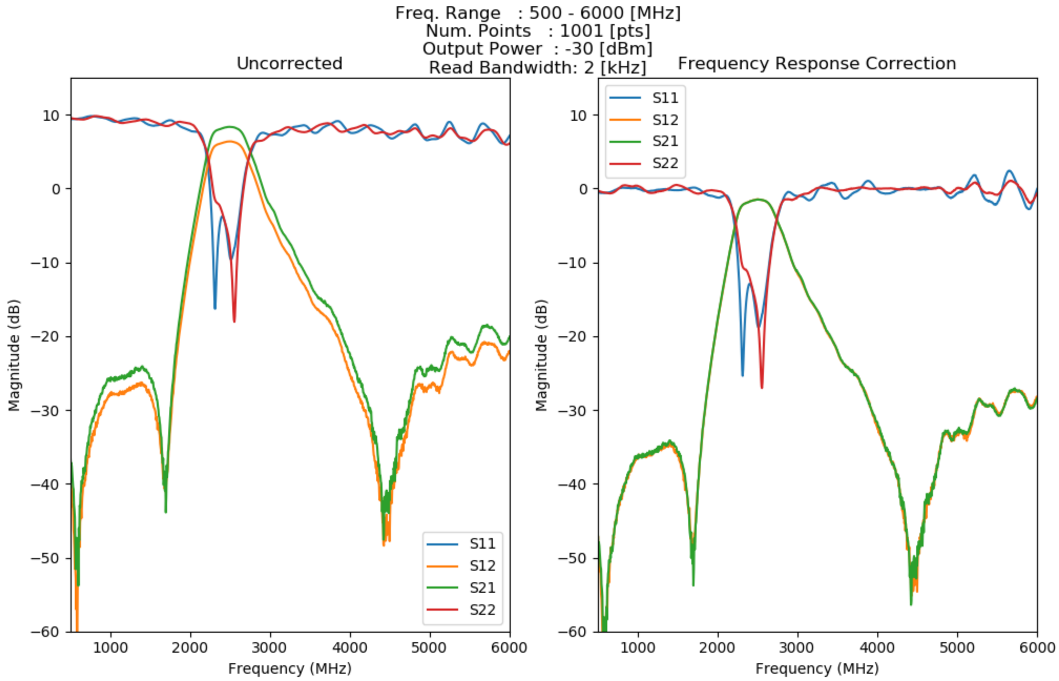


Figure 3: Uncorrected (left) and response corrected (right) measurement of a 2.5 GHz bandpass filter

As predicted, the uncorrected measurement shows significant 'gain' due to the measurement setup. The response calibrated measurement has corrected this glaring error of the uncorrected measurement by shifting the plot down to 0dB. However, significant ripple still remains in each of the traces of the corrected DUT. This is a result of response calibration not taking into account port-match or directivity error. Colliding a/b waves constructively and destructively interfere with each other to create these ripples over frequency.

Note

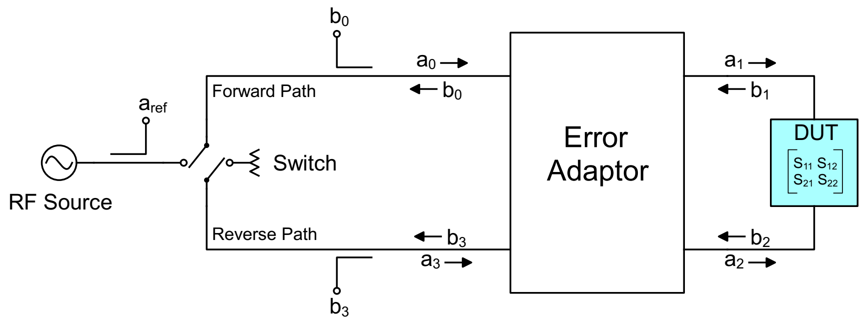
As the ideal reflection coefficient of an open-circuit and the transmission of an ideal thru are $\Gamma = 1$ and $S_{21} = S_{12} = 1$ respectively, applying a response correction to a measurement is often referred to as "dividing" or "normalizing by the trace" as the inverse channel is simply obtained by a division operation.

4 Vector Error Correction

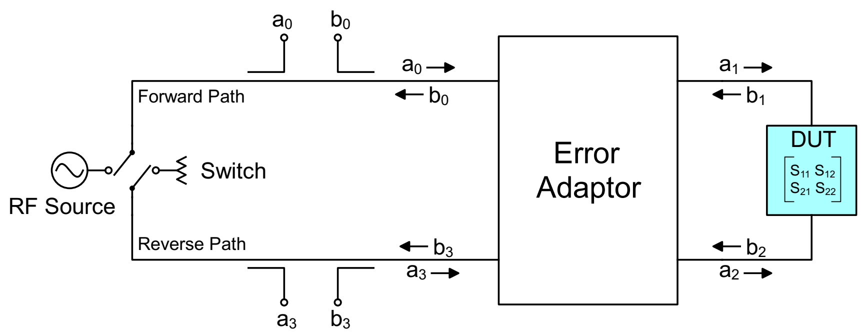
Vector error correction (VEC) takes a more sophisticated approach than response correction by modeling the measurement process of the VNA as a result of a fictitious RF-network, dubbed an *error adaptor*, that interfaces a set of *perfect reflectometers* in measuring the DUT. Signal flow graphs with error terms pertaining to the architecture of the VNA are used to describe the innards of the error adaptor.

4.1 VNA Architecture in the context of VEC

We must make the distinction between two Network Analyzer (NA) architectures as they will be referenced in later VEC sections; they are the 4-Reflectometer and the 3-Reflectometer architecture. As shown in Figure 4a below, the 3-Reflectometer architecture is able to use less components by putting the reflectometer for the reference signal before the RF source switch.



(a) 3-Reflectometer Architecture Error Model



(b) 4-Reflectometer Architecture Error Model

Figure 4: VNA architecture models in the context of VEC

***Notation*:**

In S-parameter theory, we denote $S_{mn} = \frac{b_m}{a_n}$, where $a_k = 0 \quad \forall \quad k \neq n$. In the following sections, we make a distinction between the *actual* S-parameters of a device under test (DUT) and the *measured* S-parameters of a DUT. Hence, there are four sets of a and b waves (port-1 actual & measured, port-2 actual & measured). These a and b waves are indexed from 0-3 such that the actual and measured S-parameters of the DUT have the following correspondence, respectively:

$$S = \begin{bmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{bmatrix}, \qquad S_M = \begin{bmatrix} S_{11_M} & S_{12_M} \\ S_{21_M} & S_{22_M} \end{bmatrix} = \begin{bmatrix} S_{00} & S_{03} \\ S_{30} & S_{33} \end{bmatrix}$$

This notation is best exemplified in Figure 4b, which shows the 4-reflectometer architecture. In Figure 4a, the a_{ref} signal takes on the role of measuring a_0 or a_3 depending on whether the switch is allowing transmission on the forward or reverse path (respectively).

4.2 The 1-Port Error Model

The one-port model is a fundamental building block of the two-port models. As such, understanding the derivation of its equations will be useful in understanding the two-port models. The signal flow graph for this model is shown in Figure 5, along with the error terms accounted for in the model.

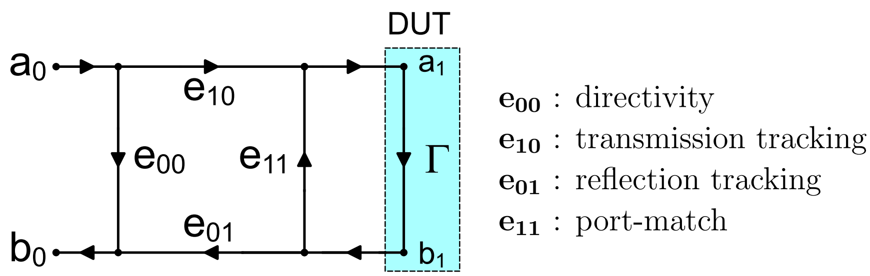


Figure 5: One-Port Error Model

4.2.1 Equations

The resulting equations for the measured and actual S-parameters (which are Γ_M and Γ respectively) from this model are as follows:

$$\Gamma_M = \frac{e_{00} - \Delta_e \Gamma}{1 - e_{11}\Gamma} \tag{2}$$

$$\Gamma = \frac{\Gamma_M - e_{00}}{\Gamma_M e_{11} - \Delta_e} \tag{3}$$

where $\Delta_e = e_{00}e_{11} - e_{10}e_{01}$.

The derivation for these equations, performed by use of the signal flow graph in Figure 5, is presented below:

$$\Gamma_M = \frac{b_0}{a_0}$$

$$\begin{aligned} b_0 &= e_{00}a_0 + e_{01}b_1 & a_1 &= e_{11}b_1 + e_{10}a_0 \\ &= e_{00}a_0 + e_{01}\Gamma a_1 & a_1(1 - e_{11}\Gamma) &= e_{10}a_0 \\ &= a_0\left(\frac{e_{00} - e_{00}e_{11}\Gamma + e_{01}e_{10}\Gamma}{1 - e_{11}\Gamma}\right) & a_1 &= a_0\frac{e_{10}}{1 - e_{11}\Gamma} \end{aligned}$$

$$\Gamma_M = \frac{e_{00} - \Delta_e\Gamma}{1 - e_{11}\Gamma} \quad (2) \quad \Rightarrow \quad \Gamma = \frac{\Gamma_M - e_{00}}{\Gamma_M e_{11} - \Delta_e} \quad (3)$$

***Notation*:**

It is most intuitive to view the 1-port error-adaptor as a two-port network whose S-parameter matrix is $E = \begin{bmatrix} e_{00} & e_{01} \\ e_{10} & e_{11} \end{bmatrix}$. Then, the determinant of the error-adaptor network is $det(E) =$

$e_{00}e_{11} - e_{10}e_{01} = \Delta_e$. For the remainder of this Application Note, the use of the Δ notation will be reserved for expressing the determinant of matrices.

4.2.2 Using the Equations

Observe that the 1-port model presented above has 4 error terms ($e_{00}, e_{01}, e_{10}, e_{11}$), but only 3 terms are needed to obtain the actual reflection coefficient, Γ, in Equation (3) (e_{00}, e_{11}, Δ_e). As we are only using this model to obtain corrected measurements (ie. obtain Γ from Γ_M), we can get away with setting up a system of 3 equations to find the 3 unknown terms. Re-writing Equation (2), we may form a system of linear equations:

$$\begin{bmatrix} 1 & \Gamma_1\Gamma_{M1} & -\Gamma_1 \\ 1 & \Gamma_2\Gamma_{M2} & -\Gamma_2 \\ 1 & \Gamma_3\Gamma_{M3} & -\Gamma_3 \end{bmatrix} \begin{bmatrix} e_{00} \\ e_{11} \\ \Delta_e \end{bmatrix} = \begin{bmatrix} \Gamma_{M1} \\ \Gamma_{M2} \\ \Gamma_{M3} \end{bmatrix} \tag{4}$$

Solving the system in *Equation* 4 only requires 3 linearly independent measurements to be made. Therefore any three components may be used as long as their actual reflection coefficients are known. The topic of *actual* S-parameters is discussed in the Application Note titled *Calibration Standards and the SOLT Method*.

4.2.3 Example: No systematic error present

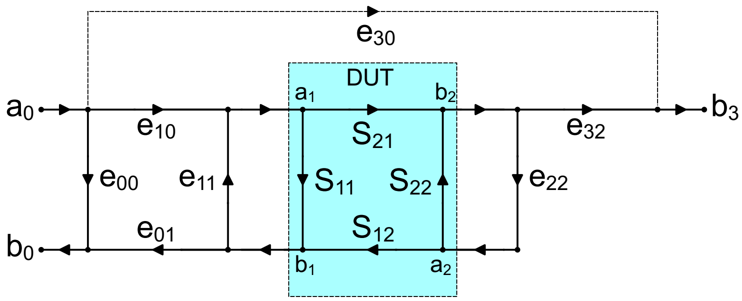
Suppose that we have a VNA which, unknown to us, experiences no error in its measurements. If we were to use the 1-port VEC model to correct our 1-port measurements, ***what should our error terms be?*** Looking at *Equation* (2) directly, you may be able to intuit an answer, however, let's reason through the signal flow graph to affirm our understanding. For no systematic measurement error to be present, we would expect that,

1. Transmission to the DUT would be undistorted ($e_{10} = 1$)
2. Reflection from the DUT would be undistorted ($e_{01} = 1$)
3. None of our transmitted signal would directly transfer into our reflected signal ($e_{00} = 0$)
4. There are no impedance mismatches between any components ($e_{11} = 0$)

Our 3 error terms are then $0, 0$, and -1 for e_{00}, e_{11}, and Δ_e respectively. Substituting this into *Equation* (2) confirms our results that $\Gamma_M = \Gamma$.

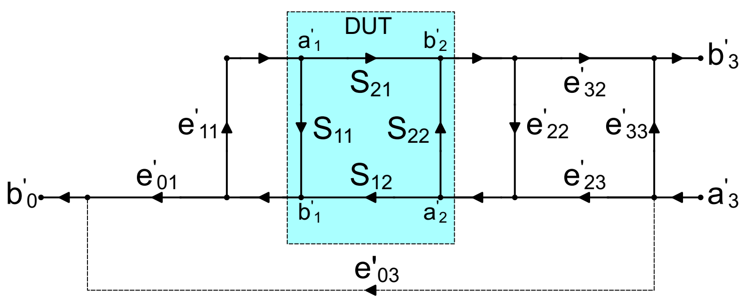
4.3 The 12-Term Error Model

The 12-term model is a 2-port VEC model that accounts for directivity, tracking, port-match, and leakage errors. It makes the fundamental assumption that the RF-source switch is imperfect, and so an entirely different set of error terms must exist for when the switch is allowing transmission in the forward versus the reverse signal path. This model may be used with either a 3-Reflectometer or 4-Reflectometer model (see Figure 4). Figure 6 shows the two signal flow graphs that represent the 12-term model: the forward path model and reverse path model.



$\mathbf{e_{00}}$: port-1 directivity
$\mathbf{e_{10}}$: port-1 trans. tracking
$\mathbf{e_{01}}$: port-1 reflect. tracking
$\mathbf{e_{11}}$: port-1 match
$\mathbf{e_{22}}$: port-2 match
$\mathbf{e_{32}}$: port-2 reflect. tracking
$\mathbf{e_{30}}$: port-1-2 leakage

(a) Forward Path Model



$\mathbf{e'_{33}}$: port-2 directivity
$\mathbf{e'_{23}}$: port-2 trans. tracking
$\mathbf{e'_{32}}$: port-2 reflect. tracking
$\mathbf{e'_{22}}$: port-2 match
$\mathbf{e'_{11}}$: port-1 match
$\mathbf{e'_{01}}$: port-1 reflect. tracking
$\mathbf{e'_{30}}$: port-2-1 leakage

(b) Reverse Path Model

Figure 6: 12-Term Error Model

The 12-term model attempts to take into account all the major types of systematic error discussed in Section 2. Note that Figure 6 shows a total of 14 terms. Similar to the one-port model being able to use 3 terms instead of 4 terms when making S-parameter measurements (meaning not recovering individual a/b waves), each of the two 7-term models presented are able to combine terms to create 6-term models, thus dubbing this VEC method the 12-term model.

4.3.1 Equations

The equations describing the measured and actual S-parameters from the 12-term model are as follows [2]:

Measured:

$$S_{11_M} = e_{00} + e_{10}e_{01}\frac{S_{11} - e_{22}\Delta_S}{1 - e_{11}S_{11} - e_{22}S_{22} + e_{11}e_{22}\Delta_S} \tag{5}$$

$$S_{21_M} = e_{30} + e_{10}e_{32}\frac{S_{21}}{1 - e_{11}S_{11} - e_{22}S_{22} + e_{11}e_{22}\Delta_S} \tag{6}$$

$$S_{12_M} = e'_{03} + e'_{23}e'_{01}\frac{S_{12}}{1 - e'_{11}S_{11} - e'_{22}S_{22} + e'_{11}e'_{22}\Delta_S} \tag{7}$$

$$S_{22_M} = e'_{33} + e'_{23}e'_{32}\frac{S_{22} - e'_{11}\Delta_S}{1 - e'_{11}S_{11} - e'_{22}S_{22} + e'_{11}e'_{22}\Delta_S} \tag{8}$$

$$\text{where } \Delta_S = det(S) = S_{11}S_{22} - S_{12}S_{21}$$

Actual:

$$S_{11} = \left[\left(\frac{S_{11_M} - e_{00}}{e_{10}e_{01}}\right)\left[1 + \left(\frac{S_{22_M} - e'_{33}}{e'_{23}e'_{32}}\right)e'_{22}\right] - e_{22}\left(\frac{S_{21_M} - e_{03}}{e_{01}e_{32}}\right)\left(\frac{S_{12_M} - e'_{03}}{e'_{23}e'_{01}}\right)\right]\frac{1}{D} \tag{9}$$

$$S_{21} = \left[\left(\frac{S_{21_M} - e_{03}}{e_{01}e_{32}}\right)\left[1 + \left(\frac{S_{22_M} - e'_{33}}{e'_{23}e'_{32}}\right)(e'_{22} - e_{22})\right]\right]\frac{1}{D} \tag{10}$$

$$S_{12} = \left[\left(\frac{S_{12_M} - e'_{03}}{e'_{23}e'_{01}}\right)\left[1 + \left(\frac{S_{11_M} - e_{00}}{e_{10}e_{01}}\right)(e_{11} - e'_{11})\right]\right]\frac{1}{D} \tag{11}$$

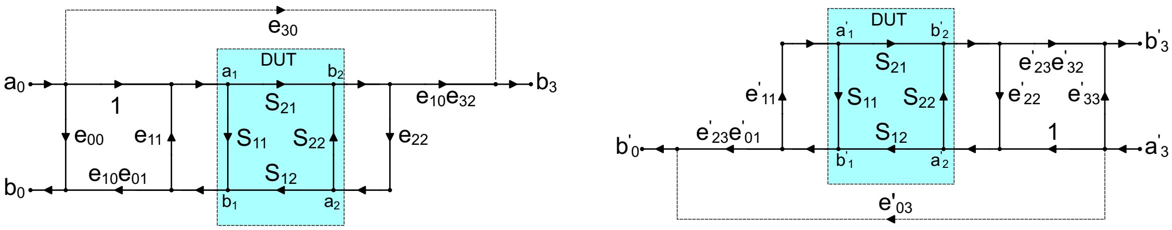
$$S_{22} = \left[\left(\frac{S_{22_M} - e'_{33}}{e'_{23}e'_{32}}\right)\left[1 + \left(\frac{S_{11_M} - e_{00}}{e_{10}e_{01}}\right)e_{11}\right] - e'_{11}\left(\frac{S_{21_M} - e_{03}}{e_{01}e_{32}}\right)\left(\frac{S_{12_M} - e'_{03}}{e'_{23}e'_{01}}\right)\right]\frac{1}{D} \tag{12}$$

$$D = \left[1 + \left(\frac{S_{11_M} - e_{00}}{e_{10}e_{01}}\right)e_{11}\right]\left[1 + \left(\frac{S_{22_M} - e'_{33}}{e'_{23}e'_{32}}\right)e'_{22}\right] - \left(\frac{S_{21_M} - e_{03}}{e_{01}e_{32}}\right)\left(\frac{S_{12_M} - e'_{03}}{e'_{23}e'_{01}}\right)e_{22}e'_{11}$$

There's no other way to put it: the equations for the 12-term model are a mess. However, deriving these equations may help you brush up on signal flow graph analysis and your algebra.

4.3.2 Using the Equations

To correct measurements using the 12-term model, we must 'calibrate the VNA' by performing a set of measurements that allow us to obtain the error terms of the model. Only then do Equations (9) - (12) become useful. First, we combine terms in our model to have only 12-terms:



(a) 6-term forward model (b) 6-term reverse model

Figure 7: Normalized 12-term model

To obtain the error terms of the forward model:

1. One-Port Cal: Obtain $e_{00}, e_{11}, e_{10}e_{01}$ as described in section 4.2 (Calculate $e_{10}e_{01}$ from Δ_e).

2. Isolation (optional): If using the leakage terms, measure matched loads at both ports simultaneously to solve for e_{30} (see *Equations* 5-8). If not, set the leakage term to zero.

3. Thru-Measurement: Connect ports 1 and 2 to obtain e_{22} and $e_{10}e_{32}$ by using *Equations* 5-8.

Repeat this process to solve for the reverse-model error terms. When using these equations, it is key to understand what your *actual* S-parameters are, be they ideal or calibration kit standard listed values. Plug these values into the equations accordingly. If you are looking at other sources for the model equations, make sure to understand what assumptions they are making about the standards used for measurement.

4.3.3 Example: Using Ideal SOLT Standards

Let's take a look at how to solve for the forward error terms when we are using ideal SOLT calibration standards, meaning $\Gamma_{open} = 1$, $\Gamma_{short} = -1$, $\Gamma_{load} = 0$, $S_{thru} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$.
For the one-port error terms, there is not much insight to be gained from using ideal standards – we must still solve the system of linear equations. However, for the isolation measurement, our calculation simplifies greatly . *Equation* 6 is reproduced below.

$$S_{21_M} = e_{30} + e_{10}e_{32}\frac{S_{21}}{1 - e_{11}S_{11} - e_{22}S_{22} + e_{11}e_{22}\Delta_S} \tag{6}$$

As we are using matched loads at each port for this measurement, our actual S-parameters are $S = \mathbf{O}$, the 2x2 zero-matrix. By substituting these parameters into the equation above we end up with the result:

$$e_{30} = S_{21_M}$$

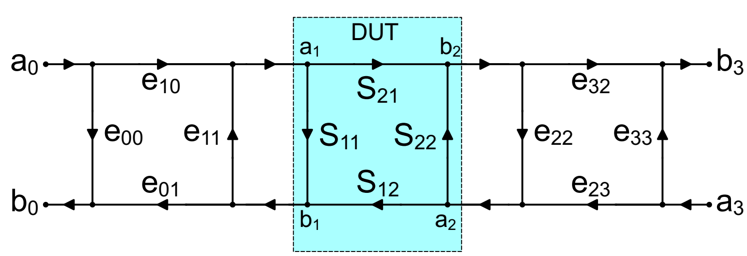
Now we substitute our ideal thru parameters into *Equation* 5, along with the thru measurement S-parameters:

$$S_{11_M} = e_{00} + e_{10}e_{01}\frac{0 - e_{22}(-1)}{1 + e_{11}e_{22}(-1)}$$
$$(S_{11_M} - e_{00})(1 - e_{11}e_{22}) = e_{10}e_{01}e_{22}$$
$$e_{22}(-e_{11}(S_{11_M} - e_{00}) - e_{10}e_{01}) = e_{00} - S_{11_M}$$
$$e_{22} = \frac{e_{00} - S_{11_M}}{-e_{11}S_{11_M} + e_{00}e_{11} - e_{10}e_{01}}$$
$$e_{22} = \frac{S_{11_M} - e_{00}}{e_{11}S_{11_M} - \Delta_e}$$

This process would be repeated to find analogous terms for the reverse model to complete the 12-term model.

4.4 The 8-Term Error Model

The 8-term model is a 2-Port VEC model that accounts for the same directivity, tracking, and port-match terms as the 12-term model, except that it lacks the 12-term model's assumption of an imperfect switch, and the leakage terms. As a result of the assumption of a perfect RF-source switch, the error terms remain unchanged whether transmitting on the forward or reverse path. This makes the 8-term model less effective than the 12-term model, however, the imperfect switch may be accounted for by other means within a 4-reflectometer architecture VNA (discussed in Section 4.5). Figure 8 shows the signal flow diagram for the 8-term model.



$\mathbf{e_{00}}$: port-1 directivity
$\mathbf{e_{10}}$: port-1 trans. tracking
$\mathbf{e_{01}}$: port-1 reflect. tracking
$\mathbf{e_{11}}$: port-1 match
$\mathbf{e_{22}}$: port-2 match
$\mathbf{e_{32}}$: port-2 reflect. tracking
$\mathbf{e_{23}}$: port-2 trans. tracking
$\mathbf{e_{23}}$: port-2 directivity

Figure 8: 8-Term Error Model

Notice that when one of the *a* waves is set to zero, the signal flow graph simplifies to one of the 12-term models (minus the prime notation and the leakage terms). This means we could potentially solve the 8-term model in the same manner as the 12-term model.

However, the true elegance of the 8-term model comes from its representation of a cascade of 3 two-port networks: an error adaptor at port-1, the DUT, and an error adaptor at port-2. Thus, we are able to use T-parameters (also known as cascade-parameters) to construct and apply vector error correction with the 8-term model.

4.4.1 Equations

Using the notation from [2], we will construct the equations for the 8-term model using T-parameters. First recall that,

$$T = \frac{1}{S_{21}} \begin{bmatrix} -\Delta_S & S_{11} \\ -S_{22} & 1 \end{bmatrix} \tag{13}$$

We now express the measured T-parameter matrix, T_M as a cascade of the DUT, T, with the error adaptors, T_X and T_Y:

$$T_M = T_X T T_Y \tag{14}$$

Where,

$$T_X = \frac{1}{e_{10}} \begin{bmatrix} -\Delta_X & e_{00} \\ -e_{11} & 1 \end{bmatrix} \qquad T_Y = \frac{1}{e_{32}} \begin{bmatrix} -\Delta_Y & e_{22} \\ -e_{33} & 1 \end{bmatrix}$$

$$\Delta_X = e_{00}e_{11} - e_{10}e_{01} \qquad \Delta_Y = e_{33}e_{22} - e_{23}e_{32}$$

And thus, if the matrices are invertible*,

$$T = T_X^{-1} T_M T_Y^{-1} \tag{15}$$

*by measurement noise, these matrices will always be invertible.

4.4.2 Using the Equations

To solve for our error terms, we play a game of combining terms from across the error adaptors. Namely, we dub $q = e_{10}e_{32}$ the transmission term for the 8-term model. Thus Equation (14) is transformed as follows:

$$\begin{aligned} T_M &= T_X T T_Y \\ T_M &= \frac{1}{e_{10}e_{32}} \begin{bmatrix} -\Delta_X & e_{00} \\ -e_{11} & 1 \end{bmatrix} T \begin{bmatrix} -\Delta_Y & e_{22} \\ -e_{33} & 1 \end{bmatrix} \\ T_M &= \left(\frac{1}{q}\right) ATB \end{aligned}$$

Using the one-port model, we can solve for the contents of the A and B matrices.With A and B considered as knowns, we can continue to solve for the transmission term, q.

$$det(T_M) = \left(\frac{1}{q^2}\right) det(ATB)$$

$$q = \pm\sqrt{\frac{det(ATB)}{det(T_M)}} \tag{16}$$

The equation for the transmission term q above leaves ambiguity for its sign. In practice, this may be determined by following the sign of the first entry of the $Q = qI$ matrix, where I is the 2x2 identity matrix. At last,

$$T = qA^{-1}T_M B^{-1}$$

Hopefully it has become obvious that the 8-term model presents a much more elegant solution to vector error correction than the 12-term model. As mentioned before, this elegance comes at the cost of making the assumption that error terms remain static upon changing the RF-source switch state. This assumption can be remedied when using a 4-Reflectometer VNA, as explained in the following section on switch correction. The assumption that there are no leakage terms is not compeletly true as these terms are present and only become significant when attempting to measure very low power signals.

4.5 Switch Correction

A 4-Reflectometer VNA presents enough information to account for the effects of an imperfect switch – its non-unity transmission from the RF-source to the signal path and its non-zero reflection of b waves. Figure 4b illustrates the a and b waves as they pass through such an architecture. We can use the fact that the reflectometers are all on the DUT side of the switch to obtain equations for the measured S-parameters of the DUT, S_M, in terms of the a/b waves from both the forward and reverse measurements as follows, where prime notation again indicates transmission on the reverse path [2]:

Forward

$$b_0 = S_{11_M} a_0 + S_{12_M} a_3$$
$$b_3 = S_{21_M} a_0 + S_{22_M} a_3$$

Reverse

$$b_0' = S_{11_M} a_0' + S_{12_M} a_3'$$
$$b_3' = S_{21_M} a_0' + S_{22_M} a_3'$$

This linear system of 4-equations and 4-unknowns may be used to solve for the measured S-parameters, resulting in the following equations [2]:

$$S_{11_M} = \left(\frac{b_0}{a_0} - \frac{b_0'}{a_3'}\frac{a_3}{a_0} \right) \frac{1}{d} \qquad S_{12_M} = \left(\frac{b_0'}{a_3'} - \frac{b_0}{a_0}\frac{a_0'}{a_3'} \right) \frac{1}{d}$$

$$S_{21_M} = \left(\frac{b_3}{a_0} - \frac{b_3'}{a_3'}\frac{a_3}{a_0} \right) \frac{1}{d} \qquad S_{22_M} = \left(\frac{b_3'}{a_3'} - \frac{b_3}{a_0}\frac{a_0'}{a_3'} \right) \frac{1}{d}$$

$$d = 1 - \frac{a_3}{a_0}\frac{a_0'}{a_3'}$$

As these switch correction equations contain a/b waves from both the forward and reverse measurements, they only apply to two-port measurements.

4.5.1 Example: Ideal Switch

An ideal switch has perfect transmission and zero reflection. Let's see how the switch correction equations change under these assumptions. In terms of a/b waves, this translates to the following: $a'_0 = 0$ and $a_3 = 0$ (i.e. there is nothing "bouncing off the switch at the opposite port"). The switch correction equations then simplify to,

$$S_{11_M} = \frac{b_0}{a_0} \qquad S_{12_M} = \frac{b'_0}{a'_3}$$

$$S_{21_M} = \frac{b_3}{a_0} \qquad S_{22_M} = \frac{b'_3}{a'_3}$$

These are the S-parameters you should be familiar with. Note that we don't have to take into account perfect transmission because it doesn't matter! Non-perfect transmission is measured by the reflectometers *after the switch* and it is the same wave that hits the DUT (or error adaptor).

4.6 Exploration: The Effects of Switch Correction & Two-Port Model Comparison

The capability of correcting for the imperfect switch in a 4-Reflectometer would theoretically make the 8-term and 12-term models equivalent, as they are both modeling for the same types of systematic error. The following measurements were performed with the UVNA-63 Kit, employing a 4-reflectometer model, with a 2.5GHz bandpass filter as the DUT (using the exact same measurement setup as Exploration 3.1). Figure 9 shows four plots obtained from the measurements of the magnitude response of the uncorrected S-parameter measurement, a 12-term correction, a 8-term correction, and an 8-term correction with switch correction.

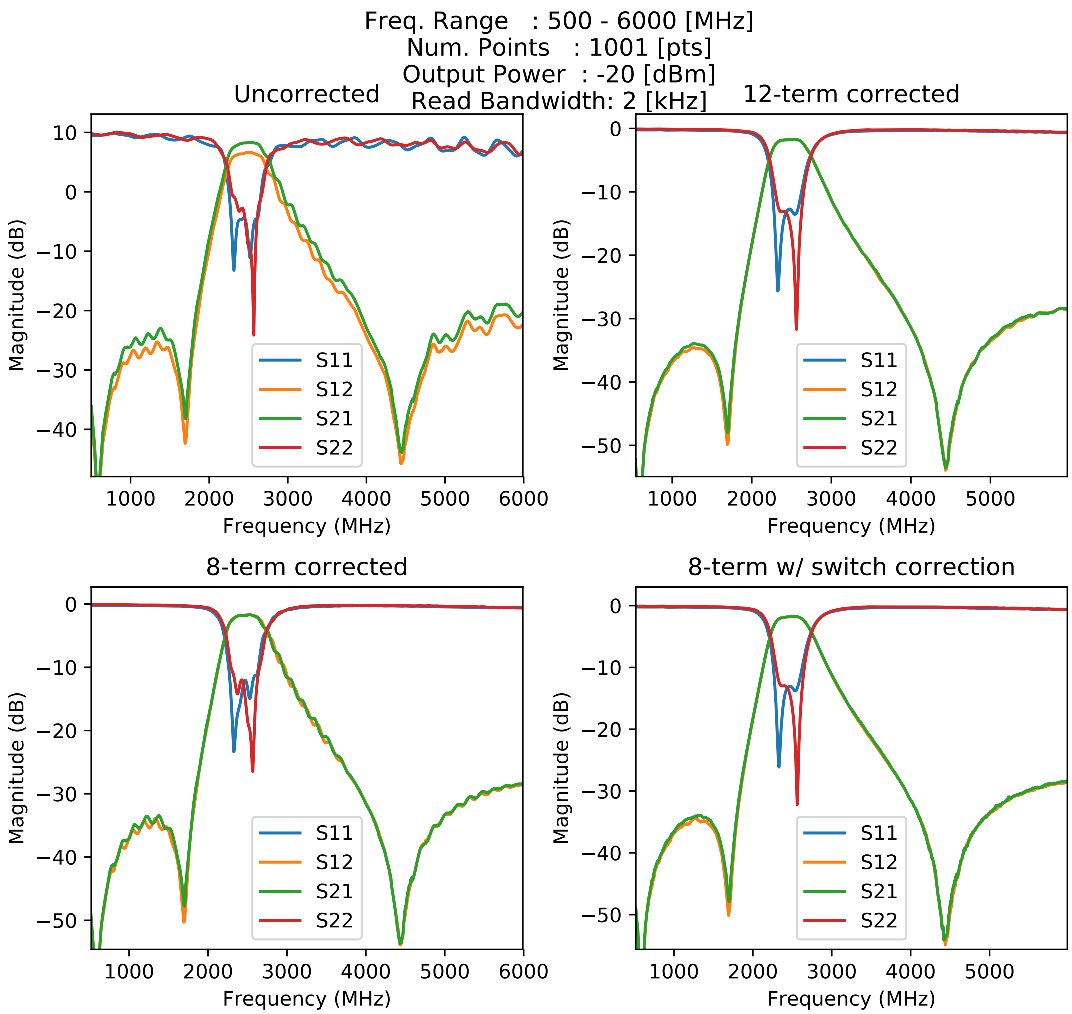


Figure 9: Measurement of 2.5 GHz bandpass filter magnitude S-parameters, with corrections from different two-port models

The first thing to notice is that all three corrections eliminate the ‘fictitious gain’ of the uncorrected measurement. Moreover, the 8-term model (with switch correction) and the 12-term model are in agreement with each other – the traces of the S-parameters are almost identical. Also note that switch correction cleans up some ripples in the transmission terms, bringing the 8-term model correction on par with the 12-term model correction. To illustrate this claim further, Figure 10 shows a comparison of the S-parameter plots in the passband of the filter from the same measurements shown in Figure 9.

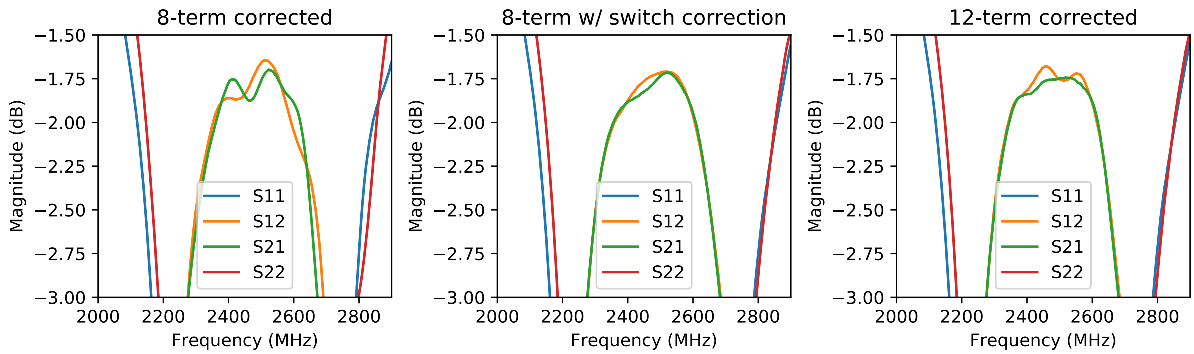


Figure 10: Closeup of the filter passbands from Figure 9, comparing the 8 and 12-term VEC models

Here we see a more detailed example of the qualitative difference between in the VEC models. We can see that the 8-term model with switch-correction provides the cleanest plot. In comparison with the frequency response correction (seen in Section 3.1), the power of vector error correction is superb.

Summary

- Systematic error is inherent to any measurement setup.
- Frequency response calibration only accounts for tracking errors.
- VEC accounts for directivity, port-match, and tracking errors.
- The methods of solving for the equations of VEC models are dependent on the number of reflectometers in the VNA architecture.
- The one-port VEC model is used in the two-port models.
- The 8-term VEC model makes the assumption of using a perfect RF-source switch, so switch correction must be employed along with it. The 12-term model makes no such assumptions.

VNA Kit Functions

For all functions, reference the VNA kit API for documentation on the format of inputs and outputs.

1. **Frequency Response Correction**:
   Using the knowledge gained from Section 3, write your own version of the following UVNA-63 kit API functions:

```
get1PortResponseModel(G,Gm)
    Returns a model for the reflection channel: Gm = H(w)G
    input:
        G: [num_pts] actual, listed, or ideal reflection coefficient
        Gm: [num_pts] measured reflection coefficient
    output:
        H: [num_pts] reflection channel

get2PortResponseModel(G1,G1m,G2,G2m,T,Tm)
    Returns a model for a 2x2 Frequency Response model (channel):
    Sijm = Hij(w)Sij
    input:
        G1,G2: [num_pts] actual, listed, or ideal reflection coefficient
            at port 1 and port 2
        Gm1,Gm2: [num_pts] measured reflection coefficient at port 1
         and port 2
        T,Tm: [num_pts,2,2] actual, listed, or ideal thru
         measurement (T) and measured thru (Tm)
    output:
        H: [num_pts,2,2] 2x2 Frequency Response model (channel)

correctResponse(Sm,H)
    Computes DUT S-parameters by inverting the channel modeled by H
    input:
        Sm: [num_pts,2,2] measured S-parameters
        H: [num_pts,2,2] Frequency Response channel model
    output:
        S: [num_pts,2,2] DUT S-parameters
```

   These functions should allow you to obtain the frequency response error model of the VNA's systemic error, and correct your raw S-parameter measurements. Test your functions by substituting these error correction methods into the `skeleton.py` example code. This will involve more than simply substituting function names, as the skeleton code functions ask for different parameters.

2. **One-Port VEC**:
   Using the knowledge gained from Section 4.2, write your own version of the following UVNA-63 kit API functions:

```
get1PortModel(G,Gm)
    returns 3 error terms from 1 port calibration
    parameters are ideal and measured reflection coefficients
    for 3 components G and Gm must have standards listed
    in the same order
    inputs:
        G: [num_pts,3] actual component reflection coefficients
        Gm: [num_pts,3] measured component reflection coefficients
    outputs:
    x: [num_pts,3,1] solution to the matrix equation Ax=b,
        where A and b are defined by the system of
        equations for the 1 port model

correct1Port(Gm,err_terms)
    corrects a reflection coefficient measurement using
    the one-port VEC terms returned from get1PortModel()
    input:
        Gm: [num_pts] measured reflection coefficient
        err_terms: [num_pts,3,1] matrix of 1-port error terms
         obtained from get1PortModel()
    output:
        G: [num_pts] corrected reflection coefficient
```

These functions should allow you to obtain the error model for one-port VEC over frequency. Example 4.2.3 should provide some guidance (be wary of numerical imprecision when interpreting your results).

3. **12-Term Model**: Using the knowledge gained from Section 4.3, write your own version of the following UVNA-63 kit API functions:

```
get12TermModel(G1,Gm1,G2,Gm2,T,Tm,isolation=None)
    Constructs the 12-Term error model
    (with optional isolation measurement),
    using measured SOLT data and listed SOLT data.
    input:
        G1,G2: [num_pts,3] listed (G) and measured (Gm)
         reflection coefficients
            of standards used on port 1 and 2 respectively.
            Standards used in G,Gm must be in same order
            (ie. G = [O,S,L] Gm = [Om,Sm,Lm])
        T,Tm: [num_pts,2,2] actual (T) and measured (Tm)
         reflection coefficients of thru standard
        isolation (optional): [num_pts,2,2] measured S-parameters
        of an isolation measurement
        ie. matched loads on each port.
```

```
    output:
        fwd_terms: [num_pts,6] [e00 ,e11 ,e10e01  ,e10e32  ,e22 ,e30]
        rev_terms: [num_pts,6] [ep33,ep22,ep23ep32,ep23ep01,ep11,ep03]

correct12Term(Sm,fwd_terms,rev_terms)
    Applies 12-Term Model error correction with terms
    obtained from get12TermModel()
    input:
        Sm: [num_pts,2,2] raw S-parameter measurements
        fwd_terms: [num_pts,6] [e00 ,e11 ,e10e01  ,e10e32  ,e22 ,e30]
        rev_terms: [num_pts,6] [ep33,ep22,ep23ep32,ep23ep01,ep11,ep03]
    output:
        S: [num_pts,2,2] 12-term corrected S-parameters
```

These functions should make use of the 1-port VEC model, obtaining the 12-term VEC model and applying it to raw S-parameter data. Note that these functions allow for optional use of isolation measurements, as often is the case with VNA's in the industry. Test your functions by substituting these error correction methods into the `skeleton.py` example code.

4. **Switch Correction**: Using the knowledge gained form Section 4.5, write your own version of the UVNA-63 API a/b wave to S-parameter function with switch correction:

```
ab2S_SwitchCorrect(rec_tx1,rec_tx2,ports)
    Converts ab wave measurements from measure2Port() to
    S parameters with switch correction
    input:
        rec_tx1,rec_tx2: {1:[num_pts] , 2:[num_pts], ..., 6:[num_pts]}
            recording dictionaries returned by measure2Port()
        ports: mapping dictionary,
            possible keys: ['Tx1','Tx2','Rx1A','Rx1B','Rx2A','Rx2B']
            possible values: [1, 2, 3, 4, 5, 6]
    output:
        S: [num_pts,2,2] stack of S parameter matrices in frequency
```

5. **8-Term Model**: Using the knowledge gained from Section 4.4, write your own version of the following UVNA-63 kit API functions:

```
get8TermModel(G1,Gm1,G2,Gm2,T,Tm):
    Constructs 8-term model based on measured and actual SOLT S-parmeters.
    input:
        G1,G2: [num_pts,3] actual (G) and measured (Gm) reflection coefficients
            of standards used on port 1 and 2 respectively. Standards used in
            G,Gm must be in same order (ie. G = [O,S,L] Gm = [Om,Sm,Lm])
        T,Tm: [num_pts,2,2] actual (T) and measured (Tm) reflection coefficients
```

```
          of thru standard
    output:
        A,B: [num_pts,2,2] normalized error adapter [-delta_e e00
                                                      e11     1 ]
          at ports 1 and 2 respectively
        q: [num_ptst,1] thru error term e10e32

correct8Term(Sm,A,B,q):
    returns DUT S parameters from 8-term error model
    input:
        Sm: [num_pts,2,2] raw measured DUT S-parmeters
        A,B,q: 8-term mdoel error terms from get8TermModel()
    output:
        S: [num_pts,2,2] corrected DUT S-params
```

These functions should make use of the 1-port VEC model, obtaining the 8-term VEC model and applying it to raw S-parameter data. Again, note that these functions allow for optional use of isolation measurements, as often is the case with VNA's in the industry. Test your functions by substituting these error correction methods into the `skeleton.py` example code. This will involve more than simply substituting function names, as the skeleton code functions ask for different parameters.

References

[1] *Applying Error Correction to Vector Network Analyzer Measurements*, Keysight Technologies, web: `http://literature.cdn.keysight.com/litweb/pdf/5965-7709E.pdf`

[2] Doug Rytting, *Network Analyzer Error Models and Calibration Methods*, Agilent Technologies, web: `https://www.rfmentor.com/sites/default/files/NA_Error_Models_and_Cal_Methods.pdf`