

BIREFRINGENCE DATA ANALYSIS

By

Christopher Bailey, Ginmo Chung, Alvaro Guevara

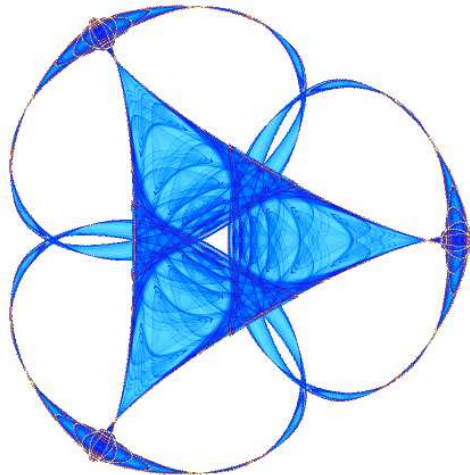
Sean Hardesty, Joseph Kenney, Sarthok Sircar

and

Douglas Allan (mentor)

IMA Preprint Series # 2133-1

(September 2006)



INSTITUTE FOR MATHEMATICS AND ITS APPLICATIONS

UNIVERSITY OF MINNESOTA
400 Lind Hall
207 Church Street S.E.
Minneapolis, Minnesota 55455-0436

Phone: 612/624-6066 Fax: 612/626-7370
URL: <http://www.ima.umn.edu>

Birefringence Data Analysis

Christopher Bailey, Ginmo Chung, Alvaro Guevara,
Sean Hardesty, Joseph Kenney, Sarthok Sircar,
and Douglas Allan

Institute for Mathematics and its Applications,
Workshop on Mathematical Modeling in Industry

Abstract

This paper describes the mathematics and numerical implementation in Matlab for analyzing birefringence data to assess laser damage of glass. The physical system under consideration is silica glass (SiO_2) for lenses used in photolithography for computer chip manufacture. The 193-nm light used in current photolithography systems is known to damage the glass over time (with billions of pulses of exposure). One feature of the damage is a permanent structural change with an associated density change that produces permanently altered optical properties. When the optical property change exceeds roughly 10^{-6} , the smallest features on a silicon wafer cannot be resolved. It is of interest to characterize the laser-induced damage with precise measurements. One method to characterize damage involves exposing a bar of glass using a controlled exposure (through an aperture) and measuring the birefringence periodically through the course of the exposure. The birefringence is measured in the direction of exposure on a two-dimensional grid of points. This paper describes how to analyze the resulting map of birefringence using a calculation of the strain field in the glass following exposure and a calculation of the related strain-induced birefringence. The methods are implemented in a set of Matlab routines that manage the reading of experimental data, subtraction of a birefringence baseline after aligning the baseline with the raw data, calculation of the theoretical birefringence using Finite Element Analysis, accounting for the measurement resolution, and finally calculation of a single overall scale factor (the unconstrained density change) that characterizes the level of damage. All these manipulations are automated and managed within a convenient Graphical User Interface that also provides for plotting both experimental and theoretical results.

Keywords: birefringence, strain-optical response, laser damage, glass, silica, elastic, photolithography

1 Introduction

Photolithography for the modern semiconductor industry demands high performance optical materials for projection systems with highly uniform index of refraction and low birefringence

[1, 2, 3]. The energetic UV photons from pulsed ArF Excimer lasers (193nm, or 6.4eV) currently in use are known to create permanent damage in silica over the course of billions of pulses. Small changes in optical properties resulting from permanent structural changes due to laser damage can produce image degradation, which can cause loss of functionality for the smallest features on a chip. Thus it is of interest to have a precise measure of laser-induced damage.

The goal of this project is to develop a set of algorithms, implemented in Matlab, that reads and analyzes a birefringence map for a glass sample after exposure to a UV laser. The purpose of the analysis is to characterize how much strain (density change) has been produced in the glass by the laser exposure. This result will be reduced to a single number (the fractional density change $\frac{\delta\rho}{\rho}$) accompanied by a quality of fit assessment. The analysis involves many steps and several choices that are managed by a Graphic User Interface (GUI). The steps include (1) reading a baseline file containing a map of the birefringence of the sample before laser exposure, (2) reading a raw data file containing the post-exposure birefringence, (3) optimally aligning these two maps in order to subtract the baseline from the raw data to get the resulting laser-induced birefringence, (4) locating the center of the exposure relative to the sample edges, (5) using the damage location as an input to a theoretical calculation of the strain field in the sample, (6) calculating a predicted birefringence based on the strain field, (7) incorporating the experimental resolution of the birefringence measurement apparatus due to its beam size, (8) comparing the predicted and measured birefringence, (9) finding the single overall scale factor $\frac{\delta\rho}{\rho}$ that best scales the theory to the experiment, and (10) calculating an error bar on the final $\frac{\delta\rho}{\rho}$. The report is organized as follows: Section 2 briefly describes the mathematics involved in the continuum elastic theory to calculate the strain field and in the strain-optical response to calculate the final birefringence results; Section 3 discusses the various aspects of the algorithms used to solve the equations of Section 2; Section 4 discusses steps involved in data alignment and subtraction, including the finding of the location of the exposure relative to the sample edges; Section 5 discusses the Finite Element Analysis formulation; Section 6 briefly mentions the convolution used to match the resolution of the birefringence measurement equipment; Section 7 describes the extraction of the Model density scale to match the experimental data; Section 8 discusses the overall organization of the data files and the GUI; Section 9 mentions suggested improvements and additional features; finally Section 10 gives an Acknowledgement.

This work was performed in the context of a 10-day workshop on Mathematical Modeling in Industry at the Institute for Mathematics and its Applications (<http://www.ima.umn.edu>).

2 Mathematical formulation

2.1 Elastic model

The constitutive equation for linear elasticity, relating stress and strain tensors[4], is given by Hooke's law

$$\sigma_{ij} = C_{ijkl}(\epsilon_{kl} - \epsilon_{kl}^0) \quad (1)$$

where $\epsilon_{ij}^0(\vec{x}) = -\frac{1}{3}\frac{\delta\rho}{\rho}f(\vec{x})\delta_{ij}$ is an initial strain field caused by densification of the experimental glass sample due to the laser exposure. Here σ_{ij} is the stress tensor, ϵ_{kl} the strain tensor, and C_{ijkl} the rank 4 tensor of elastic stiffnesses. The three-dimensional function $f(\vec{x})$ is meant to represent the normalized degree of density change in the glass, e.g. it could be taken as a cylinder along the centerline of the sample with value $f(\vec{x}) = 1$ inside a given radius and $f(\vec{x}) = 0$ outside, to represent a beam of exposing laser light sent through a circular aperture and exposing the sample along its centerline. The overall scale of density change is given by the dimensionless constant $\frac{\delta\rho}{\rho}$, which is the negative of the volume strain $\frac{\delta\rho}{\rho} = -\frac{\delta V}{V}$ where ρ is density and V is volume. In an isotropic material, the fourth-rank tensor C_{ijkl} simplifies[4] to the special form

$$C_{ijkl} = \mu(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}) + \lambda\delta_{ij}\delta_{kl} \quad (2)$$

so that

$$\sigma_{ij} = \lambda\text{tr}(\epsilon)\delta_{ij} + 2\mu\epsilon_{kl}. \quad (3)$$

The Lamé constants λ, μ are related to the Young modulus E and Poisson ratio ν by

$$\lambda = \frac{\nu E}{(1 + \nu)(1 - 2\nu)} \quad (4)$$

and

$$\mu = \frac{E}{2(1 + \nu)}. \quad (5)$$

The equation for equilibrium in the absence of any applied forces is

$$\sigma_{ij,j} = 0 \quad (6)$$

where the subscript $,j$ denotes differentiation with respect to coordinate x_j so this equation states that the divergence of the stress tensor is zero. Anticipating a Finite Element Analysis formulation, we solve the weak form of equation(6) by multiplying with a test function w_i and integrating by parts:

$$0 = \int_{\Omega} w_i \sigma_{ij,j} d^3x = - \int_{\Omega} w_{i,j} \sigma_{ij} d^3x + \int_{\partial\Omega} w_i \sigma_{ij} n_j d^2x \quad (7)$$

where n_j is the j^{th} component of the outward normal to the surface $\partial\Omega$. Substituting in for σ_{ij} from equation(1), we have

$$\int_{\Omega} w_{(i,j)} C_{ijkl} \epsilon_{kl} d^3x = \int_{\Omega} w_{(i,j)} C_{ijkl} \epsilon_{kl}^0 d^3x. \quad (8)$$

where $w_{(i,j)} = \frac{1}{2}(w_{i,j} + w_{j,i})$. Equation (8) naturally lends itself to a finite element discretization.[6]

2.2 Optics

The birefringence magnitude and the slow axis direction for the damaged glass are given by the eigenvalues and the eigenvectors of the indicatrix (\mathbf{B}) of the sample, respectively. (The "slow axis" direction of birefringence is that direction of linear light polarization that experiences the largest index of refraction for light traveling in a given direction.)

As in all electromagnetism problems[5], we begin with the well known Maxwell's equations:

$$\vec{\nabla} \cdot \vec{D} = 0 \quad (9)$$

$$\vec{\nabla} \times \vec{E} = \frac{\partial \vec{B}}{\partial t} \quad (10)$$

$$\vec{\nabla} \cdot \vec{B} = 0 \quad (11)$$

$$\vec{\nabla} \times \vec{H} = -\frac{\partial \vec{D}}{\partial t}. \quad (12)$$

Assuming a linear, nonmagnetic material we also include the constitutive equations:

$$\vec{D} = \varepsilon_0 \varepsilon \vec{E} \quad (13)$$

$$\vec{B} = \mu_0 \vec{H} \quad (14)$$

where ε_0 and μ_0 are the electric permittivity and magnetic permeability of the material, respectively; ε is the relative dielectric tensor (assumed constant with respect to time) and \vec{D} , \vec{E} , \vec{B} , \vec{H} are the electric displacement, electric field, magnetic flux density and the magnetic field vectors, respectively.

Taking the curl of equation (10) and using vector identities together with other definitions given above[5], we obtain the electromagnetic wave equation

$$\vec{\nabla}(\vec{\nabla} \cdot \vec{E}) - \nabla^2 \vec{E} = -\frac{1}{c^2} \varepsilon \frac{\partial^2 \vec{E}}{\partial t^2} \quad (15)$$

where $c = \frac{1}{\sqrt{\varepsilon_0 \mu_0}}$ is the speed of light in vacuum. Assuming a plane wave solution to equation (15) of the form

$$\vec{E} = \vec{E}_0 e^{i(k_0 \hat{k} \cdot \vec{r} - \omega t)} \quad (16)$$

we arrive at the eigenvalue equation governing the birefringence of the sample:

$$\mathbf{B}(I - \hat{k}\hat{k})\vec{E}_0 = \frac{1}{n^2}\vec{E}_0 \quad (17)$$

where $n = \frac{ck_0}{\omega}$ is the refractive index and $\mathbf{B} = \varepsilon^{-1}$ is the indicatrix, which is a symmetric second rank tensor and can be thought of a 3-D ellipsoid. The projection matrix $(I - \hat{k}\hat{k})$ takes the projection of the 3-D ellipsoid which is perpendicular to the wave propagation direction, resulting in an ordinary 2-D ellipse whose semi-axes give the minimum and maximum indices of refraction. (Recall that by the definition of the plane wave of equation (16), \hat{k} is the propagation direction of the wave's phase velocity.)

It is convenient to express the total indicatrix as a sum of an unperturbed starting value (\mathbf{B}_0) and a change in indicatrix ($\Delta\mathbf{B}$) brought about by some kind of influence on the sample, where

$$\mathbf{B} = \mathbf{B}_0 + \Delta\mathbf{B}. \quad (18)$$

In our case the unperturbed (undamaged) glass is isotropic, so

$$\mathbf{B}_0 = \text{diag} \left(\frac{1}{n_0^2}, \frac{1}{n_0^2}, \frac{1}{n_0^2} \right) \quad (19)$$

where n_0 is the index of refraction of the unexposed glass. $\Delta\mathbf{B}$ has many forms depending on the source of the perturbation. In the case of strain-induced birefringence, there is a linear relationship that couples the change in indicatrix and the elastic strain in the material as follows:

$$\Delta B_{ij} = p_{ijkl} \varepsilon_{kl} \quad (20)$$

where \mathbf{p} is a fourth rank tensor and has only two independent values (taken as $p_{1111} = p_{11}$ and $p_{1122} = p_{12}$) for an isotropic material.[4]

3 Algorithm Development

A brief outline of the algorithm is presented, which is followed by a discussion of the problems and the challenges involved at different stages.

1. *A baseline measurement of the birefringence of the glass sample prior to any laser exposure is read in from a data file.*
2. *An experimental file containing the birefringence field of the same sample after laser exposure is then read in. The two fields are aligned so that the baseline can be subtracted from the post-exposure field. After subtraction, the resulting field represents only the laser-induced birefringence.*
3. *The theoretical birefringence field is calculated using a Finite Element code, followed by calculating the eigenvalues of the indicatrix \mathbf{B} . The calculation takes into account the sample boundary conditions, the exposure geometry and a nominal fractional density change (1ppm), making the problem linear in nature. This field is then aligned with the subtracted file calculated above and a best-fit value of the density is deduced to give the best agreement between theory and measurement.*

There are several features that makes this problem mathematically and physically interesting. For example, birefringence is a quantity with both magnitude and direction, but it is not a vector. Performing operations such as subtraction, interpolation, or alignment of such fields requires some care.

The alignment of the fields involves a two-dimensional translation which may be a sub-pixel value. Typically the data sets are on a uniform grid of 0.5 mm spacing, which is a little coarser than some of the features we hope to study. Sub-pixel alignment of data sets requires some kind of interpolation scheme, such as FFTs. Optimizing the alignment with slightly noisy data offers another challenge.

4 Data Fitting

Our data sets are birefringence images on a rectangular grid. At each point we have a magnitude $M : \mathbf{R}^2 \rightarrow \mathbf{Z}^+$, and a slow axis angle $T : \mathbf{R}^2 \rightarrow \{-\frac{\pi}{2}, \frac{\pi}{2}\}$. We take a series of these images. The first is before any damage has been done to the glass block, and we call this the baseline. We then damage the glass block with a laser, and then take another image. The birefringence map measured after laser damage will be called the raw image of birefringence. We run a list of processing steps on the images.

Here M will be the magnitude of the birefringence, and T will be the angle of the slow axis. Since we have multiple pieces of data all around we will use the subscripts b, r, s, m for the baseline, raw, subtracted and model, respectively.

Corners

Part of the rectangular grid is a glass block. We would like to find the corners of the glass block. Luckily the magnitude of the birefringence is very large along the edge of the sample. This is partially due to mechanical damage to the edge of the block, and partially due to the fact that the measuring beam is not fully on the glass block. This is wonderful for us, since we can look for the first and last row and column with large data values. We also do a sanity check to see that the corners we've found make a rectangle.

Alignment

We then need to align the data so that the datasets are lined up. We find a shift that lines up the baseline and the raw data, and shift them so they are on the same grid. Here we assume that the raw data looks like the baseline plus damage (Raw = Baseline + Damage). Any image registration technique would pick up on large features, such as the corners, and the general inhomogeneities in the glass which would not significantly change after the damage.

Subtraction

The data we have are not vector quantities, so we cannot simply subtract the magnitude and the slow axis to get the effect of the damage. Instead we use the maps

$$C_{r,b}^1 = M_{r,b} * \sin(2T_{r,b}) \quad (21)$$

$$C_{r,b}^2 = M_{r,b} * \cos(2T_{r,b}) \quad (22)$$

on each of the images. We subtract these functions and then transform back.

$$C_s^1 = C_r^1 - C_b^1 \quad (23)$$

$$C_s^2 = C_r^2 - C_b^2 \quad (24)$$

$$M_s = \text{sqrt}(C^1)^2 + (C^2)^2 \quad (25)$$

$$T_s = \frac{1}{2} \arctan\left(\frac{C^1}{C^2}\right) \quad (26)$$

Subtracting these quantities and transforming back gives the change in birefringence due to optical changes in the glass induced by the laser exposure.

Finding the damage

Once we have the subtracted data, we can begin to look for the center of damage. We expect the magnitude of the birefringence will be constant on concentric circles around the center of damage, so we are looking to pull out circular features from the magnitude data. We can find these features by convolving the image with annular kernels. We define the kernel K_i as the set of points with radius between r_i and $r_i + \delta r$

$$K_i = \frac{\chi_{B(r_i+\delta r) \setminus B(r_i)}}{r_i} \quad (27)$$

(normalized by the radius r_i) and then look at the set of convolutions of K_i with the subtracted magnitude M_S

$$f_i = M_S * K_i. \quad (28)$$

We first look at the maximum value of f_i for each i

$$\text{argmax}_i \|f_i\|_\infty. \quad (29)$$

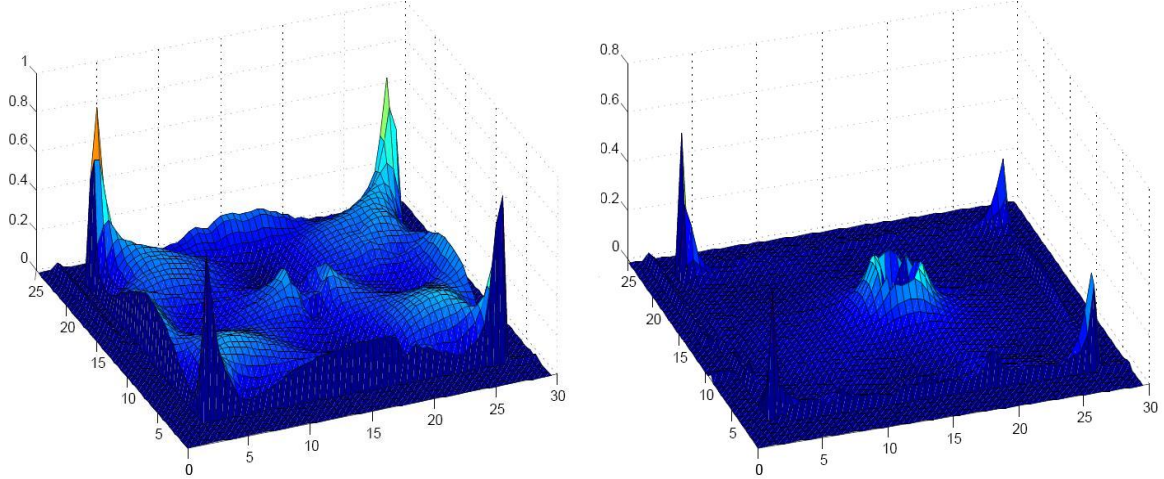


Figure 1: Birefringence maps of (a) Raw data and (b) Raw data minus baseline. The vertical axis is birefringence magnitude.

The corresponding r_i value that gives the highest response is a good estimate for the size of the ring which has the maximum birefringence, and the position of this response gives an estimate of the center of damage

$$(x_0, y_0) = \operatorname{argmax}_{x,y} f_i(x, y) \quad (30)$$

Then we pass this data to the modeling program, which will simulate damage by a laser focused at the center we have found, and the resulting birefringence measurements. We will call these M_m, T_m . The model also gives us a change in density, $\frac{\delta\rho}{\rho}$, which is one of the variables of the glass that we are interested in studying.

5 Theoretical Estimate: Elastic model and Optics

Finite Element Details

The first step towards a finite element version of equation(8) is to rewrite the tensors relations as matrix equations, viz.

$$C = \begin{pmatrix} \lambda + 2\mu & \lambda & \lambda & 0 & 0 & 0 \\ \lambda & \lambda + 2\mu & \lambda & 0 & 0 & 0 \\ \lambda & \lambda & \lambda + 2\mu & 0 & 0 & 0 \\ 0 & 0 & 0 & 2\mu & 0 & 0 \\ 0 & 0 & 0 & 0 & 2\mu & 0 \\ 0 & 0 & 0 & 0 & 0 & 2\mu \end{pmatrix} \quad (31)$$

$$\vec{\epsilon} = (\epsilon_{11} \quad \epsilon_{22} \quad \epsilon_{33} \quad \epsilon_{23} \quad \epsilon_{13} \quad \epsilon_{12})^T, \quad (32)$$

$$\int_{\Omega} \vec{\epsilon}(w)^T C \vec{\epsilon}(u) d^3x = \int_{\Omega} \vec{\epsilon}(w)^T C \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \end{pmatrix}^T \left(-\frac{1}{3} \frac{\delta\rho}{\rho} \right) f(x, y, z) d^3x. \quad (33)$$

The next step is to come up with a basis of interpolants in which to solve equation(8) approximately. Let the domain Ω be composed of a union of N non-overlapping tetrahedra T_k :

$$\Omega = \bigcup_{k=1}^N T_k, T_i \cap T_j = \phi. \quad (34)$$

We define the reference element \hat{T} in the coordinates q_i as

$$\hat{T} = \{\vec{q} \in \mathbf{R}^3 : q_1 \geq 0, q_2 \geq 0, q_3 \geq 0, 1 - q_1 - q_2 - q_3 \geq 0\}. \quad (35)$$

Let the coordinates of the four vertices of T_k be $\vec{r}_1^k, \vec{r}_2^k, \vec{r}_3^k, \vec{r}_4^k$. For T_k , we may map between the reference coordinates q_i and the physical coordinates x_i by means of

$$\vec{x}(\vec{q}) = J_k \vec{q} + \vec{r}_4^k \quad (36)$$

$$\vec{q}(\vec{x}) = J_k^{-1}(\vec{x} - \vec{r}_4^k) \quad (37)$$

where the Jacobian J_k is given by

$$J_k = \begin{pmatrix} \vec{r}_1^k - \vec{r}_4^k & \vec{r}_2^k - \vec{r}_4^k & \vec{r}_3^k - \vec{r}_4^k \end{pmatrix}. \quad (38)$$

The interpolants are defined in the reference coordinates by

$$\vec{h}(\vec{q}) = H \vec{q} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}, \quad (39)$$

where

$$H = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1 & -1 & -1 \end{pmatrix}. \quad (40)$$

We will need derivatives of these interpolants in the physical coordinates x_i , to be computed by the chain rule:

$$\begin{aligned} \nabla \vec{h}(\vec{q}(\vec{x})) &= (\nabla \vec{h})(\vec{q}(\vec{x})) \nabla \vec{q}(\vec{x}) \\ &= H J^{-1}. \end{aligned} \quad (41)$$

So the interpolant derivatives, and hence the strain, are constant on each tetrahedron T_k .

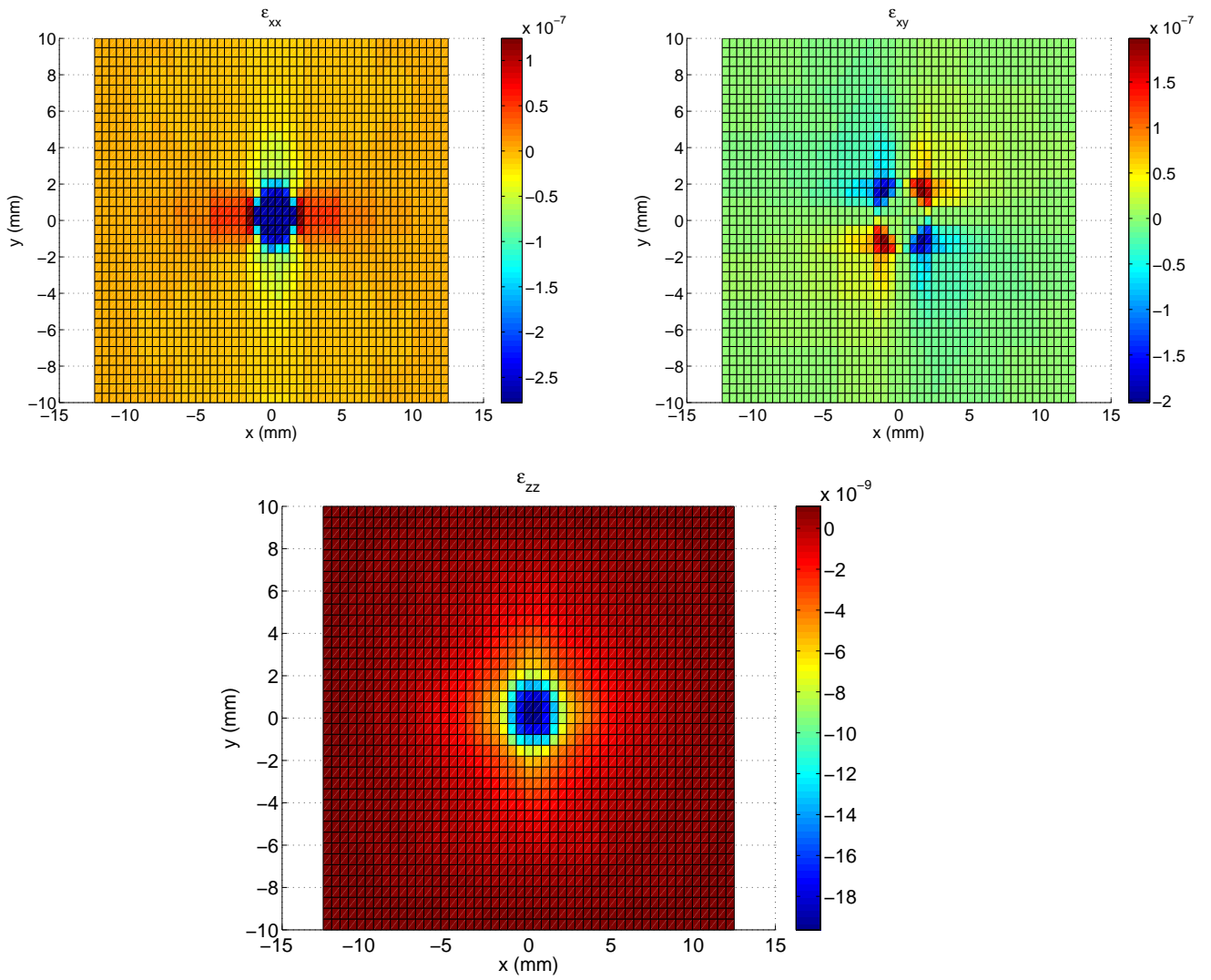


Figure 2: Plots of the strain components using our Finite Element Model

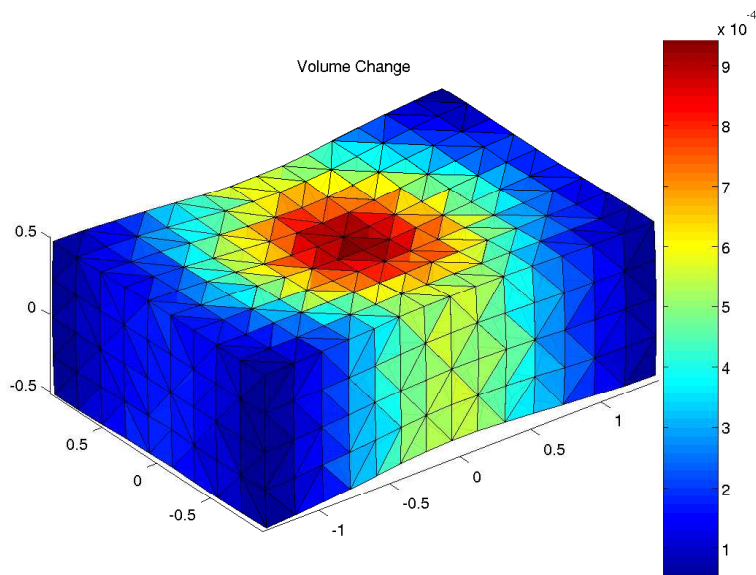


Figure 3: Coarse run output the finite element model

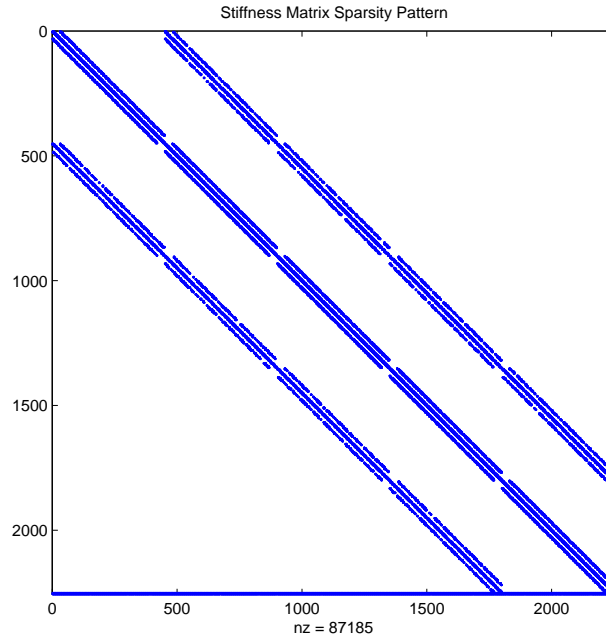


Figure 4: Structure of the stiffness matrix

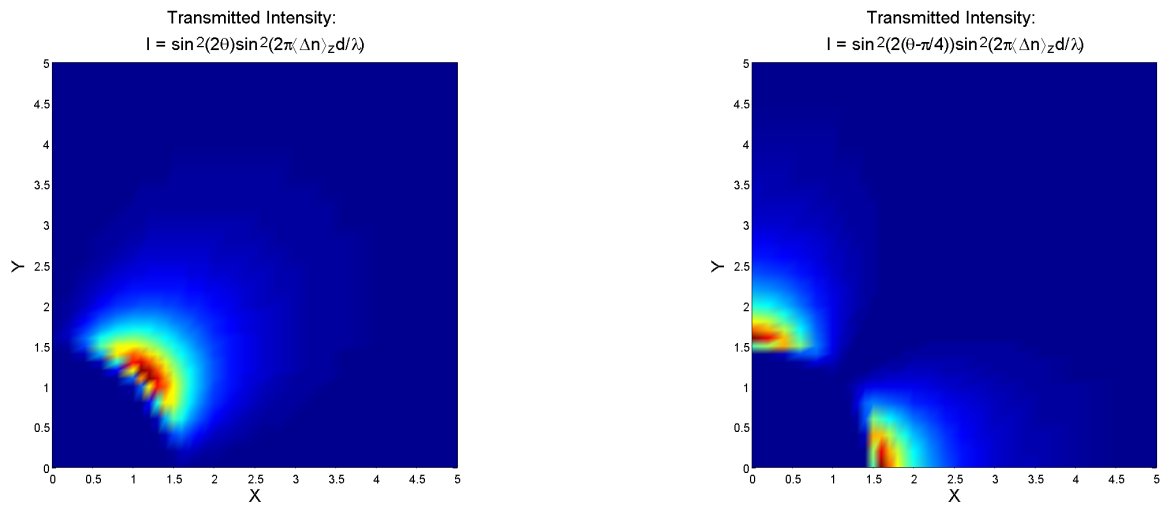


Figure 5: Model intensity maps (first quadrant) of the damaged glass sample when the polarimeter is (a) aligned with the x-axis (b) at an angle of 45° with the x-axis

Skipping some of the nastier details, the final form for the local stiffness matrix and forcing vector are given by

$$K_k = \int_{T_k} D^T C D d^3x \quad (42)$$

$$f_k = \int_{T_k} D^T C \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \left(\frac{-1}{3}\right) \frac{\delta \rho}{\rho} f(\vec{x}) d^3x, \quad (43)$$

where

$$D = \begin{pmatrix} \partial_1 \vec{h}^T & 0 & 0 \\ 0 & \partial_2 \vec{h}^T & 0 \\ 0 & 0 & \partial_3 \vec{h}^T \\ 0 & \frac{1}{2} \partial_3 \vec{h}^T & \frac{1}{2} \partial_2 \vec{h}^T \\ \frac{1}{2} \partial_3 \vec{h}^T & 0 & \frac{1}{2} \partial_1 \vec{h}^T \\ \frac{1}{2} \partial_2 \vec{h}^T & \frac{1}{2} \partial_1 \vec{h}^T & 0 \end{pmatrix} \quad (44)$$

So the pieces of the rows of D are just columns of $\nabla \vec{h}(\vec{q}(\vec{x}))$. The other issue that crops up is the computation of the constraints for the elimination of rigid-body modes. The idea is to discretize the equations

$$\int_{\Omega} \vec{u}(\vec{x}) d^3x = 0 \quad (45)$$

$$\int_{\Omega} \vec{u}(\vec{x}) \times \vec{x} d^3x = 0. \quad (46)$$

The result is six additional constraints, which are added on to K in the code as rows. This means that instead of solving a linear system, we have a linear least squares problem

$$\min_{q \in \mathbf{R}} \|Kq - f\|_2^2. \quad (47)$$

Calculation of Line Integrals

We did not emphasize this in the sections on birefringence measurement and theory, but the measurement entails an accumulation of birefringence along a path through the sample from front to back. Hence the theoretical model for birefringence needs a set of line integrals along the light path, through the sample, for all the strain components (or equivalently all the indicatrix components) that will be needed in the birefringence calculation. Calculation of the line integrals needed for the optical considerations was tricky, particularly as this calculation can become needlessly time-consuming if not designed well. The idea is to build a list of faces and then a list of tetrahedra in terms of those faces (instead of directly in terms of the vertices). Then, we calculate the intersection points between faces and the

lines. The code need then only go through all the elements and find the lengths of the relevant intersections, as the strain is constant on each T_k . We refer the interested reader to the Matlab code for details.

Sparse Least Squares Problem

To solve (47), we have used Matlab's function *lsqr*, which solves the normal equations iteratively. This process would benefit greatly from a preconditioner akin to that discussed in Benzi and Tuma[7], as the convergence is slow for large problems because the condition number of K gets worse. We have implemented such a thing, though it is done in Matlab and is too slow to be practically useful. Another thought here is that we might do well to just pin a few points on the boundary faces in order to remove the extra degrees of freedom rather than go to all this trouble - Matlab has very good approximate factorization routines for linear systems (*cholinc*) and permutation algorithms which preserve sparsity (*symamd*), so the preconditioning could be faster in that case. An added plus of solving the linear system instead of the least squares problem is that the condition number would be much better. That is, $cond(A^T A) = cond(A)^2$, so by solving the linear system of equations obtained after pinning a few points we would roughly double the number of correct digits in the answer. (The least squares problem has roughly the square of the condition number of the linear problem, and the computer solution loses roughly $\log_{10}(cond(A))$ decimal digits to machine precision error.) If this procedure were adopted then the choice of how to pin certain degrees of freedom would have to be made carefully to avoid introducing unphysical artifacts into the solution.

6 Convolution technique

Once the birefringence map is computed, it is necessary to take into account the fact that the laser beam used for measurement purposes has a radius that is not infinitesimally small to collect data at exactly one point of interest. To compensate this situation, the birefringence map is convolved with the 2-D Gaussian with $xmean=(M/2) * \delta_x$ and x -standard deviation= $FWHMx/(2 * sqrt(2 * log(2)))$ where M is the integer part of $5 * sx/\delta_x$ and $FWHMx$ is the Full-Width Half-Maximum for the x variable, and analogous definitions for y mean and y -standard deviation. The values δ_x and δ_y depend on the desired spatial separation for the theoretical output. The result of this operation is a smoothed map, due to the bell shape of the Gaussian, as can be appreciated in figure 6.

7 Matching data to the model: best fit approximation

Roughly speaking the model data should be a scaled version of the subtracted data, $M_m \approx lM_s$ and the angles of the slow axis should be roughly the same $T_m \approx T_s$. We are very interested in obtaining l , which could be roughly computed by setting $l \approx \frac{\|M_m\|}{\|M_s\|}$, for an appropriate choice of norm. We are assuming that the change in birefringence is linearly related to the change in density, so the change in density of our sample would be $l \frac{\delta \rho}{\rho}$. Scaling

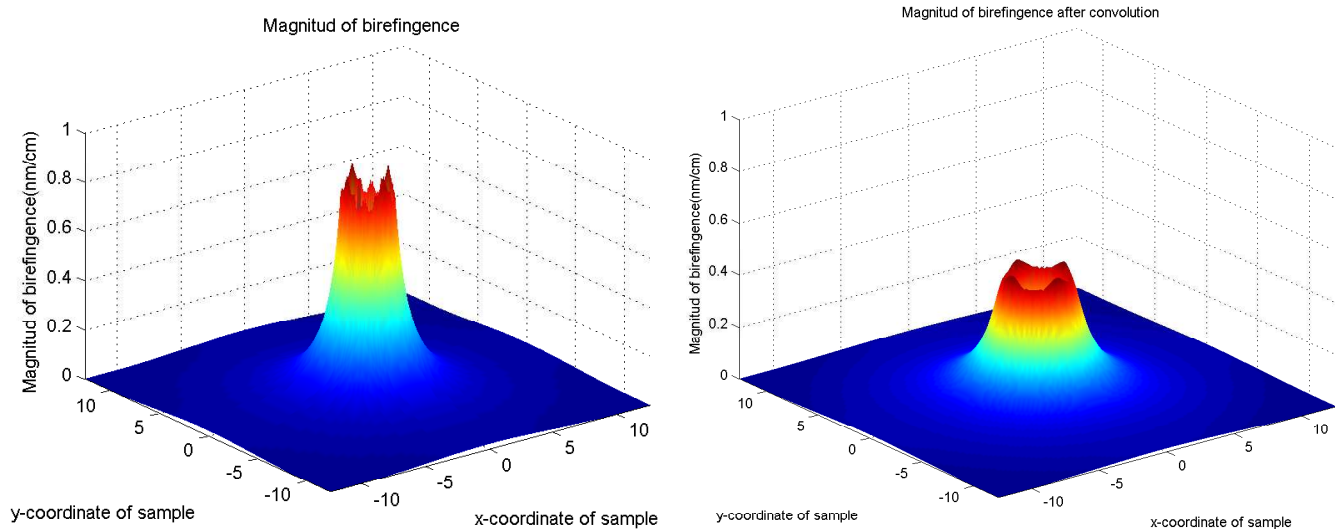


Figure 6: Model birefringence maps of (a) before convolution (b) after convolution

the sample data, and subtracting it from the data from the model, $||M_m - lM_s||$ will give an idea of how well our process matches up with the theoretical results.

8 Graphic User Interface

Up to this point, we have mentioned the experimental and theoretical aspects of the proposed problem. However for there to be useful and simple implementation of our numerical analysis, we needed to develop a graphical user interface (GUI) that allowed easy manipulation of these techniques. This GUI was developed using Matlab using the GUIDE application. The GUI that we developed is shown in Figure 7, along with the descriptions of all components of the application. Figure 7 points out that there are several areas of concern in the GUI to adequately communicate all necessary information between the user and the analysis program. We will discuss each part individually and briefly describe its importance.

Project File Interface

The first part of the GUI that we will discuss is the one which deals with loading the vital information into the GUI such as property values and file information. We have designed this portion of the program to not limit any of the program variables that might be introduced into the program. This is important for future expansions of the code that might be introduced. In order to allow this ability of the code, some communication formatting must be used. In the below table there are a list of commands that would be typed into the property file (**test.proj**) and to the right their entry into the data structure (**C**) used to store information into the application memory. The notation used is identical to that which would be used in a Matlab code to locate that information. What we can see from the examples in Table 1 is that the interpretation of the line entry is very general, but must have a specified format to

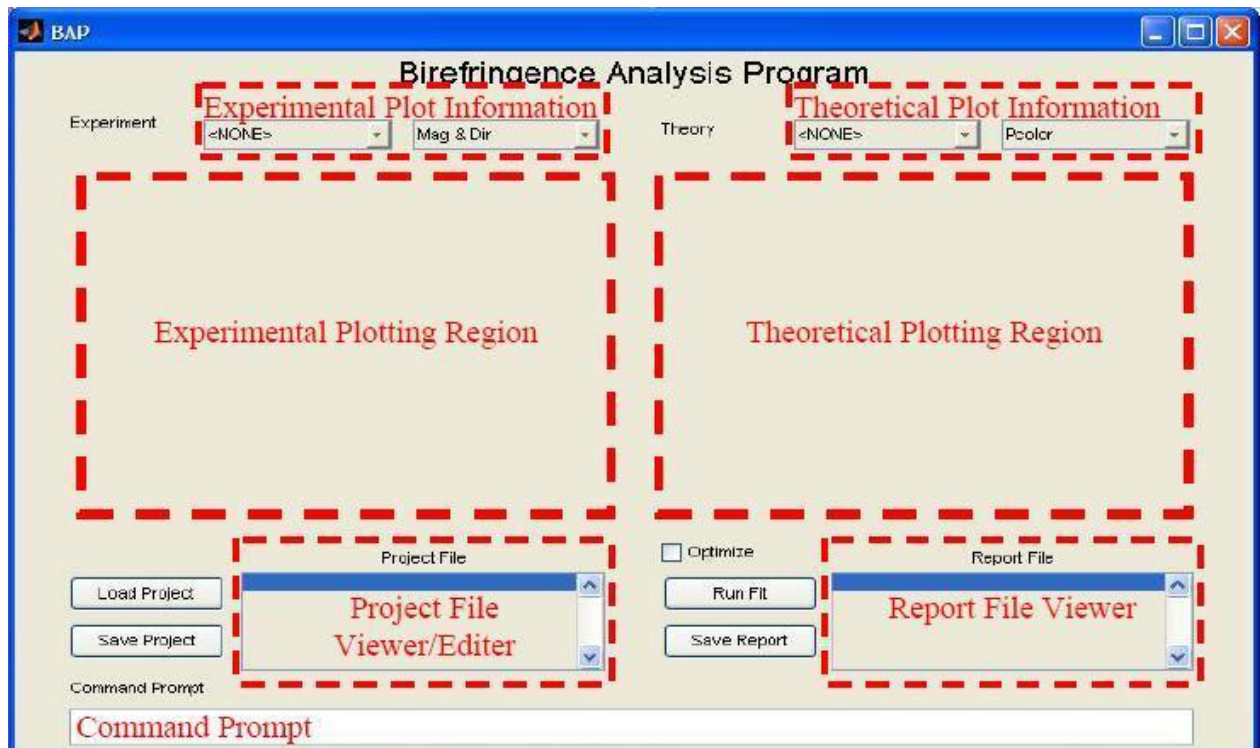


Figure 7: Screen view of the GUI along with the general functional regions highlighted in red boxes. The top two red boxes indicate where plot information is chosen for experimental and theoretical comparisons. These boxes decide what graphs will be displayed in the plotting regions below the information. Other important regions are the project file and report file viewer boxes and the command prompt described at the bottom.

correctly implement the line. All line entries begin with the name of the variable (NAME), followed by an underscore, and ended with the value for that variable name (Value). This underscore is an essential divider that separates the data into its two components NAME and Value. If the value can be converted to a number or number array then the NAME will be attached to the C structure as shown in the first four examples. However if the value is a string, (i.e. can't be converted to a number) then the NAME is converted to a structure whose first element denoted by 1 and whose value is the string supplied. Lines that cannot be broken into the NAME, Value format will not be inserted into the structure as in the sixth example in Table 1.

Table 1.

(test.proj) line entry	C structure location	Value in C structure
L-100	C.L	100
WL-632.8	C.WL	632.8
dRhoRho 1e-7	C.dRho	$1e - 7$
DIM-[50 40]	C.DIM	50, 40
PROJ-test.proj	C.PROJ1	test.proj
PROPERTYINFO	No Entry	No Entry
PROPERTYINFO -comment	C. PROPERTYINFO1	comment

In the Figure, we have two buttons that execute commands dealing with the project file and they have the titles, 'Load Project' and 'Save Project' which should be self explanatory. If you should choose the 'Save Project' button then the file will be saved to the location listed in the PROJ command in the project file as on line five in Table 1. The final functionality that should be mentioned is that if one should click any line in the 'Project File Viewer/Editor' a prompt will come up asking if a property value would like to be changed. If the answer is yes type 'Y'. If you type yes, then enter the new value and the line will be updated.

Report File Interface

In this section, we have four components: 'Run Fit' button, 'Optimize' tab, 'Save Report' button, and the 'Report File Viewer'. The 'Run Fit' button initiates the data fitting if the theory simulation is completed or computes the simulation first otherwise. At the end of the data fitting, then a report is generated which can be viewed in the 'Report File Viewer'. By selecting the 'Save Report' button all data, report and project files, and simulation results will be saved to their respective locations. The 'Optimize' tab allows one to select between a very simple data fitting process which calculated the density change only, or an optimized fit that can vary multiple parameters to fit the data.

Theory and Experimental Plot Controls

These controls allow one to view and compare theoretical and experimental data in various ways such as magnitude and slow axis plots, surface plots of the magnitude, or even intensity profiles at different polarizer and analyzer orientations. An example of the intensity plots produced by the software is shown in Figure 5, where we show only the first quadrant of the intensity map for a case with fourfold symmetry, based on Finite Element Analysis model calculations.

9 Improvements and additional features

Our 10 days of effort during the IMA Workshop did not allow us to include a number of features that would make this birefringence analysis package more powerful. For example, in some cases a sample may be damaged in multiple locations, and additional code features could facilitate analysis of multiple-damage spots in terms of data subtraction and identification of the damage location on the sample. In addition, it is typically the case that multiple exposures are performed on the same damage site, with birefringence measurements at each level of exposure. These birefringence maps should share a common damage location relative to the sample edges, even though the individual data sets do not share a common origin. The birefringence analysis would be improved by reading all the relevant data sets at once and finding an averaged damage location. We also did not yet implement an algorithm to line up the data sets with sub-pixel resolution.

10 Acknowledgement

We acknowledge the hospitality of the Institute for Mathematics and its Application at the University of Minnesota during the Workshop on Mathematical Modeling in Industry, and Corning Incorporated for providing example data files for analysis.

References

- [1] N.F. Borrelli, C. Smith, D.C. Allan and T.P. Seward III, *J. Opt. Soc. Am. B* **14** (7) (1997) 1606-1615.
- [2] J.L. Ladison, J.F. Ellison, D.C. Allan, D.R. Fladd, A.W. Fanning and R. Priestly, *SPIE* **4346** (2001) 1416-1423.
- [3] U. Neukirch, D.C. Allan, N.F. Borrelli, C.E. Heckle, M. Mlejnek, J. Moll and C. Smith, *SPIE* **5754**, (2005) 638-645.
- [4] J. F. Nye, "Physical Properties of Crystals", Oxford University Press, NY, 1985.
- [5] M. Born and E. Wolf, "Principles of Optics: Electromagnetic Theory of Propagation, Interference and Diffraction of Light", 7th Edition, Cambridge University Press, New York, 1999.

- [6] T. J. R. Hughes, “The Finite Element Method”, Dover Publications Inc., Mineola, NY, 2000.
- [7] M. Benzi and M. Tuma, “A robust preconditioner with low memory requirements for large sparse least squares problems”, *SIAM J. Sci. Comput.* **25** (2) (2003) 499-512.