

cablecheck: a measurement-to-report pipeline for high-frequency automotive data cables

Design, mathematics and verification of Project A

Sreeram Anil

18 September 2026 (v0.1.1)

Contents

1 Purpose and scope	2
2 Architecture	2
3 Mathematics	3
3.1 Power waves and the scattering matrix	3
3.2 Wave-chain (T) parameters and cascading	3
3.3 Fixture removal	3
3.4 Mixed-mode conversion and the port map	5
3.5 Derived quantities	5
3.6 Impedance along the length and characteristic impedance	6
3.7 Limits, margins and verdict	7
4 The synthetic reference model	8
5 Verification	9
6 Outputs	10
7 Limitations and how the following projects build on this one	11
8 Appendix A: reference-set results	12
9 References	13

1 Purpose and scope

A cable-test laboratory produces one Touchstone file per sample from its vector network analyser (VNA). Between that file and a defensible statement of the form “this sample passes the 1000BASE-T1 link-segment limits with 1.9 dB of return-loss margin at 391 MHz” lie a dozen numerical steps that are usually spread over instrument macros, spreadsheets and a person’s memory. `cablecheck` collects those steps into one tested Python package:

1. read the raw file (Touchstone v1/v2, any port count, or a VNA CSV dump);
2. remove the test fixture so that only the cable is judged (three methods, of increasing assumption strength);
3. convert the single-ended measurement into mixed-mode (differential/common) quantities using an explicit *port map*;
4. compute the quantities a cable specification talks about: insertion loss, return loss, mode conversion (LCL/LCTL), pair-to-pair crosstalk (NEXT/FEXT), propagation delay and skew, velocity ratio, impedance profile along the length, fitted characteristic impedance;
5. compare each quantity against a limit line stored as data (TOML), with the margin sign convention fixed so that “positive is good” for everything;
6. emit the verdict, the worst margin and where it occurs, a self-contained HTML report, an optional PDF, a JSON record, and a row set in a SQLite results database with the input file hashes, the limit-file identity and the software version;
7. do all of that from a command line, from a batch job file, or from a small Tk interface for lab staff.

The package has no dependency beyond NumPy, SciPy and Matplotlib. Every formula in this report corresponds to a named function in the source, and the test-suite re-runs a set of *synthetic reference measurements* whose physics is known in closed form, so that any code change that moves a number is caught.

Two design decisions deserve to be stated up front. First, the limit lines shipped with the package are transcribed from *public* OPEN Alliance and IEEE task-force material and are flagged transcribed-unverified; the format is the deliverable, the lab’s controlled copy of each standard is the source of truth and must be entered by the lab. Second, the reference data are synthetic. No real measurement of any manufacturer’s cable is included; the models are built from multiconductor transmission-line theory with parameters chosen to be representative, not to imitate any product.

2 Architecture

```
src/cablecheck/
io/touchstone.py  Touchstone reader/writer (v1.x, v2.0 subset)  io/csvvna.py  VNA CSV reader
network.py        Network container; S<->Z, renormalisation, wave-chain T, ABCD->S
mixedmode.py      PortMap, single-ended -> mixed-mode conversion
deembed.py        explicit fixture de-embedding, 2x-thru bisection, port extension
tdr.py            frequency->time transform, DC extrapolation, windows, gating
quantities.py     IL, RL, LCL, LCTL, NEXT, FEXT, delays, NVP, impedance profile, fitted impedance
limits/           limit-line library (TOML) and safe expression evaluator
evaluate.py       margins, verdict, headline
pipeline.py       files in -> RunResult out (with hashes and context)
report/           plots, HTML, PDF
cli.py            command line
synth.py          MTL-based synthetic cables/fixtures  reference.py  reference data set
db.py             SQLite results database
gui.py            Tk interface
```

The data flow is linear: `read_touchstone` → `deembed` → `to_mixed_mode` → `compute_quantities` → `evaluate` → `report/DB`. Each stage consumes and returns plain dataclasses (`Network`, `Mixed-ModeNetwork`, `Trace`, `QuantityResult`, `Evaluation`, `RunResult`) so that any stage can be replaced

or tested alone. The later projects of this series plug in at well-defined points: the impedance-profile algorithm (Project D) replaces `quantities.impedance_profile`; the measurement-automation project (B) produces the `SampleInfo` context and calls `pipeline.run_sample`; the analytics (E) and cross-site (F) projects read the results and runs tables.

3 Mathematics

3.1 Power waves and the scattering matrix

Let port i have a real reference impedance $Z_{0i} > 0$, terminal voltage V_i and current I_i flowing into the network. The power waves (Kurokawa (Kurokawa 1965)) are

$$a_i = \frac{V_i + Z_{0i}I_i}{2\sqrt{Z_{0i}}}, \quad b_i = \frac{V_i - Z_{0i}I_i}{2\sqrt{Z_{0i}}},$$

and the scattering matrix $S(f)$ is defined by $\mathbf{b} = \mathbf{S}\mathbf{a}$. The package stores $\mathbf{S}$ as an array of shape (n_f, n, n) with S_{ij} the response at port i to a unit incident wave at port j . With $\mathbf{G} = \text{diag}(\sqrt{Z_{0i}})$ the impedance matrix and its inverse relation are

$$\mathbf{Z} = \mathbf{G}(\mathbf{I} + \mathbf{S})(\mathbf{I} - \mathbf{S})^{-1}\mathbf{G}, \quad \mathbf{S} = (\mathbf{z} - \mathbf{I})(\mathbf{z} + \mathbf{I})^{-1}, \quad \mathbf{z} = \mathbf{G}^{-1}\mathbf{Z}\mathbf{G}^{-1}.$$

Renormalisation from one set of real reference impedances to another is done through $\mathbf{Z}$ (`network.renormalize`); this is the route by which a 50 Ω single-ended measurement is compared with a 100 Ω differential limit without ever forming the ill-conditioned direct formula. The identity $\mathbf{S} \rightarrow \mathbf{Z} \rightarrow \mathbf{S}$ and the fact that a 75 Ω line renormalised to 75 Ω becomes a pure delay ($|S_{11}| = 0, |S_{21}| = 1$) are unit tests.

3.2 Wave-chain (T) parameters and cascading

For a $2n$ -port whose ports are split into a *left* group L (ports $1..n$) and a *right* group R (ports $n+1..2n$), partition $\mathbf{S}$ into $n \times n$ blocks $S_{LL}, S_{LR}, S_{RL}, S_{RR}$. The block wave-chain matrix $\mathbf{T}$ relates the waves on the left to those on the right,

$$\begin{bmatrix} \mathbf{b}_L \\ \mathbf{a}_L \end{bmatrix} = \mathbf{T} \begin{bmatrix} \mathbf{a}_R \\ \mathbf{b}_R \end{bmatrix}, \quad \mathbf{T} = \begin{bmatrix} S_{LR} - S_{LL}S_{RL}^{-1}S_{RR} & S_{LL}S_{RL}^{-1} \\ -S_{RL}^{-1}S_{RR} & S_{RL}^{-1} \end{bmatrix},$$

with the inverse map $S_{LL} = \mathbf{T}_{12}\mathbf{T}_{22}^{-1}$, $S_{LR} = \mathbf{T}_{11} - \mathbf{T}_{12}\mathbf{T}_{22}^{-1}\mathbf{T}_{21}$, $S_{RL} = \mathbf{T}_{22}^{-1}$, $S_{RR} = -\mathbf{T}_{22}^{-1}\mathbf{T}_{21}$ (Frickey 1994). Two networks connected right-group-to-left-group have $\mathbf{T} = \mathbf{T}_A\mathbf{T}_B$. The test-suite verifies that two lossless line sections of 1 m and 2.5 m cascade to the analytic 3.5 m section to 10^{-9} .

3.3 Fixture removal

3.3.1 Explicit de-embedding

A measured cable sits between a left fixture L (VNA cable, launch board, connector) and a right fixture R :

$$\mathbf{T}_{\text{meas}} = \mathbf{T}_L \mathbf{T}_{\text{DUT}} \mathbf{T}_{R'} \implies \mathbf{T}_{\text{DUT}} = \mathbf{T}_L^{-1} \mathbf{T}_{\text{meas}} \mathbf{T}_{R'}^{-1},$$

where R' is the right fixture *viewed from the DUT side* (its port groups swapped, `network.reverse_ports`). Fixture files are ordinary Touchstone networks with the convention that

ports 1.. n are the VNA side and $n+1$.. $2n$ the DUT side; they are interpolated onto the measurement grid and renormalised to its reference impedances if necessary. The operation is exact: the test `test_embed_then_deembed_is_identity` embeds a synthetic pair between two synthetic fixtures and recovers it to 10^{-10} , and the regression test `test_deembedding_recovers_truth` recovers the bare-DUT insertion loss from the noisy reference “measurement” to better than 0.02 dB and the return loss to 0.5 dB wherever the reflection is above the noise floor.

3.3.2 2x-thru bisection

When only a *fixture-fixture* (2x-thru) measurement exists, the halves must be inferred. Two estimators are implemented and selected automatically from the product of the thru delay T (from the slope of the unwrapped S_{21} phase) and the sweep bandwidth B .

Gated estimator ($TB \geq 4$, the two halves resolvable in time). Under the mirror-symmetry assumption $B = \text{reverse}(A)$, the 2-port cascade formulas give

$$S_{11}^{2x} = S_{11a} + \frac{S_{21a}^2 S_{22a}}{1 - S_{22a}^2}, \quad S_{21}^{2x} = \frac{S_{21a}^2}{1 - S_{22a}^2}.$$

S_{11a} is obtained by time-gating S_{11}^{2x} to the first fraction of the thru delay (Section 3.7); the two equations are then solved *in closed form*, $S_{22a} = (S_{11}^{2x} - S_{11a})/S_{21}^{2x}$ and $S_{21a} = \sqrt{S_{21}^{2x}(1 - S_{22a}^2)}$, with the square-root branch fixed by continuity of the unwrapped phase. The estimator is exact for mirror-symmetric halves once the gate isolates the launch. Its accuracy is limited by how well the gate separates the launch from the mid-point: on the synthetic fixture with 20 GHz bandwidth the error in S_{11a} is 0.006 at 1 GHz and 0.03 at 4 GHz where $|S_{11a}| \leq 0.08$, and it degrades once the fixture becomes strongly reflective (error 0.1 at 8 GHz, $|S_{11a}| = 0.19$) because internal multiple reflections then straddle the gate. It is therefore documented as valid for $|S_{11,\text{fixture}}| \lesssim 0.1$.

Symmetric estimator (electrically short fixture, the typical 1-600 MHz automotive case where a 30 mm launch is far below the 1.7 ns time resolution). If each half is additionally assumed symmetric in itself, $S_{11a} = S_{22a}$, then $\text{reverse}(A) = A$, $T_{2x} = T_a^2$ and $T_a = \sqrt{T_{2x}}$. The matrix square root is taken by eigendecomposition where the eigenvalues are separated and by the Schur method (`scipy.linalg.sqrtm`) where they are nearly degenerate (low frequency, $T \approx I$); among the four sign combinations the one continuous with the previous frequency point is kept, and at the first point the one with $\text{Re } S_{21a} > 0$. The error of this estimator equals the half’s own asymmetry: on the synthetic fixture (launch line plus a lumped connector at the DUT end, asymmetry 0.005 at 600 MHz) the error is 0.003, and on a fixture without the lumped element it is exact to 10^{-6} (`test_symmetric_bisection_is_exact_for_symmetric_half`).

For balanced-pair fixtures both estimators are applied per mode after mixed-mode conversion (differential and common separately); mode conversion inside the fixture is neglected, which is stated in the fixture’s provenance comment that ends up in the report.

3.3.3 Port extension

The weakest correction removes a per-port electrical delay τ_i and, optionally, a $\sqrt{f}$ loss ℓ_i (dB at a reference frequency):

$$S'_{ij} = S_{ij} \mathcal{L}_i \mathcal{L}_j e^{+j\omega(\tau_i + \tau_j)}, \quad \mathcal{L}_i = 10^{\ell_i \sqrt{f/f_{\text{ref}}}/20}.$$

3.4 Mixed-mode conversion and the port map

A physical pair has a “+” and a “−” wire at each end. The *port map* is an ordered list, one entry per VNA port, of the form A+near, A−near, A+far, A−far (pair, polarity, end); unpaired ports are written se:near. For a pair whose wires are on single-ended ports p and q with equal Z_0 , the differential and common power waves are (Bockelman and Eisenstadt 1995)

$$a_d = \frac{a_p - a_q}{\sqrt{2}}, \quad a_c = \frac{a_p + a_q}{\sqrt{2}},$$

and likewise for b . Stacking one differential row and one common row per (pair, end) into the real orthogonal matrix M gives

$$S_{\text{mm}} = M S M^T, \quad Z_d = 2Z_0, \quad Z_c = Z_0/2,$$

which for $Z_0 = 50 \Omega$ is exactly the 100 Ω differential / 25 Ω common-mode reference the automotive standards use. The implementation handles any number of pairs (an 8-port file with two pairs, or a 4-port file whose ports belong to two different pairs for a near-end crosstalk measurement) and addresses mixed-mode parameters by labels such as $S(("d", "A", "far"), ("d", "A", "near")) = S_{dd21}$ of pair A. A perfectly balanced synthetic pair yields $|S_{cd}| < 10^{-9}$, an asymmetric one does not, and the transform inverts to 10^{-12} (tests in `test_mixedmode.py`).

3.5 Derived quantities

All loss-type quantities are reported as positive decibels so that “higher is better” for everything except insertion loss and delay:

quantity	definition	limit type
insertion loss	$IL = -20 \log_{10} S_{dd21} $	maximum
return loss	$RL = \min_{\text{ends}} (-20 \log_{10} S_{dd,ii})$	minimum
LCL (mode conversion, near end)	worse of $-20 \log_{10} S_{cd,ii} $ and $-20 \log_{10} S_{dc,ii} $, both ends	minimum
LCTL (mode conversion, transfer)	worse of $-20 \log_{10} S_{cd,ji} $ and $-20 \log_{10} S_{dc,ji} $, both directions	minimum
NEXT $X \rightarrow Y$	$-20 \log_{10} S_{dd}(Y_{\text{near}} \leftarrow X_{\text{near}}) $, worse direction	minimum
FEXT $X \rightarrow Y$	$-20 \log_{10} S_{dd}(Y_{\text{far}} \leftarrow X_{\text{near}}) $	minimum
phase delay	$\tau_p(f) = -\varphi_{\text{abs}}(f)/(2\pi f)$	maximum
group delay	$\tau_g(f) = -d\varphi/d\omega$	informational
delay skew	$\max_{\text{pairs}} \tau_p - \min_{\text{pairs}} \tau_p$	maximum
NVP	$\ell/(c_0 \tau_p)$ at the top of the band	informational

For reciprocal networks $S_{cd,ii} = S_{dc,ii}$, so LCL and TCL coincide in theory; both are computed and the worse is kept because measurement noise differs.

Absolute phase. The phase delay needs φ_{abs} , whereas the VNA delivers the phase modulo 2π . The phase is unwrapped along frequency (the grid is checked to be fine enough, $|\Delta\varphi| < 0.9\pi$, otherwise an explicit error asks for more sweep points), the low-frequency group delay $\tau_{g,0}$ is estimated from the slope over the lowest 10 % of the band, and the integer number of turns at the first point is fixed by requiring $\varphi_{\text{abs}}(f_0) \approx -2\pi f_0 \tau_{g,0}$:

$$k = \text{round}\left(\frac{\varphi_{\text{unw}}(f_0) + 2\pi f_0 \tau_{g,0}}{2\pi}\right), \quad \varphi_{\text{abs}} = \varphi_{\text{unw}} - 2\pi k.$$

This makes the result independent of whether $f_0 \tau$ exceeds one half turn (a 40 m cable at 1 MHz start frequency is a test case).

3.6 Impedance along the length and characteristic impedance

3.6.1 Time-domain transform

The reflection $S_{dd11}(f)$ measured on a uniform grid of step Δf from $f_{\min}$ to $f_{\max}$ is turned into a step response as follows (tdr.to_time_domain); the notation is that of the standard VNA time-domain option (Keysight Technologies 2012), but every step is explicit in the code.

1. *DC extrapolation.* A VNA never measures $f = 0$, but the step response needs $S(0)$. Re S is extrapolated linearly from the first five points, clipped to $[-1, 1]$, Im S is taken linearly to zero ($S(0)$ of a passive reciprocal network is real), and the gap $0 < f < f_{\min}$ is filled on the same grid step.
2. *Window.* The one-sided spectrum is multiplied by the right half of a Kaiser window of parameter β (default 6, side-lobe level about -44 dB (Harris 1978)), so that DC is untouched and the band edge is tapered.
3. *Rise-time filter.* Specifications quote impedance “at 500 ps rise time”. A Gaussian low-pass $H(f) = \exp(-\ln 2 (f/f_{3\text{dB}})^2)$ with $f_{3\text{dB}} = 0.339/t_r$ gives the step exactly that 10-90 % rise time.
4. *Inverse real FFT* with zero padding to N_{fft} points gives the real impulse response $h[n]$ at $t_n = n/(N_{\text{fft}}\Delta f)$; the step response is the running sum $r[n] = \sum_{m \leq n} h[m]$, which converges to $S(0)$ (property of the DFT: $\sum_n h[n] = X[0]$).
5. *Impedance and distance.* $Z(t) = Z_{\text{ref}} \frac{1+r(t)}{1-r(t)}$ and $x = v t/2$ with $v = \text{NVP} \cdot c_0$, where the NVP is either given or estimated from the measured phase delay and the stated length.

The test `test_impedance_step_is_located_and_valued` builds a $100 \Omega / 115 \Omega$ step at 2 m from the analytic line model and checks that the profile reads 100 and 115 Ω on either side and places the step within 0.1 m.

3.6.2 Time gating

The same machinery gates a reflection to a time interval $[t_1, t_2]$: the impulse response is multiplied by a gate with raised-cosine edges of width two resolution cells, transformed back with an rFFT, divided by the window where the window exceeds 0.05, and resampled on the original grid. One subtlety cost a test failure during development and is worth recording: the impulse response is periodic with period $1/\Delta f$, so the windowed impulse at $t = 0$ spreads into *negative* time, which wraps to the end of the array; the gate therefore operates on a signed time axis $t \mapsto t - 1/\Delta f$ for $t > 1/(2\Delta f)$. With this in place the gate recovers the reflection of the first of three cascaded sections to better than 0.03 up to two thirds of the bandwidth.

3.6.3 Two characteristic-impedance readings

The TDR trace of a real, lossy cable is *not* flat even when the cable is perfectly uniform: series resistance $R(f) \propto \sqrt{f}$ makes the input impedance rise with distance (Figure 1 shows about 1 Ω/m for the synthetic 15 m pair). A statistic over the whole length would therefore fail a perfect cable. Two readings are reported instead.

TDR window statistics (impedance_mean/min/max): mean, minimum and maximum of $Z(x)$ over a near-end window $[x_{\text{mask}}, x_{\text{mask}} + w]$ that starts after the connector mask (default 0.5 m) and is 3 m long by default; both lengths live in the cable-type file. This is what a cable tester shows and what catches a local defect or a ripple close to the launch.

Fitted impedance (impedance_fitted), following the “fitted impedance” method of IEC 61156-1 (IEC 2009): the magnitude of the input impedance

$$|Z_{\text{in}}(f)| = \left| Z_{\text{ref}} \frac{1 + S_{dd11}}{1 - S_{dd11}} \right|$$

oscillates around the characteristic impedance with a period $v/(2\ell)$ in frequency (6.8 MHz for 15 m). A moving average over 30 MHz removes the oscillation, the smoothed curve is fitted by least squares to $a + b/\sqrt{f}$ (the form the standard uses, which absorbs the skin-effect rise at low frequency), and the band mean of the fit is reported. On synthetic pairs of 95, 100 and 108 Ω the method returns the nominal value within 0.3 Ω regardless of loss, and on the 15 m sample with 2.5 % periodic ripple and a local defect it reads 99.3 Ω (test_fitted_impedance_recovers_nominal). The loss-compensated *profile* itself is the subject of Project D.

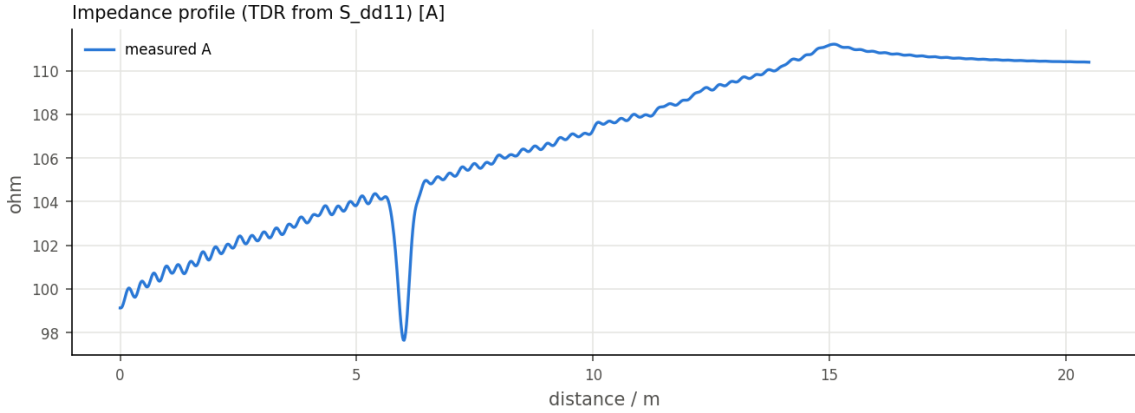


Figure 1: Impedance profile of the synthetic ripple_15m sample: 0.26 m periodic ripple, a local capacitance defect at 6 m and the lossy-line slope that motivates Project D.

3.7 Limits, margins and verdict

A *cable type* is a TOML file. Each `[[limit]]` names a quantity, a kind (max, min or range), a unit, optional per_metre and scalar flags and a list of frequency segments, each either a closed-form expression in f (MHz) or a table interpolated linearly in $\log_{10} f$. Expressions are evaluated by a small AST walker that allows arithmetic, f , length and $\text{sqrt}/\log_{10}/\log/\exp/\text{min}/\text{max}$ only, so a limit file can never execute code (test_safe_eval_whitelist). The files carry standard, clause, status and provenance fields that are printed in every report.

The shipped 1000BASE-T1 link-segment file uses the canonical insertion-loss form of every twisted-pair standard,

$$IL_{\text{lim}}(f) = 0.0023 f + 0.5907\sqrt{f} + 0.0639/\sqrt{f} \quad \text{dB}, \quad 1 \leq f \leq 600 \text{ MHz},$$

(coefficients as in the public OPEN Alliance TC9 channel document (OPEN Alliance TC9 2020) and IEEE 802.3bp (IEEE 2016)), the piecewise return-loss and $50 + 10\log_{10}(f/80) / 72 - 11.51\log_{10} f$ mode-conversion lines, a 94 ns delay bound and a $100\Omega \pm 5\%$ impedance window. A per-metre cable-component variant scales the link-segment budget by length/15, and a 100BASE-T1 Channel-Type-2 file demonstrates tabulated limits. All three are labelled transcribed-unverified on purpose.

For a trace $y(f)$ with limit $\lambda(f)$ the margin is

$$m(f) = \begin{cases} \lambda(f) - y(f) & \text{kind} = \text{max} \\ y(f) - \lambda(f) & \text{kind} = \text{min} \\ \min(y - \lambda_{\text{lo}}, \lambda_{\text{hi}} - y) & \text{kind} = \text{range} \end{cases}$$

evaluated on the *measured* frequency points inside the limit's span (limits are interpolated onto the measurement, never the reverse, so no measured point is discarded). A quantity passes when $\min_f m(f) \geq 0$; the report carries the worst margin, the frequency (or distance) where it occurs, the measured value and the limit there, and the number of failing points. The sample passes when every limited quantity passes; the *headline* is the quantity with the smallest margin. Quantities without a limit are still computed and reported as informational, and limits without a measured quantity (e.g. NEXT when only one pair was measured) raise a warning rather than silently passing.

4 The synthetic reference model

Because no real files can be shipped, the test data are generated from multiconductor transmission-line (MTL) theory (Paul 2008). A pair is two conductors above the shield with per-unit-length matrices

$$Z(f) = R(f) + j\omega L, \quad Y(f) = G(f) + j\omega C,$$

$$R(f) = \kappa \operatorname{diag}(R_{\text{dc}} + R_s \sqrt{f}), \quad R_s = \frac{\sqrt{\mu_0 \rho / \pi}}{d}, \quad G = \omega \tan \delta C,$$

with d the conductor diameter, ρ the copper resistivity, κ a proximity factor and $\tan \delta$ the dielectric loss tangent. L and C are built from the odd- and even-mode impedances and velocities so that the differential impedance $Z_d = 2Z_{\text{odd}}$ and the common-mode impedance $Z_c = Z_{\text{even}}/2$ are design inputs:

$$L_{\text{odd}} = \frac{Z_{\text{odd}}}{v_{\text{odd}}}, \quad C_{\text{odd}} = \frac{1}{Z_{\text{odd}} v_{\text{odd}}}, \quad L = \begin{bmatrix} L_s & L_m \\ L_m & L_s \end{bmatrix}, \quad C = \begin{bmatrix} C_s & -C_m \\ -C_m & C_s \end{bmatrix}, \quad L_s = \frac{L_{\text{odd}} + L_{\text{ev}}}{2}, \quad L_m = \frac{L_{\text{ev}} - L_{\text{odd}}}{2},$$

and similarly $C_s = (C_{\text{odd}} + C_{\text{ev}})/2$, $C_m = (C_{\text{odd}} - C_{\text{ev}})/2$. Mode conversion is introduced by scaling the second conductor's self-capacitance and resistance; longitudinal non-uniformity by cascading up to 240 short uniform segments whose capacitance is scaled by $1 + \delta(z)$ with δ a sinusoid (capstan or extruder-screw signature), a Gaussian bump (local defect) and seeded random roughness. Each segment's chain matrix

$$\Phi_k = \exp\left(\ell_k \begin{bmatrix} 0 & Z \\ Y & 0 \end{bmatrix}\right), \quad \begin{bmatrix} V_L \\ I_L \end{bmatrix} = \Phi \begin{bmatrix} V_R \\ I_R \end{bmatrix}$$

is evaluated by eigendecomposition (vectorised over frequency), the cable's chain matrix is the ordered product, and the S-matrix follows by solving the port boundary problem directly for $2n$ unit incident waves rather than by a block formula (`network.abcd_to_s`); the lossless case is checked against the closed-form line solution to 10^{-14} . Two coupled pairs use the 4×4 version with inter-pair mutual terms that are deliberately unequal for the two wires of a pair, which is what produces differential crosstalk in a real bundle. The fixture is a 30 mm, 52 Ω launch line per wire followed by 0.6 nH series and 0.25 pF shunt connector parasitics, and a measurement is the DUT embedded between two such fixtures plus -85 dB complex Gaussian noise with a fixed seed.

The reference set contains five samples on 801 points from 1 to 600 MHz, each with its bare-DUT “truth” file:

sample	design intent	designed outcome
good_15m	clean 15 m pair	PASS (IL margin 0.30 dB at 1 MHz)
lossy_15m	thin conductor, lossy dielectric	IL fails, -4.31 dB at 600 MHz
ripple_15m	2.5 % capacitance ripple, period 0.26 m, defect at 6 m	RL fails, -1.91 dB at 391 MHz = $\nu/(2p)$
unbalanced_10m	4 % capacitance and 10 % resistance asymmetry	LCL/LCTL fail
twopair_5m	two coupled pairs, 8-port	NEXT/FEXT exercised (FEXT fails)

The return-loss failure of ripple_15m at $f = \nu/(2p) = 0.68 c_0/(2 \cdot 0.26 \text{ m}) = 392$ MHz is the Bragg resonance of the periodic structure — precisely the signature that Project E will use to trace a defect back to a machine.

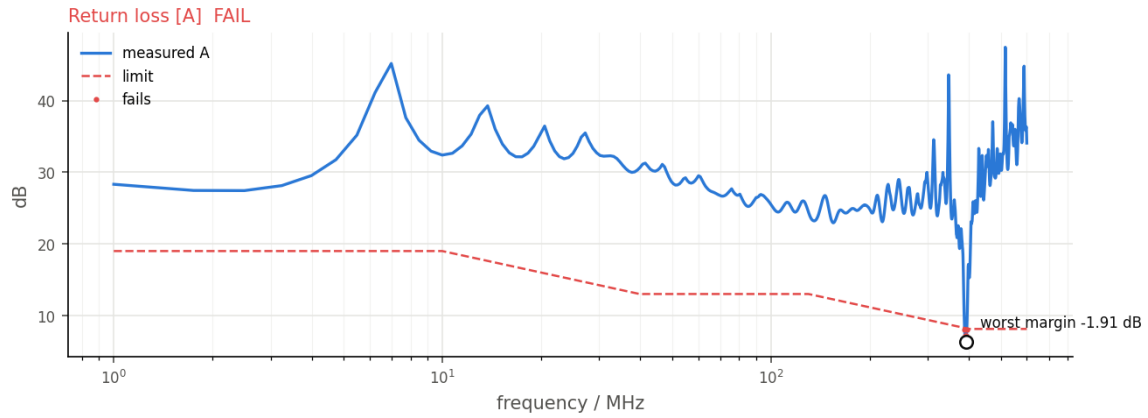


Figure 2: Return loss of ripple_15m against the 1000BASE-T1 link-segment limit, with the Bragg resonance of the 0.26 m periodicity.

5 Verification

The suite (pytest, 59 tests, about 40 s) has three layers.

Unit tests of the mathematics against closed-form solutions: analytic lossless line versus abcd_to_s; $S \leftrightarrow Z$ round trips; renormalisation to the line impedance yielding a pure delay; T-cascade of line sections; Touchstone write/read round trips for 1–6 ports in RI/MA/DB formats including the v1 two-port ordering quirk, v2 lower-triangular files and trailing noise blocks; mixed-mode conversion of balanced and unbalanced pairs and its inverse; exactness of explicit de-embedding; exactness of the symmetric bisection for a symmetric half and its error bound for an asymmetric one; gated bisection when resolved; impedance step location and value; gate isolation; phase-delay unwrapping for a 40 m cable; fitted impedance for 95/100/108 Ω pairs; crosstalk quantities of the coupled-pair model; the expression-evaluator whitelist; margin sign conventions for each limit kind.

Golden-file regression (test_regression.py): every reference sample runs through the complete pipeline with explicit fixture de-embedding, and the verdict, the pass/fail state of every quantity,

its worst margin, its location, the value and limit there and the failing-point count are compared with tests/reference/expected.json to 10^{-3} dB. The expectation is regenerated only on purpose (CABLECHECK_UPDATE_EXPECTED=1) and the resulting diff is what a reviewer reads. A second test checks that the 2x-thru path agrees with the explicit-fixture path to 0.05 dB on the good sample, and a third that de-embedding the noisy measurement recovers the truth file.

End-to-end: the CLI is driven through main() on the lossy sample and must return exit code 1 with a FAIL and an insertion_loss line, write a JSON, populate the database, list it, print a limit set and bisect the 2x-thru; the batch runner processes a job file with a 4-port and an 8-port sample; HTML, PDF and JSON writers and every table of the database are exercised. The Tk interface was smoke-tested under a virtual display by scripting the widgets (Figure 3).

The full result table of the reference set, as produced by the pipeline at the time of writing, is reproduced in Appendix A.

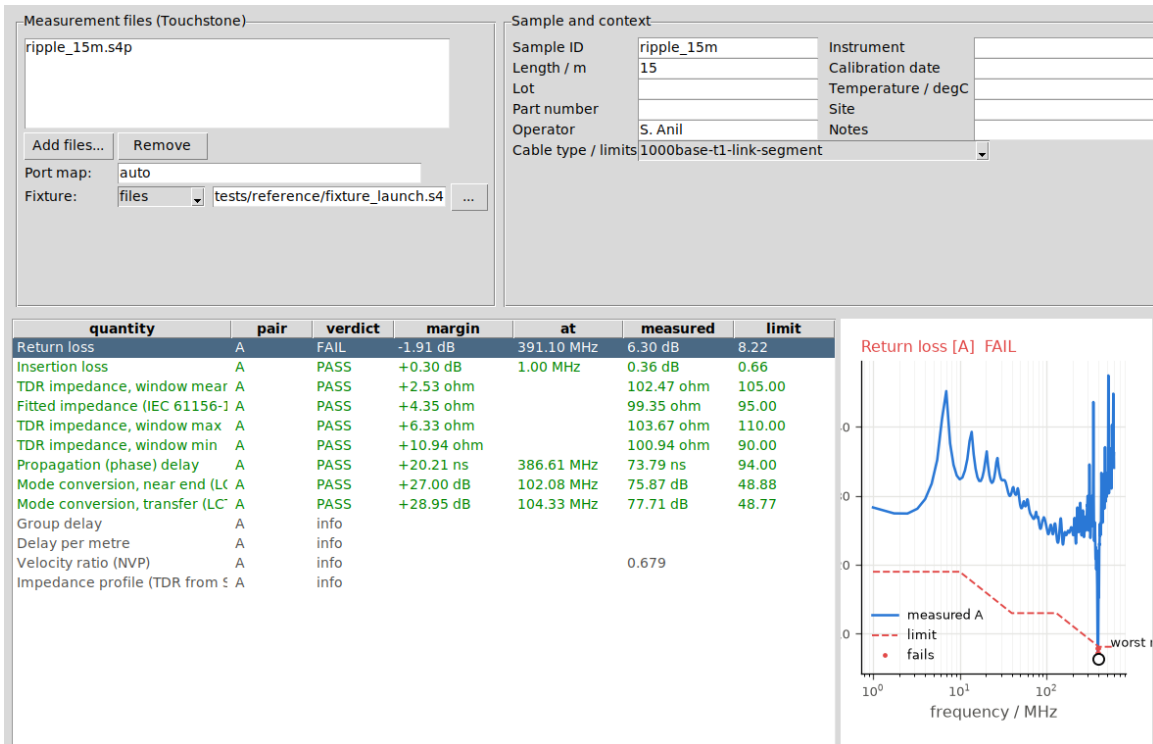


Figure 3: The lab interface after evaluating ripple_15m with the fixture removed.

6 Outputs

HTML report. Self-contained (inline SVG figures, no external assets), with the verdict banner, the headline sentence, the sample and measurement context, the source-file table with SHA-256 hashes, warnings, the results table, a margin bar chart and one figure per curve with the limit and the worst point marked.

PDF report. The same content rendered with Matplotlib's PDF backend so that no LaTeX or browser is needed on the lab PC.

JSON record. Everything in the report plus the traces themselves, for downstream analytics.

Results database. SQLite with four tables: samples (identity and build data), runs (who, what, when, temperature, instrument, calibration date, limit-file identity and status, software version, verdict, headline, report path, and the full JSON blob), files (name, path, SHA-256, port count,

span, port map, fixture method) and results (one row per quantity and pair with its worst margin). Everything an auditor needs to reproduce a verdict later is in these tables. `cablecheck db list/show/export` query it.

Command line. `cablecheck evaluate FILE... --cable-type ID --sample-id S --length L [--fixture-left F] [--thru T] [--port-map ...] [--pdf --json --db results.sqlite], cablecheck batch job.json, cablecheck limits list/show, cablecheck fixture thru.s4p -o half.s4p, cablecheck db ..., cablecheck synth --out DIR, cablecheck gui. The exit code is 0 for PASS and 1 for FAIL so that the command composes with shell scripts and the automation of Project B.`

Lab interface. File list, port map, fixture choice, sample and context fields, cable-type selector, a verdict banner with the headline, a colour-coded results table and the plot of the selected quantity; buttons save the HTML/PDF/JSON and append to a database. Evaluation runs in a worker thread so the window stays responsive.

7 Limitations and how the following projects build on this one

- The limit files are transcribed from public material and marked as such; the lab must load its controlled limits. The format was designed so that this is a data entry task, not a code change.
- The 2x-thru bisection is exact only under mirror symmetry (gated) or full symmetry (symmetric estimator); a lumped connector at the DUT end of a launch violates the latter by its asymmetry, which is the residual error. IEEE P370-class methods (impedance-corrected bisection) are the natural upgrade and belong to Project D's time-domain work.
- The TDR profile is the plain reflection-coefficient profile; the rise along a lossy line and the connector masking are handled by window and statistics, not by loss compensation. Project D delivers the loss-compensated, layer-peeled profile and quantifies how much each transform choice moves the reading; with that package installed, `--profile-engine zprofile` routes the impedance quantities through it. (Version 0.1.1 also fixed the start of the step-response running sum — half of an edge at the reference plane sits at negative time and wraps to the end of the DFT array — which moved the TDR window statistics by up to 1 ohm on the reference set; the golden-file test caught every changed number, which is what it is for.)
- Sample context is typed by hand or supplied in a job file; Project B will generate it from the instrument, the barcode reader and the climate chamber.
- The three-term loss fit, temperature derating and production correlation of Project E, and the round-robin statistics of Project F, consume the `results/runs` tables and the JSON traces produced here.

8 Appendix A: reference-set results

Full table produced by run_sample on the five reference samples (explicit fixture de-embedding for the 4-port files; the 8-port file evaluated without fixture).

sample	quantity	pair	verdict	worst margin	at	measured	limit
good_15m	insertion_loss	A	PASS	+0.30 dB	1.0 MHz	0.36	0.66
good_15m	impedance_mean	A	PASS	+3.37 ohm	-	101.63	105.00
good_15m	impedance_fitted	A	PASS	+5.00 ohm	-	100.00	95.00
good_15m	impedance_max	A	PASS	+7.12 ohm	-	102.88	110.00
good_15m	return_loss	A	PASS	+8.51 dB	1.7 MHz	27.51	19.00
good_15m	impedance_min	A	PASS	+10.03 ohm	-	100.03	90.00
good_15m	phase_delay	A	PASS	+20.37 ns	3.2 MHz	73.63	94.00
good_15m	lcl	A	PASS	+27.14 dB	69.9 MHz	76.55	49.41
good_15m	lctl	A	PASS	+28.54 dB	58.7 MHz	77.20	48.65
lossy_15m	insertion_loss	A	FAIL	-4.31 dB	600.0 MHz	20.16	15.85
lossy_15m	impedance_mean	A	PASS	+1.66 ohm	-	103.34	105.00
lossy_15m	return_loss	A	PASS	+3.18 dB	2.5 MHz	22.18	19.00
lossy_15m	impedance_max	A	PASS	+4.24 ohm	-	105.76	110.00
lossy_15m	impedance_fitted	A	PASS	+4.98 ohm	-	100.02	105.00
lossy_15m	impedance_min	A	PASS	+10.41 ohm	-	100.41	90.00
lossy_15m	phase_delay	A	PASS	+20.19 ns	3.2 MHz	73.81	94.00
lossy_15m	lctl	A	PASS	+26.98 dB	107.3 MHz	75.60	48.63
lossy_15m	lcl	A	PASS	+28.40 dB	66.1 MHz	77.57	49.17
ripple_15m	return_loss	A	FAIL	-1.91 dB	391.1 MHz	6.30	8.22
ripple_15m	insertion_loss	A	PASS	+0.30 dB	1.0 MHz	0.36	0.66
ripple_15m	impedance_mean	A	PASS	+3.40 ohm	-	101.60	105.00
ripple_15m	impedance_fitted	A	PASS	+4.35 ohm	-	99.35	95.00
ripple_15m	impedance_max	A	PASS	+7.21 ohm	-	102.79	110.00
ripple_15m	impedance_min	A	PASS	+10.09 ohm	-	100.09	90.00
ripple_15m	phase_delay	A	PASS	+20.21 ns	386.6 MHz	73.79	94.00
ripple_15m	lcl	A	PASS	+27.00 dB	102.1 MHz	75.87	48.88
ripple_15m	lctl	A	PASS	+28.95 dB	104.3 MHz	77.71	48.77
unbalanced_10m	lctl	A	FAIL	-36.96 dB	146.3 MHz	10.12	47.08
unbalanced_10m	lcl	A	FAIL	-20.88 dB	86.4 MHz	28.83	49.71
unbalanced_10m	insertion_loss	A	PASS	+0.39 dB	1.0 MHz	0.27	0.66
unbalanced_10m	impedance_mean	A	PASS	+3.96 ohm	-	101.04	105.00
unbalanced_10m	impedance_fitted	A	PASS	+4.04 ohm	-	99.04	95.00
unbalanced_10m	impedance_max	A	PASS	+7.59 ohm	-	102.41	110.00
unbalanced_10m	return_loss	A	PASS	+9.49 dB	3.2 MHz	28.49	19.00
unbalanced_10m	impedance_min	A	PASS	+9.79 ohm	-	99.79	90.00
unbalanced_10m	phase_delay	A	PASS	+44.47 ns	2.5 MHz	49.53	94.00
twopair_5m	fext	A->B	FAIL	-13.20 dB	1.0 MHz	70.47	83.67
twopair_5m	next	A->B (far end)	FAIL	-6.92 dB	111.8 MHz	46.35	53.27
twopair_5m	next	A->B	FAIL	-6.60 dB	111.1 MHz	46.71	53.32
twopair_5m	insertion_loss	B	PASS	+0.52 dB	1.0 MHz	0.13	0.66
twopair_5m	insertion_loss	A	PASS	+0.52 dB	1.0 MHz	0.13	0.66
twopair_5m	impedance_fitted	B	PASS	+4.03 ohm	-	100.97	105.00
twopair_5m	impedance_fitted	A	PASS	+4.03 ohm	-	100.97	105.00
twopair_5m	impedance_mean	A	PASS	+4.20 ohm	-	100.80	105.00
twopair_5m	impedance_mean	B	PASS	+4.20 ohm	-	100.80	105.00
twopair_5m	impedance_max	A	PASS	+7.90 ohm	-	102.10	110.00
twopair_5m	impedance_max	B	PASS	+7.90 ohm	-	102.10	110.00
twopair_5m	impedance_min	A	PASS	+9.26 ohm	-	99.26	90.00
twopair_5m	impedance_min	B	PASS	+9.26 ohm	-	99.26	90.00
twopair_5m	return_loss	B	PASS	+12.47 dB	7.0 MHz	31.47	19.00
twopair_5m	return_loss	A	PASS	+12.48 dB	7.0 MHz	31.48	19.00
twopair_5m	lcl	A	PASS	+28.13 dB	90.1 MHz	77.63	49.50
twopair_5m	lctl	A	PASS	+28.27 dB	66.9 MHz	77.50	49.22
twopair_5m	lcl	B	PASS	+28.56 dB	78.9 MHz	78.50	49.94
twopair_5m	lctl	B	PASS	+29.05 dB	127.5 MHz	76.81	47.76
twopair_5m	phase_delay	A	PASS	+68.77 ns	4.7 MHz	25.23	94.00
twopair_5m	phase_delay	B	PASS	+68.77 ns	7.0 MHz	25.23	94.00

9 References

- Bockelman, David E., and William R. Eisenstadt. 1995. "Combined Differential and Common-Mode Scattering Parameters: Theory and Simulation." *IEEE Transactions on Microwave Theory and Techniques* 43 (7): 1530–39. <https://doi.org/10.1109/22.392911>.
- Frickey, Dean A. 1994. "Conversions Between S, Z, Y, h, ABCD, and T Parameters Which Are Valid for Complex Source and Load Impedances." *IEEE Transactions on Microwave Theory and Techniques* 42 (2): 205–11. <https://doi.org/10.1109/22.275248>.
- Harris, Fredric J. 1978. "On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform." *Proceedings of the IEEE* 66 (1): 51–83. <https://doi.org/10.1109/PROC.1978.10837>.
- IEC. 2009. "IEC 61156-1: Multicore and Symmetrical Pair/Quad Cables for Digital Communications – Part 1: Generic Specification." International Electrotechnical Commission.
- IEEE. 2016. "IEEE Std 802.3bp-2016: Physical Layer Specifications and Management Parameters for 1 Gb/s Operation over a Single Twisted-Pair Copper Cable (1000BASE-T1), Clause 97." IEEE Standards Association.
- Keysight Technologies. 2012. "Time Domain Analysis Using a Network Analyzer, Application Note 5989-5723EN." Keysight Technologies.
- Kurokawa, K. 1965. "Power Waves and the Scattering Matrix." *IEEE Transactions on Microwave Theory and Techniques* 13 (2): 194–202. <https://doi.org/10.1109/TMTT.1965.1125964>.
- OPEN Alliance TC9. 2020. "Channel and Components Requirements for 1000BASE-T1 Link Segment Type a, Version 2.3." OPEN Alliance SIG. https://opensig.org/wp-content/uploads/2024/01/TC9-Gbps-channel-and-components-2.3_public.pdf.
- Paul, Clayton R. 2008. *Analysis of Multiconductor Transmission Lines*. 2nd ed. Wiley-IEEE Press.