# TDTR_vH3: rapid data processing, analysis, and note-keeping for general thermal modeling.

by Greg Hohensee, based on Joe Feser's MATLAB scripts and beam-offset code.

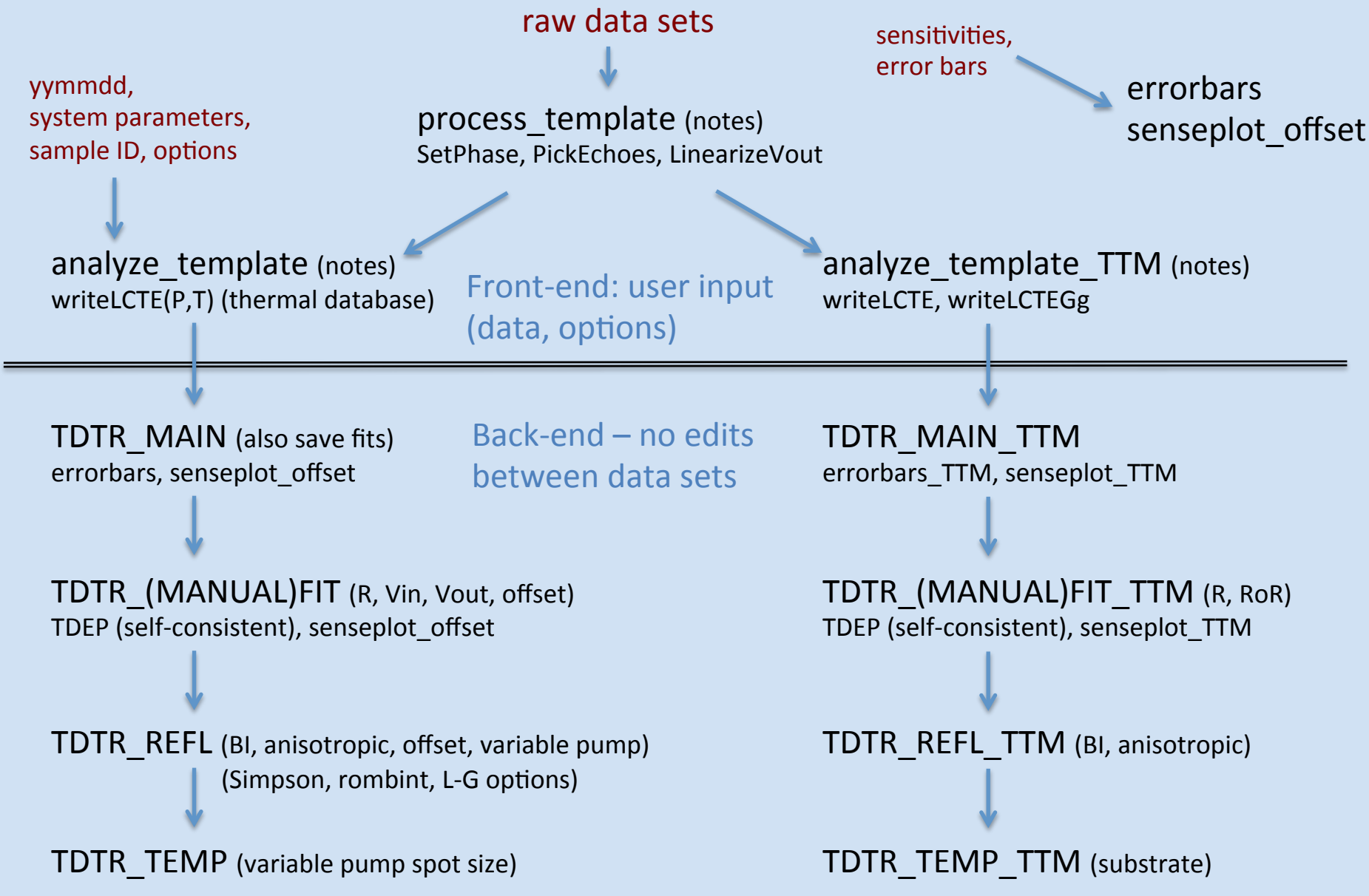
# Why?

- In weekly meetings, I was expected to:
  - Re-analyze whole data sets (sometimes!)
  - Not mix up thermal parameters by accident.
  - Fit data manually for intuition.
  - Show ALL thermal model parameters and plot fit results in a specific format.
  - For low temperature measurements, self-consistently update thermal parameters for steady-state heating.

# Features Overview

- Two front-end scripts for each TDTR session: your analysis saved, no back-end customization required.
- Assisted phase & time shifts, ps acoustics and Brillouin reading.
- Integrated database for your materials' thermal parameters.
- Manual fitting to any signal, with option for sensitivity calculations.
- Self-consistent steady-state temperature adjustment.
- Generates and saves detailed fit result figure.
- NEW: Parametric sensitivity plots

# Script Hierarchy



# Parameter Bundles, or: I should have made objects instead of cell arrays.

- datparams - {tdelay data datadir offset}
- matparams - {LCTE aniso BI n_toplayer TCR doughnut}
- sysparams - {tau_rep f r_pump r_probe}
- calparams - {Zind sigfit intscheme nnodes consider_error LCTE_err T0_err P0_err}
- Tparams - {T0, T_LCTE, A_pump, A_probe, absC, perpulse, jabs, jtrans}

- INITIALIZE_CELLPARAMS_vH3.m contains extensive comments explaining all these parameters. It also handles unpacking and repacking of these cell parameter arrays for many other scripts; update this script if you make changes to the cell arrays.

# Front-end: Process and Analyze

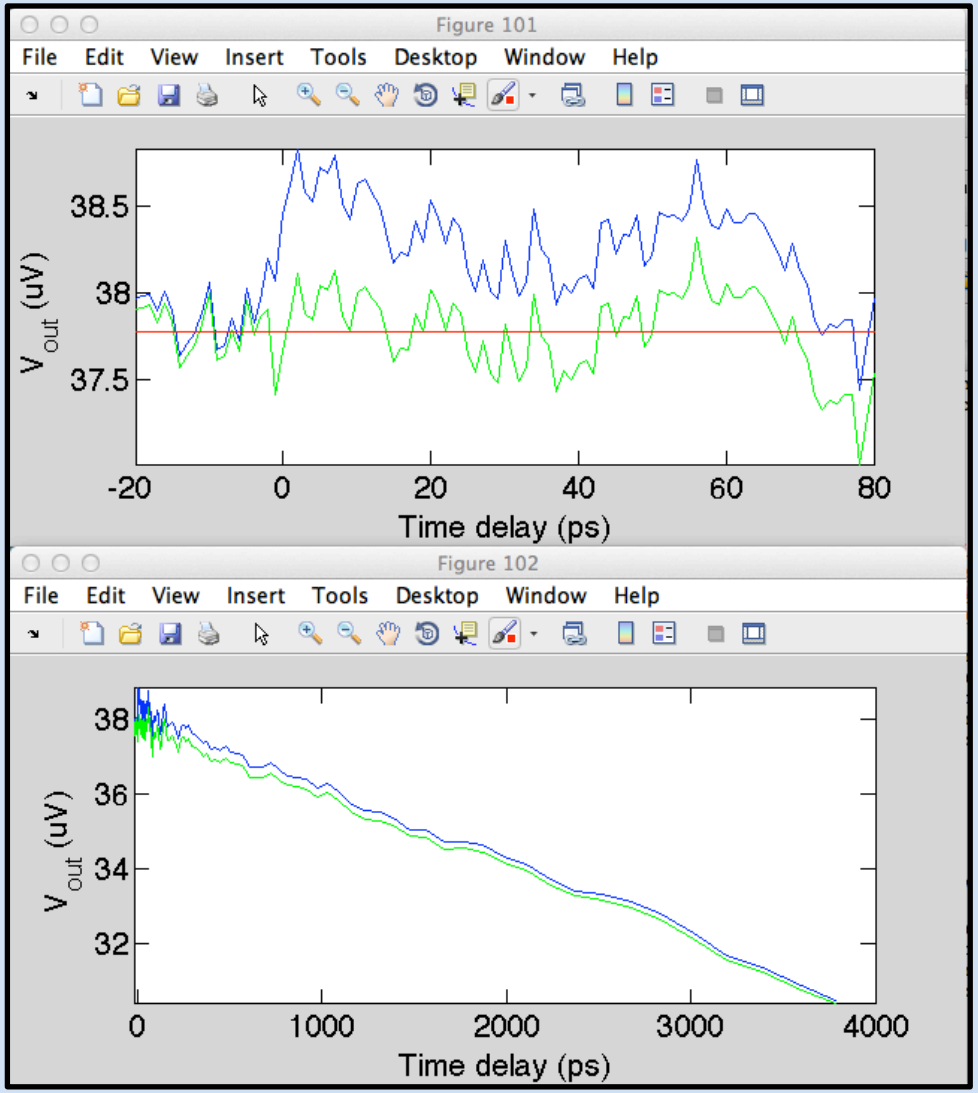
- Easy documentation and revisiting: just save new (dated) copies of process_yymmdd.m and analyze_yymmdd.m scripts for each TDTR session.

- **Process** helps setting zero of delay time, setting the phase, picking picosecond acoustics, and examining the signals of your data series at particular delay times.
  - Saves phase- and time-shifted data with the "_shifted.txt" suffix to "yymmdd_edit" folder.

- **Analyze** lets you set all system and thermal parameters, and customize many aspects of the modeling and fitting process. Model parameters are specified as arrays, so it can loop through a full TDTR session and save the solutions.

# SetPhase.m

- Shows short- and long-time delay $V_{out}(t)$, raw versus shifted.
- Options:
  - Manual (Joe Feser's code)
  - Automatic (compare average $V_{out}(t)$ before/after 0 ps)
  - Linear fit (for seeing ratio without $V_{out}(t)$ noise)
- Other process scripts:
  - **set_t0.m** – helps you set zero of time delay.
  - **PickEchoes.m** – lets you pick picosecond acoustic timing.
  - **PickEchoesB.m** – helps with Brillouin signals.
  - **AutoSetPhase.m** – Auto shifts data without user input.



# TDTR_MANUALFIT_vH3.m: manual fitting for intuition

```
Pressure is 3.9 GPa
Manual fitting to ratio -V(in)/V(out)...

Z =

    0.4373


Mat =

Current material fit parameters LCTE(i,j):
L(2) = 0.1000 W/m-K
L(1) = 7.0000 W/m-K

Enter 1 if done, 2 for a sensitivity plot, 3 to rescale; else hit "Enter": 4
Hey! Invalid input. Go home, you are drunk.
Adjust parameter L(2): 0.15
Adjust parameter L(1): 9
```

PP
T
T0
TCR
TDebye
T_LCTE
Tparams
VinVout
Vin_data
Vin_raw
Vout_data
Vout_raw
Voutlinfit
XPsubs
Xguess

- Can generate a sensitivity plot at any time.
  - To speed up sensitivity plotting, go into senseplot_offset_vH3.m script and edit “LCTE_sens_consider” and “sys_consider” variables to select which sensitivities to calculate, and which ones to skip.

- “3 to rescale”: manually shift normalization reference level for $V_{in}(t)$ and $V_{out}(t)$ fitting, in case of noise or acoustics.

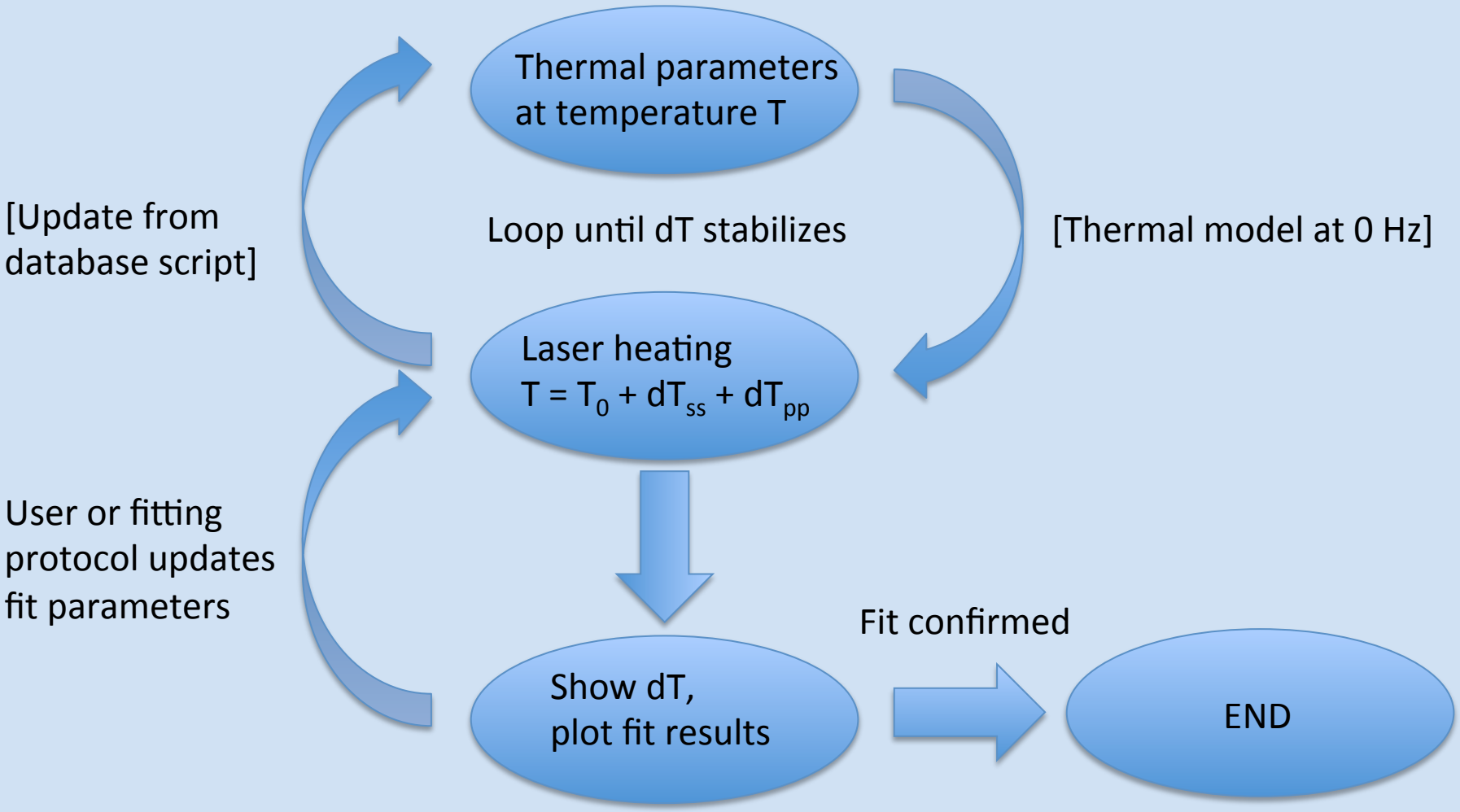
# Sensitivity conventions and coupled thermal parameters

- Sensitivity (*and error bars*) are **never** computed relative to raw data.
  - Sensitivities and error bar computations use perturbations from your best-fit model: this avoids extrinsic effects from noise and second-guessing your manual fitting judgment.

- Anisotropy "eta" = $\Lambda_x$ / $\Lambda_z$, in-plane divided by cross-plane thermal conductivity. $\Lambda_z$ is the "lambda" variable that you're familiar with.
  - Convention: if model layer is anisotropic, perturbing $\Lambda_z$ also updates eta so that $\Lambda_x$ remains fixed relative to $\Lambda_z$. Sensitivity to eta, S(eta), is equivalent to S($\Lambda_x$).

- Transducer absorption layers:
  - Program detects absorption and transducer layers (user specifies this in analyze script!). Perturbations are done for the transducer as a whole, not the two model layers separately.

# Temperature-dependent thermal parameters, self-consistently



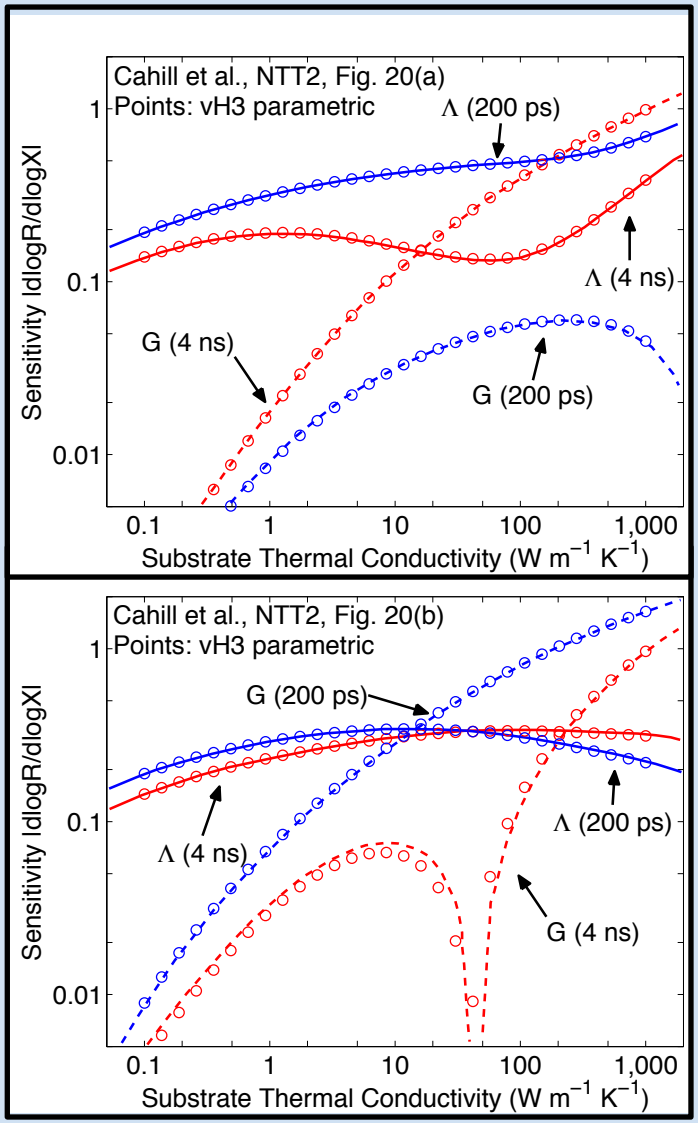
# TDTR_TTM_vH2: Two-channel model for *substrates only*

- Two-channel thermal model designed by Rich.
  - See: R. B. Wilson *et al*., *Phys. Rev. B*. **88**, 144305 (2013).
- Can fit thermal model to the ratio of two TDTR measurements at two different modulation frequencies. Also sensitivities.
  - See: G. T. Hohensee *et al*., *Phys. Rev. B.* **89**, 024422 (2014).
- Designed for transducer on bulk spin-ladder sample: for multilayer 2- or 3-channel thermal modeling, ask Rich.
- User may need to customize anisotropy in two-channel substrate using the back-end scripts.
- TTM_vH2 has *not* been updated to vH3.

# parametric_senseplot_vH3.m: Parametric Sensitivity Plotter

- Controlled by sense_setup.m, or sense_setup_retro.m for users of Joe Feser's TDTR_MAIN script.
- S(X) = dlog(R)/dlog(X) versus model parameter Y or offset, not just time delay.
- R can be ratio, normalized V(in) or V(out), beam offset, or FWHM of beam offset.

- Verified against Cahill group publications
  - David Cahill *et al.*, "Nanoscale Thermal Transport II," *Appl. Phys. Rev.* **1**, 011305 (2014)
  - Yee Kan Koh, *Ph.D. Thesis*, UIUC (2010)
  - Xiaojia Wang *et al.*, *Phys. Rev. B.* **88**, 075310 (2013)
  - But not beam offset! Proceed with caution, verify against published sensitivities [Such as: J. Feser et al., *Rev. Sci. Instrum.* **85**, 104903 (2014)]



# Thank you for your attention!

Slides are be posted on GitHub along with the TDTR_vH3 package.

http://github.com/gthohensee/TDTR_vH3_pub