

# Update on Time Resolved X-Ray Diffraction Codes

Eric Landahl

February 3, 2017

## 1 About this document

This is a summary of current progress on the TRXD code written and maintained by Eric Landahl, with considerable help from Soohyong Lee and G. Jackson Williams. This document will update regularly.

## 2 Purpose of the Codes

1. Rocking curve calculations are benchmarked against established codes where possible as well as analytical solutions
2. Written entirely in MatLAB
  - Compiling not required
  - Runs on multiple operating systems
  - Octave compatible
  - Student friendly
3. Documented, evolving, and can be used and adapted by others
  - Maintained on GitHub and updated regularly
  - Revision control and branching is used for development
  - Extensive comments, manuals, and benchmarks included with distribution
4. Efficient and flexible operation
  - Extensive use of remeshing: the spatiotemporal grid for electron diffusion, thermal diffusion, acoustic propagation, and diffraction calculations do not need to be the same, or even have evenly spaced steps
  - Adaptive step sizes in the diffraction algorithm allow speed and accuracy to be tuned
  - Vectorization and boolean indexing are used for speed and are parallel computing compatible
5. Multiple crystals and x-ray energies
  - Presently focusing on 7 - 14 keV, Ge, GaAs, Si, and InSb symmetric 004 reflection
  - Distribution includes self contained diffraction parameter (e.g.  $\chi$ ) database
6. Easy to develop strain functions
  - Distribution includes database of common material properties (e.g. sound velocities, specific heat, etc.)

### 3 Dynamical diffraction calculations: Issues with implementation of the Wie et al. algorithm

There are a lot of problems in taking a nearly 25 year old paper designed to calculate x-ray diffraction from multilayers and superlattices and apply it to continuously varying strain parameters. In this section I will be trying to clean up and solve some issues that have plagued the implementations of this algorithm by the time-resolved x-ray community. For now I will just briefly list some of the issues.

#### 3.1 Substrate boundary condition

The iterative solution proposed by Wie et al. relies on an unstrained semi-infinite crystal being present below the strained layers. This means that the strain must approach zero before the calculation begins. Jackson suggested multiplying the strain by

$$e^{-z/L_{ext}} \quad (1)$$

where  $z$  is the depth below the surface and  $L_{ext}$  is the extinction length. This tapers off the large impulsive acoustic strains that travel deep into the crystal, which otherwise would at some timepoints violate the zero-strain boundary condition. The problem with this approach is that the tapering is too severe. Specifically for thermal transport problems, where the shape of the strain profile is being studied over several microns, this tapering alters the outcome of the rocking curve calculation. This implementation instead uses a deeper strain tapering function,

$$\frac{1 - \operatorname{erf}\left(\frac{2z}{L_{ext}} - 8\right)}{2}. \quad (2)$$

A consequence of this new taper function (see Fig.1) is that the calculation must begin much deeper. The code checks to make sure that the given strain profile is at least  $5L_{ext}$  deep, and begins calculations just inside of this point, where the strain is therefore forced to be zero.

#### 3.2 Depth step size

An unfortunate consequence of choosing such a deep starting point is that more iterations of the Wie et al. algorithm are required, if a fixed depth step is used. This motivates some thought about the choice of the correct depth step. Wie et al. intended their method to be used for multilayers and superlattices, where the depth step was given by the thickness of a particular layer. In practice, we have used fixed depth steps, reduced in size until the output result no longer changes. A better approach for continually variable strain profiles is to use an adaptative step size. This allows the step size to be changed as the strain varies to maintain a maximum error tolerance. An adaptative step size algorithm is used in the function `WieAdapt.m`. Briefly, a full step is compared to the (presumably more accurate) result for two successive half steps. The difference is compared to a relative error tolerance to determine the next step size. A plot illustrating the adaptative step size algorithm at will be added to this document soon. Other more sophisticated adaptative algorithms could be developed. Another option is solving the simplified Tagaki-Taupin equations using an adaptative Ordinary Differential Equation routine (e.g. MatLAB's `ode45`).

#### 3.3 Ambiguity in sign of the square root of an imaginary number

The Appendix in Wie et al. gives a useful, but somewhat under-developed description of taking the square roots of a complex number. I have found that for my given choices of the signs of  $\chi$  that the following sign choices give the physically realistic solutions:

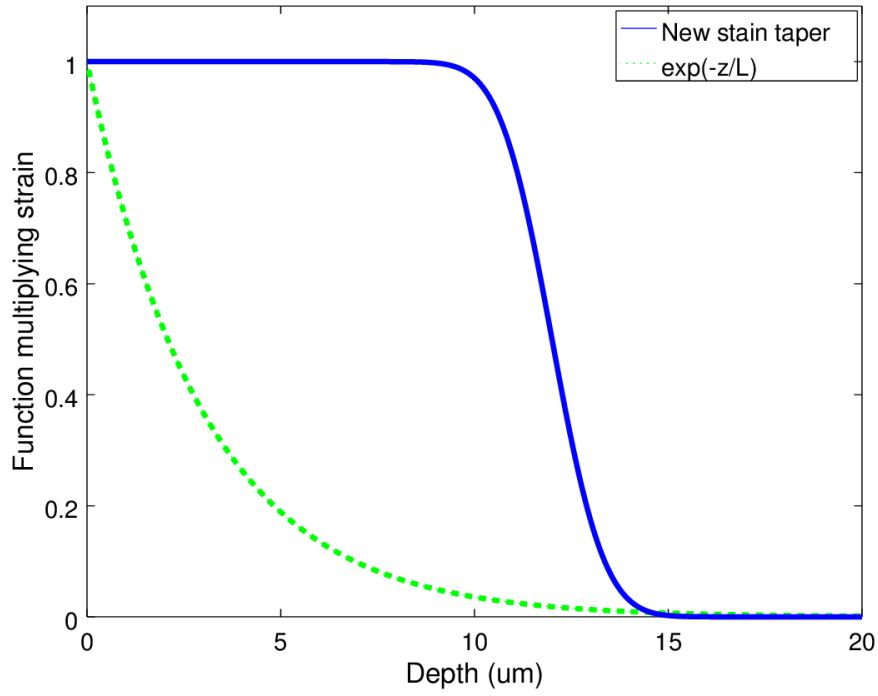


Figure 1: Envelope or taper function applied to depth-dependent strain profile to make sure that the strain is unchanged within several extinction lengths of the surface, but tapers to zero before the calculation begins at  $5L_{ext}$ . For this example, an extinction depth of  $L_{ext} = 3\mu m$  was used.

parameter	TRXD	$\chi_{0h}$
$\chi_0'$	-	-
$\chi_0''$	+	+
$\chi_h'$	-	+
$\chi_h''$	+	+

Table 1: Sign conventions for susceptibilities  $\chi_0 = \chi_0' + i\chi_0''$  is the susceptibility for the incident beam and  $\chi_h = \chi_h' + i\chi_h''$  is the susceptibility for the diffracted beam.

line number	variable	description
19	<b>energy</b>	x-ray energy in keV
16	<b>crystal</b>	'GaAs', 'Ge', 'InSb', or 'Si'
20	<b>fluence</b>	in mJ/cm <sup>2</sup> , or timepoint number
15	<b>model</b>	see next section
81	<b>plot_opts</b>	'none' or 'final'
82	<b>angle_res</b>	blurs plots, only used for full 2D calcs

Table 2: Parameters that can be easily set by the user when running the script `TRXD_Driver_example.m`.

$$TBA \tag{3}$$

The sign conventions I used for the electric susceptibilities are compared to those of Sergey Stepanov's  $\chi_{0h}$  program in Table 1. The signs are adjusted upon input in the program `TRXD.m`. More work will need to be done to clarify exactly why these particular sign conventions should be used.

### 3.4 Normalized depth

The biggest challenge, and the one not completely resolved is the proper choice of normalized length scale  $A$ . The definition given by Wie et al. does not appear to agree with either analytical models or GID-SL. I have implemented a linear correction in  $\sin(\theta_B)$  that returns the proper values for GaAs from 8 - 11 keV. Extending to other crystals and a wider energy range should be possible with a higher-order, empirical correction. This area definitely needs further study.

## 4 User's Manual

The manual still needs to be written. In the meantime, the best way to access the codes is to run the simple script `TRXD_Driver_example.m`. Table 2 gives a summary of some parameters that can be easily adjusted by the user.

### 4.1 Benchmarking against Sergey Stepanov's GID-SL diffraction calculations

The first step upon downloading the code package should always to use the 'benchmark' method as described in this section. It is not necessary to run GID-SL, since the standard benchmark file is included in the distribution. However, to check other crystals, energies, or simple analytic strain models GID-SL can be run from [http://x-server.gmca.aps.anl.gov/GID\\_sl.html](http://x-server.gmca.aps.anl.gov/GID_sl.html), and then choosing the option for "Symmetric Bragg diffraction from multilayers and superlattices". The standard benchmark condition is 10 keV, GaAs 004, with a constant strain of  $10^{-4}$  over the first 2  $\mu m$  of depth. GID-SL should be set to these parameters, with a single line of code in the textbox specifying the single

**Template: symmetric Bragg diffraction from multilayers and superlattices.**  
(simplified input -- for **coplanar cases only**)

---

**X-rays specified by:** Energy (KeV) Value=10 Line= Cu-Ka1 ? Polarization= Sigma

**Crystal:** GaAs ? Auto DB for f',f" Sigma=0. A W0=1. Wh=1.

**Bragg Reflection:** 4 0 0 Substrate da/a=0.

**Geometry specified by:** angle of Bragg planes to surface ('+' for g0>gh) , Value:0. degr.

**Scan:** from -.01 to +.01 degr. Scan points:401 Invert axis: ☐ Plot argument: incidence angle

**Approximations:** alpha\_max=1.E+8 \*|xh| ?

☐ watch progress Submit Query (single click, please!)

---

**Top layer profile (optional):**  
period=  
t= sigma= da/a= code= x= code2= x2= code3= x3= code4= x0= xh= xhdf= w0= wh=  
end period

t=20000 code=GaAs da/a=1e-4

**Available codes:**  
[?] Crystals:  
ALP  
alpha-  
Fe  
ALsb  
Alumini

Figure 2: Proper configuration of GID-SL to duplicate the standard benchmark results.

program	line number	variable	standard benchmark value
TRXD_Driver_example.m	19	energy	10
TRXD_Driver_example.m	16	crystal	'GaAs'
TRXD_Driver_example.m	15	model	'benchmark'
/main/TRXD.m	190	longitudinal	=1e-4 * (z<=2e-6);

Table 3: Parameters that are used in for comparission with the GID-output file /benchmarks/benchmark.txt included in the distribution.

strained layer as  $t = 20000$  code = GaAs da/a =  $1E4$ . Note that the thickness units used by GID-SL are in Angstroms. The angular range should be set from -0.01 to +0.01 degrees. Figure 2 shows a screenshot of how this screen should look. The results should be opened in .dat format and saved as /benchmarks/benchmark.txt. This file will be automatically plotted alongside the TRXD calculations when the method is chosen to be 'benchmark'. The standard benchmark conditions are given in Table 3. The results of running the code should give one logarithmic and one linear scale plot, each with 3 lines corresponding to the unstrained rocking curve, the TRXD calculated strained rocking curve, and the output of GID-SL. *There are still some minor discrepancies between GID-SL and TRXD that are being worked out.*

## 4.2 Testing using an external strain file provided by SHL

Soo provided a file Sool.txt containing a previously calculated strain profile. This strain file was padded with zeros out to  $20 \mu m$  and resaved as strain\_file.txt. A one dimensional array of times was saved as time\_file.txt, and a one dimensional array of depths was saved as depth\_file.txt. These 3 files will be read anytime the model variable is set to 'strainFile1D' to look at single timepoints. In this model, the otherwise unused variable fluence is used to set the particular timepoint chosen. To briefly summarize the provided file, an initially shallow strain genrates a travelling bipolar impulsive

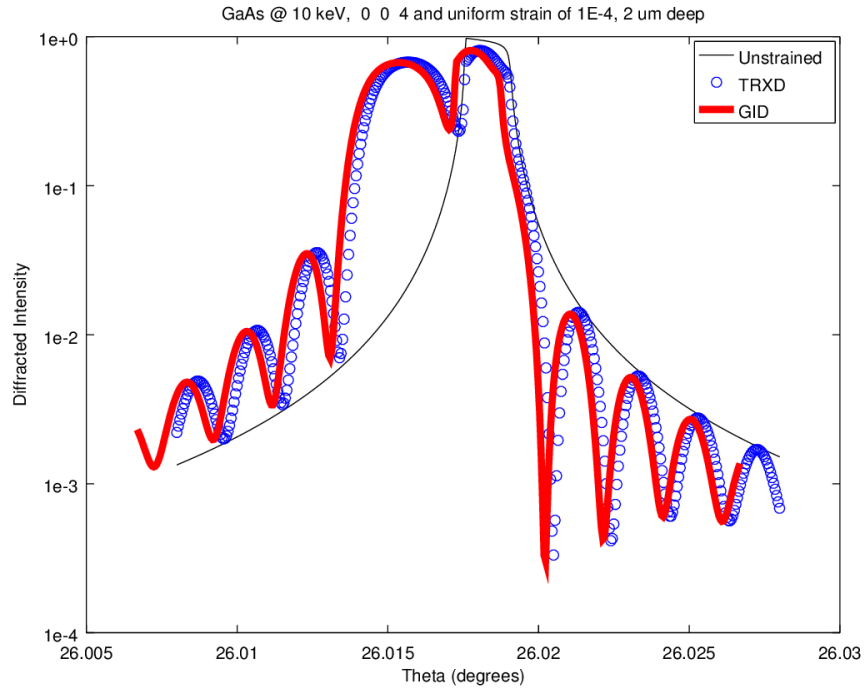


Figure 3: Results using the configuration given in Table 3 and Fig. 2.

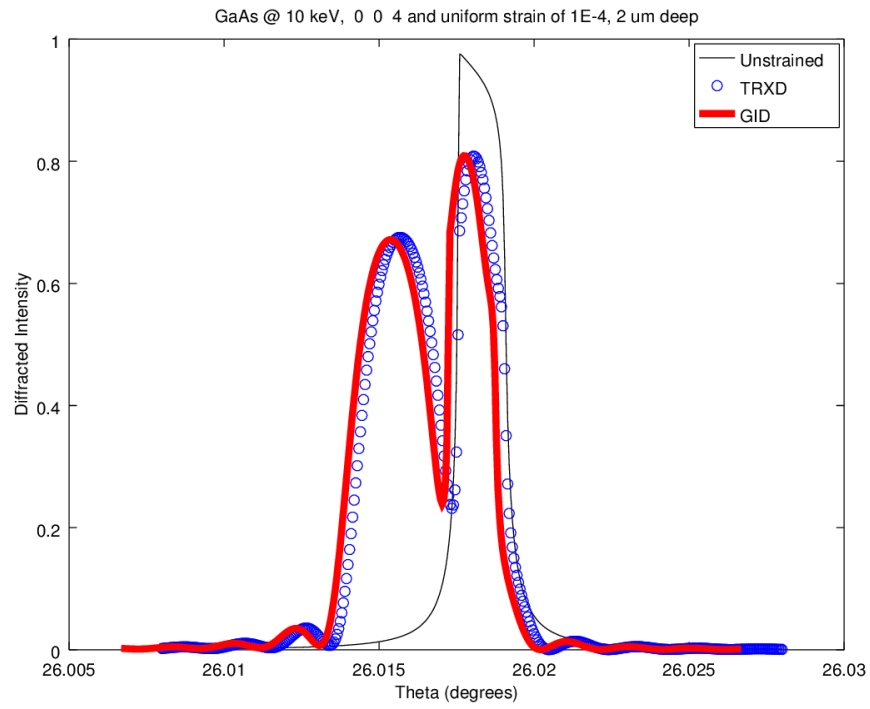


Figure 4: Results using the configuration given in Table 3 and Fig. 2.

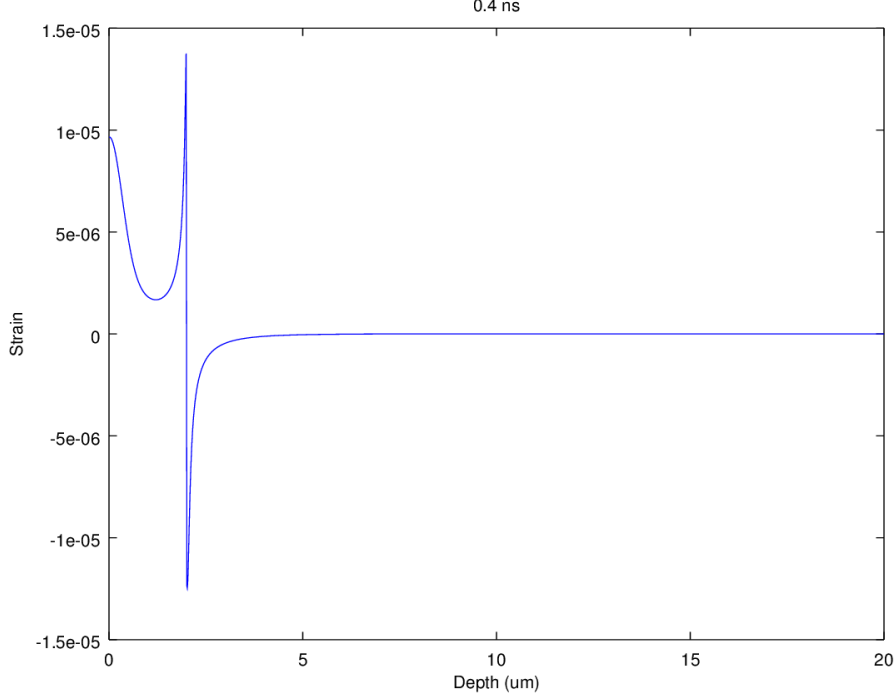


Figure 5: External file, 400 ps strain profile.

wave with a short duration. As it moves away from the surface, the initial strain profile undergoes diffusion, resulting in a deeper but less dramatic strain. The `model` variable can be set to 'strainFile1D' to look at single timepoints. In this model, the otherwise unused variable `fluence` is used to set the particular timepoint chosen.

To demonstrate the validity of the code, I have chosen two timepoints and run them for GaAs 004 10 keV. The first is 400 ps (200th timepoint). The strain is shown in Fig. 5. The strain is confined to about 2  $\mu m$  of the surface. The rocking curves for strained and unstrained crystal are shown in Figs. 6 and 7. As expected, the x-ray diffraction pattern shows a slight shift and considerable modulation fringes. Both can be analyzed. The average strain is calculated using the differential Bragg formula,

$$\frac{\Delta d}{d} = -\Delta\theta \cot(\theta_B) \quad (4)$$

where  $d$  is the lattice plane separation,  $\Delta d$  is the change in lattice parameter,  $\Delta\theta$  is the peak centroid shift determined from the two rocking curves, and  $\theta_B$  is the Bragg angle. The program uses this formula to produce the following output at 400 ps:

The centroid shift is -0.086 mdeg or -1.5e-06 radians. For GaAs 0 0 4 at 10.0 keV the estimated average strain is 3.1e-06 .

From inspection of Fig. 5, 3.1e-06 is a reasonable estimate for the average strain in the first few microns. The fringe spacing can be analyzed to determine the thickness of the strained layer using

$$t = d \frac{\theta_B}{\delta\theta_{fringe}} \quad (5)$$

where  $t$  is the thickness of the strained layer and  $\delta\theta_{fringe}$  is the fringe separation angle. It is easiest to determine the fringe spacing from the logarithmic plot (Fig. 7). For the 400 ps data, the fringe spacing

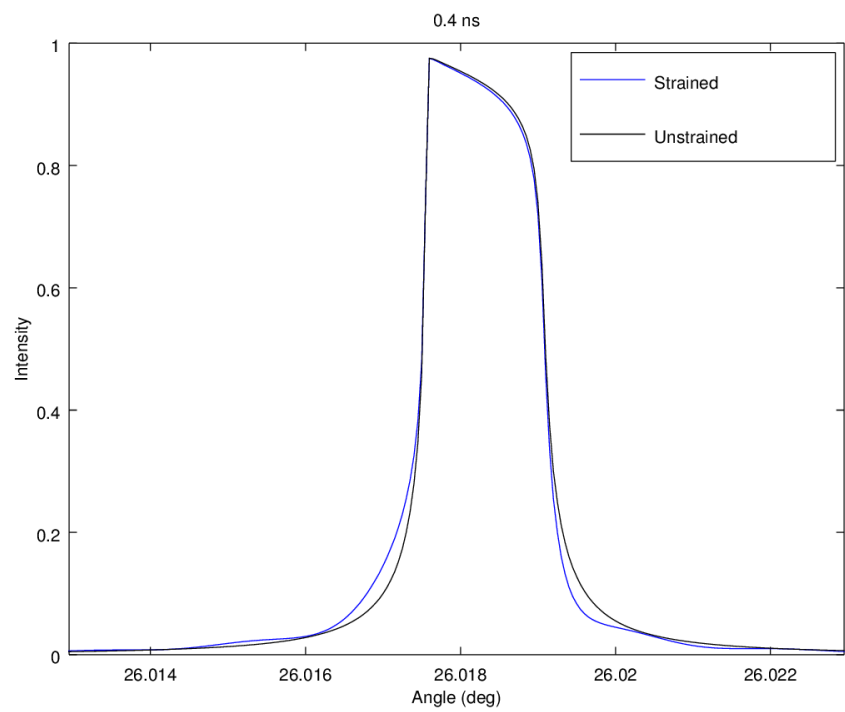


Figure 6: External file, 400 ps rocking curve (linear scale).

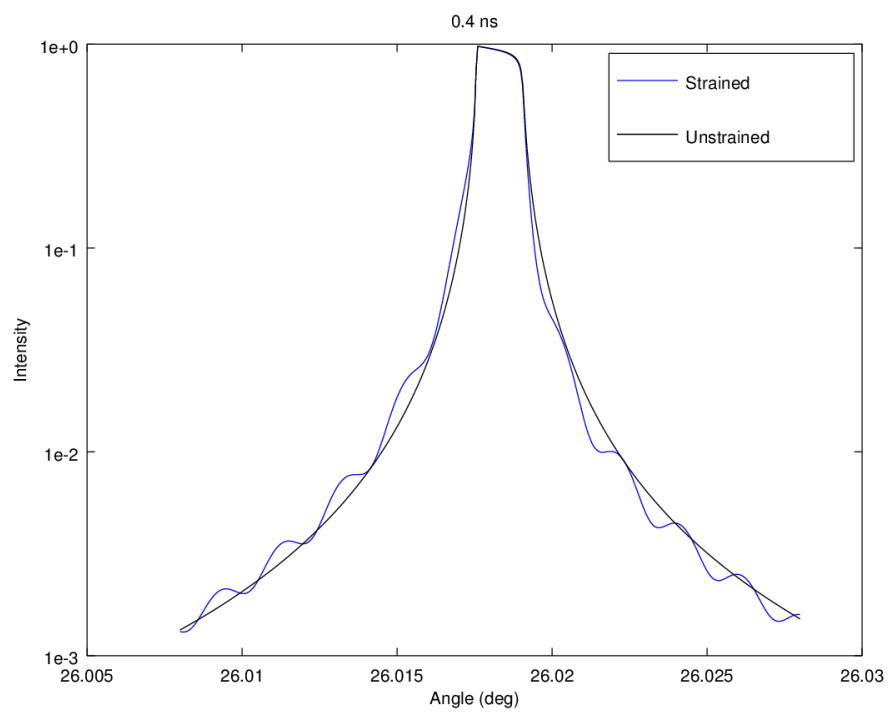


Figure 7: External file, 400 ps rocking curve (logarithmic scale).



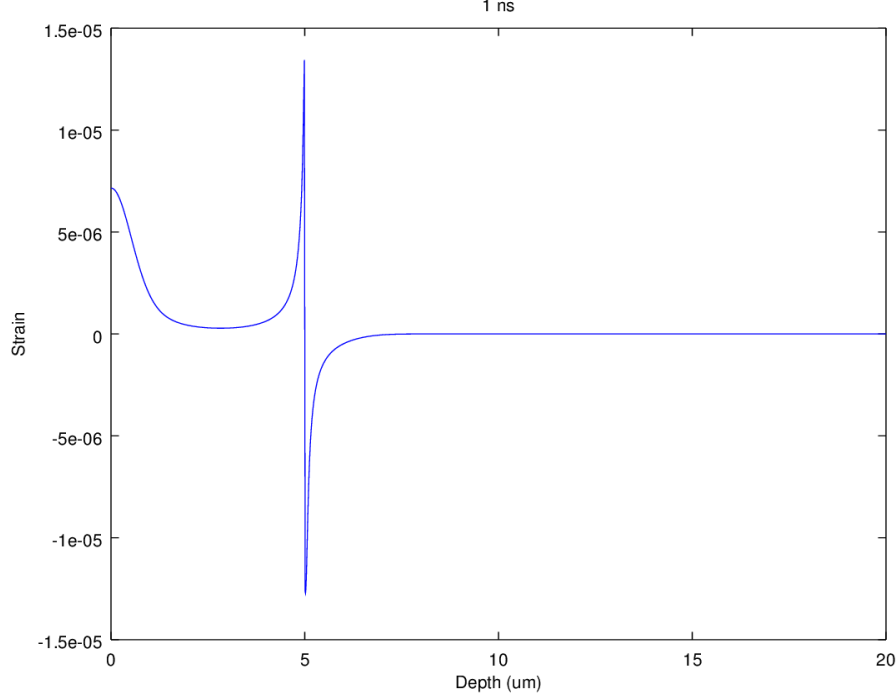


Figure 8: External file, 1 ns strain profile.

is 2 mdeg, and for the GaAs 004 reflection the lattice spacing  $d = 0.14133$  nm. This calculation yields a strain layer depth  $t = 1.837 \mu\text{m}$ , very close to inspection of Fig. 5.

This analysis is repeated for a 1 ns time delay, and is shown in Figs. 8, 9, and 10. The fringes are closer together, indicating that the sharp impulsive strain wave is farther from the surface. The fringe spacing is now only 0.8 mdeg, corresponding to  $4.6 \mu\text{m}$ , in agreement with the 1 ns strain profile shown in Fig. 8. However, as expected, the average strain is about the same as at 400 ps. The code produces the following output:

The centroid shift is -0.079 mdeg or  $-1.4\text{e-}06$  radians. For GaAs 0 0 4 at 10.0 keV the estimated average strain is  $2.8\text{e-}06$  .

Although not shown here, the average strain value remains very similar at the end of this particular run (about 2 ns) although the fringes disappear as the impulsive wave moves out of the x-ray probe depth. It should be noted that actual experimental observation of these very finely structured fringes would likely be impossible due to finite angular and temporal resolution, which are not taken into account using this method.

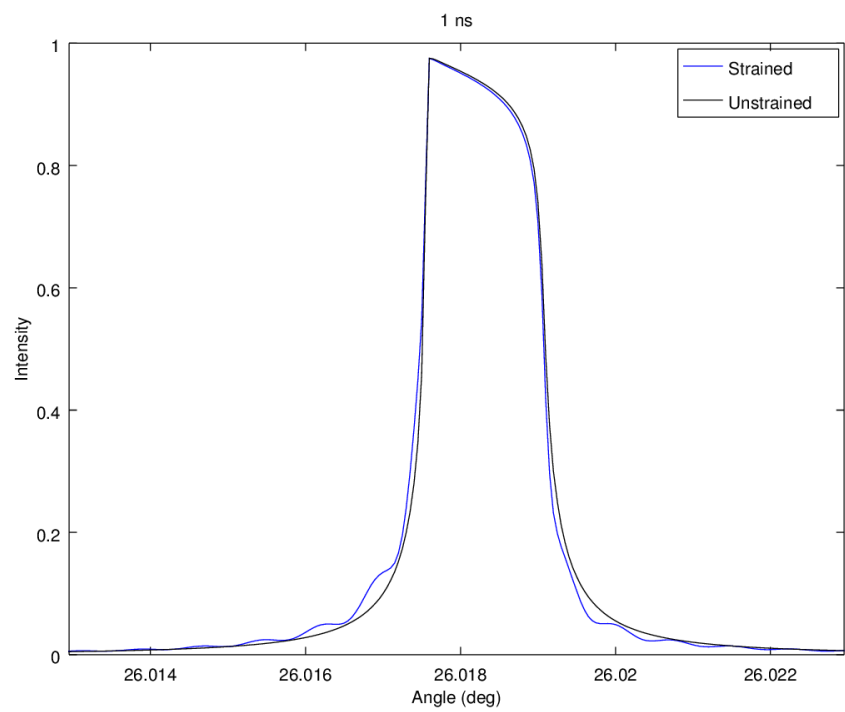


Figure 9: External file, 1 ns rocking curve (linear scale).

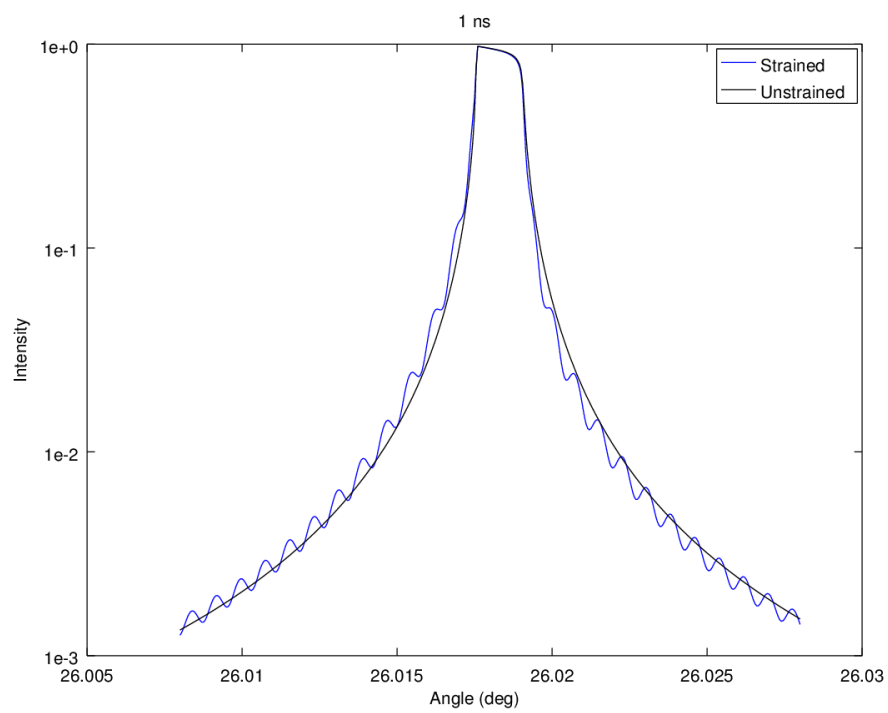


Figure 10: External file, 1 ns rocking curve (logarithmic scale).

### 4.3 Thermal film model

Please see the associated documents in `/strain_functions/thermalFilmdocs/`.

## Appendix: Code

The most recent and up to date code should be obtained from [https://github.com/elandahl/dynamical-diffraction/tree/TRXD/strain\\_functions](https://github.com/elandahl/dynamical-diffraction/tree/TRXD/strain_functions).