

LASER INTERFEROMETER GRAVITATIONAL WAVE OBSERVATORY
- LIGO -
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
CALIFORNIA INSTITUTE OF TECHNOLOGY

| | | |
|--|-----------------------|------------|
| Technical Note | LIGO-T020001-R | 2002/03/14 |
| A Thermo-mechanical Core Optic Finite Element Model | | |
| Ryan Lawrence | | |

Distribution of this draft:

xyz

This is an internal working note of the
LIGO Laboratory and the
LIGO Scientific Collaboration.

California Institute of Technology
LIGO Project - MS 51-33
Pasadena CA 91125
Phone (626) 395-2129
Fax (626) 304-9834
E-mail: info@ligo.caltech.edu

Massachusetts Institute of Technology
LIGO Project - MIT Room NW17-161
Cambridge, MA 01239
Phone (617) 253-4824
Fax (617) 253-7014
E-mail: info@ligo.mit.edu

WWW: <http://www.ligo.caltech.edu/>

1 Table of Contents

Contents

| | | |
|----------|---|-----------|
| 1 | Table of Contents | 2 |
| 2 | Introduction | 3 |
| 2.1 | Running the program | 4 |
| 3 | The Thermal Model | 4 |
| 3.1 | Boundary Conditions | 5 |
| 3.2 | Symmetry Considerations | 5 |
| 3.3 | Calculating the Thermal Lenses | 5 |
| 3.3.1 | Matrix Elements | 6 |
| 4 | The Mechanical Model | 7 |
| 4.1 | Boundary Conditions | 8 |
| 4.2 | Symmetry Considerations | 9 |
| 4.3 | Extracting the Surface Distortion | 9 |
| 4.3.1 | Matrix Elements | 9 |
| 5 | Tests Against Analytical Models | 10 |
| 5.1 | Case 1: Large beam, fused silica optic. | 11 |
| 5.2 | Case 2: Large beam, sapphire optic. | 14 |
| 5.3 | Case 3: Small beam, sapphire optic. | 17 |
| | Appendix A: Notation | 21 |
| | Appendix B: Description of the M-files | 22 |
| .1 | beam.m | 22 |
| .2 | calculate_matrices.m | 22 |
| .3 | find_distortions.m | 22 |
| .4 | get_variable.m | 23 |
| .5 | numerical_optic.m | 23 |
| .6 | optic_params.m | 24 |
| .7 | numerical_optic_plot.m | 24 |

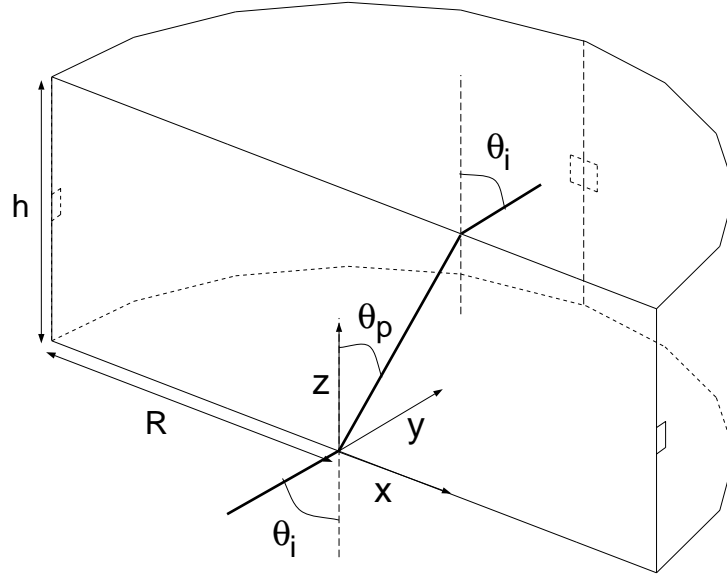


Figure 1: Geometry of the finite element model.

2 Introduction

The Thermo-mechanical Core Optic Finite Element Model is a set of Matlab scripts which simulate arbitrarily angled Gaussian beam heating in both transmission and reflection on a cylindrical optic suspended in a vacuum. The optical path distortion due to thermal lensing is calculated for transmitted beams, as well as the thermo-mechanical distortion of all surfaces. The size of the heating beam, the radius and height of the optic, and all the relevant material parameters for the optic (including anisotropic thermal conductivity, anisotropic thermal expansion, and anisotropic elastic moduli) are input by the user. While the scripts run from the Matlab command prompt, they do however require the FEMLAB add-on for Matlab, available with a substantial educational discount from Comsol Inc. (see <http://www.femlab.com>).

In the model, we define the optic as a cylinder of radius R and height h with index of refraction n . The origin of the Cartesian coordinates (x, y, z) lies on the center of the optic's face, with $+z$ normal to the surface and pointing into the optic's bulk. The heating beam is incident on the optic at an angle θ_i with respect to z in the $(x, y = 0, z)$ plane, and propagates at an angle $\theta_p = \arcsin\left(\frac{\sin \theta_i}{n}\right)$ with respect to $-z$ through the bulk. See figure 1 for a diagram of the geometry.

We consider the thermal load of a Gaussian beam in three separate cases:

1. Absorption in the coating at $z = 0$ (termed in the code as "Center Surface Absorption", since the heating beam is centered on the optic's face here),
2. Absorption in the coating at $z = h$ (termed in the code as "Offcenter Surface Absorption", since the center of the heating beam at $z = h$ is, in general, away from the center of the optic), or
3. Bulk absorption along the heating beam's path through the optic,

The model first solves the heat equation (equations 1 and 3) to find the temperature field $T(x, y, x)$. Then, using the derived temperature field to define the thermo-mechanical forces acting within the

Table 1: List of required M-files.

| File Name | Description |
|-------------------------------------|---|
| <code>optic_params.m</code> | Material and Geometry Parameters (script) |
| <code>numerical_optic.m</code> | The model engine (script) |
| <code>numerical_optic_plot.m</code> | Plots the solution in 3D (function) |
| <code>beam.m</code> | Gaussian-profiled thermal load (function) |
| <code>calculate_matrices.m</code> | Calculates the scattering matrices and saves to file (function) |
| <code>find_distortions.m</code> | Interpolates to find raw thermal distortions and saves to file (function) |
| <code>get_variable.m</code> | Retrieves variables stored in a fem object (function) |

See Appendix A for detailed descriptions of each of these subroutines. All software is available at <http://pendragon.mit.edu:8080/Beamsplitter/>.

`optic`, the model solves the stress-strain problem (equations 8 and 9) to find the vector mechanical distortion over the entire volume $\vec{u}(x, y, z)$. Thermal lenses along the heating beam’s path, as well as the “conjugate optical path” (the heating beam axis reflected through the $(x = 0, y, z)$ plane) are found by interpolating T to a rectangular grid, integrating over the desired optical axis, then converting to coordinates perpendicular to the vacuum optical axis. The raw normal direction thermal distortion for each face ($u_z|_{z=0}$ and $u_z|_{z=h}$) are centered on the location of the optical axis at each face, interpolated to a rectangular grid, and converted to coordinates perpendicular to the vacuum optical axis (x', y', z') .

2.1 Running the program

Table 1 lists the Matlab M-files required to run the model. The script `optic_params.m` defines the required geometry, material parameters, and location of absorption to solve for. The script `numerical_optic.m` is run from the Matlab command prompt to solve the model for the parameters defined in `optic_params.m`. All other m-files are ancillary files that need not be altered or run at the command prompt, unless customization of the finite element model is desired. The purpose and usage of all m-files is described in detail in appendix B.

3 The Thermal Model

The steady-state heat transfer equation for an arbitrary body in 3 dimensions is given by the Fourier equation:

$$-\vec{\nabla} \cdot (\underline{K} \vec{\nabla}) T(\vec{r}, t) = H_b(\vec{r}, t) \quad (1)$$

Where c is the heat capacity, ρ is the density, \underline{K} is the thermal conductivity (in general, a 3×3 matrix), T is the local temperature, and H_b is the rate of bulk heating (power per unit volume).

For the case of uniform bulk absorption of a unit power Gaussian beam of waist radius w

incident on the optic at an angle of θ_i , we have:

$$\begin{aligned} H_b(x, y, z) &= \frac{\cos \theta_p \cos \theta_i}{h} \frac{2}{\pi w^2} \exp \left(-2 \frac{(x \cos \theta_i - z \tan \theta_p)^2 + y^2}{w^2} \right) \\ &= \text{beam} \left(\frac{\cos \theta_p \cos \theta_i}{h}, w, \sqrt{(x \cos \theta_i - z \tan \theta_p)^2 + y^2} \right) \end{aligned}$$

where we have defined:

$$\text{beam}(\alpha, w, r) \equiv \frac{2\alpha}{\pi w^2} \exp \left(-2 \frac{r^2}{w^2} \right). \quad (2)$$

3.1 Boundary Conditions

If the body interacts only with a thermal radiation bath of temperature T_∞ (i.e. it is suspended in a vacuum enclosure with walls at temperature T_∞), the boundary equations are given by the Stefan-Boltzmann law:

$$\hat{n} \cdot (\underline{K} \vec{\nabla} T) = -\sigma \epsilon (T - T_\infty)^4 + H_s(\vec{r}, t)$$

Where \hat{n} is the unit normal on the surface, ϵ is the emissivity, σ is the Stefan-Boltzmann constant, and H_s is the rate of surface heating due to an external source (power per unit area). If we assume that $\frac{T - T_\infty}{T_\infty} \ll 1$, we may linearize the boundary conditions so that:

$$\hat{n} \cdot (\underline{K} \vec{\nabla} T) = -4\sigma \epsilon T_\infty^3 (T - T_\infty) + H_s(\vec{r}, t) \quad (3)$$

For the case of uniform surface absorption of a unit power Gaussian beam of waist radius w incident on the optic at an angle of θ_i , we have:

$$\begin{aligned} H_s(x, y, z) &= \frac{2 \cos \theta_i}{\pi w^2} \exp \left(-2 \frac{(x \cos \theta_i - z \tan \theta_p)^2 + y^2}{w^2} \right) \\ &= \text{beam} \left(\cos(\theta_i), w, \sqrt{(x \cos \theta_i - z \tan \theta_p)^2 + y^2} \right) \end{aligned} \quad (4)$$

3.2 Symmetry Considerations

To lessen the time necessary for computation, we can utilize symmetry to simplify the model. Since the heating beam is identical under reflection through the $(x, y = 0, z)$ plane, the solution will be identical under reflection through this plane. In other words, we have the additional boundary condition:

$$\left. \frac{\partial T}{\partial y} \right|_{y=0} = 0$$

and we may save computation time by modeling only half of the optic.

3.3 Calculating the Thermal Lenses

For each case of absorption (center surface, offcenter surface, and bulk), there are 2 relevant thermal lenses which are calculated:

1. ϕ_i , thermal lensing along the main (heating) beam path, defined as:

$$\phi_i(x', y') = \frac{dn}{dT} \int_0^h T((x - z \tan(\theta_p)) \cos(\theta_i), y, z) dz / \cos(\theta_p)$$

2. ϕ_o , lensing along the conjugate beam path, defined as:

$$\phi_o(x', y') = \frac{dn}{dT} \int_0^h T((x + z \tan(\theta_p)) \cos(\theta_i), y, z) dz / \cos(\theta_p)$$

3.3.1 Matrix Elements

If a beam of wavelength λ and modal content \mathbf{k} , represented as a column vector in some basis $\{|k\rangle\}$ (e.g., Hermite-Gauss) is propagated through a given optical path distortion $\phi(x', y')$, we may calculate the resulting modal content \mathbf{k}' in the original basis via the *scattering matrix* \underline{S} :

$$\mathbf{k}' = \underline{S}\mathbf{k}$$

where the components of \underline{S} are given by:

$$\underline{S}(k, k') = \langle k | \exp\left(\frac{2\pi i}{\lambda}\phi\right) | k' \rangle$$

Calculating this matrix is, in general, computationally intensive. To save ourselves time down the road, use the identity:

$$\underline{S} = \text{Exp}m(i\phi) \quad (5)$$

where $\text{Exp}m$ is the matrix exponential, and $\underline{\phi}$ is the *source matrix* given by:

$$\underline{\phi}(k, k') = \langle k | \frac{2\pi}{\lambda}\phi | k' \rangle$$

Thus, for an absorbed power P_a absorbed in the region represented by the distortion ϕ , the scattering matrix is calculated as:

$$\underline{S} = \text{Exp}m(iP_a\underline{\phi})$$

In this model, we calculate the source matrices only, in the Hermite-Gauss basis, using the Melody utility function `hgme.m`. More precisely, for each of the 3 cases of absorption:

1. Bulk Absorption

- $\text{Phi}_{s_i}(k', k) = \langle k' | \frac{2\pi}{\lambda}\phi_i | k \rangle$
- $\text{Phi}_{s_o}(k', k) = \langle k' | \frac{2\pi}{\lambda}\phi_o | k \rangle$

2. Center Surface Absorption

- $\text{Phi}_{c_f}(k', k) = \langle k' | \frac{2\pi}{\lambda}\phi_i | k \rangle$

3. Offcenter Surface Absorption

- $\text{Phi}_{c_ib}(k', k) = \langle k' | \frac{2\pi}{\lambda}\phi_i | k \rangle$
- $\text{Phi}_{c_ob}(k', k) = \langle k' | \frac{2\pi}{\lambda}\phi_o | k \rangle$

4 The Mechanical Model

In the limit of small distortions $\mathbf{u} = \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix}$, strains in a material are completely described by local derivatives of the deformation components:

$$\epsilon_{ij} = \frac{1}{2} \left(\frac{\partial u_j}{\partial x_i} + \frac{\partial u_i}{\partial x_j} \right)$$

The 3×3 strain tensor is symmetric, so we may define a 6-element strain vector as:

$$\vec{\epsilon} \equiv (\epsilon_{xx}, \epsilon_{yy}, \epsilon_{zz}, 2\epsilon_{xz}, 2\epsilon_{yz}, 2\epsilon_{xy}).$$

Similarly, we define the 6-element stress vector as:

$$\vec{\sigma} \equiv (\sigma_{xx}, \sigma_{yy}, \sigma_{zz}, \sigma_{xz}, \sigma_{yz}, \sigma_{xy})$$

and use Hooke's law to relate stresses (forces) to strains (physical displacements) via the elastic modulus matrix \underline{c} :

$$\vec{\sigma} = \underline{c}\vec{\epsilon} - (T - T_\infty)\vec{\gamma} \quad (6)$$

where $\vec{\gamma}$ is the stress temperature modulus vector, related to the thermal expansion coefficient vector $\vec{\alpha} = (\alpha_x, \alpha_y, \alpha_z, 0, 0, 0)$ by:

$$\vec{\gamma} = \vec{\alpha}\underline{c}$$

and the elastic modulus matrix \underline{c} has elements:

$$\underline{c} \equiv \begin{pmatrix} c_{11} & c_{12} & c_{13} & 0 & 0 & 0 \\ c_{12} & c_{22} & c_{23} & 0 & 0 & 0 \\ c_{13} & c_{23} & c_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & c_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & c_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & c_{66} \end{pmatrix}$$

For an isotropic material with Young's modulus E and Poisson ratio ν , we have:

$$c_{11} = c_{22} = c_{33} = \frac{E(1 - \nu)}{(1 + \nu)(1 - 2\nu)}$$

$$c_{12} = c_{13} = c_{23} = \frac{E\nu}{(1 + \nu)(1 - 2\nu)}$$

$$c_{44} = c_{55} = c_{66} = \frac{E}{2(1 + \nu)}$$

To solve for the displacements \mathbf{u} , we use the stress equilibrium conditions in the steady state (i.e., the net force in every direction must vanish):

$$\frac{\partial \sigma_{ii}}{\partial x_i} + \frac{\partial \sigma_{ji}}{\partial x_j} + \frac{\partial \sigma_{ki}}{\partial x_k} = 0 \quad (7)$$

Inserting equation 6 into equation 7 and simplifying, we find the differential equation governing $\mathbf{u}(x, y, z)$:

$$\vec{\nabla} \cdot (\underline{\mathbf{C}} \otimes \vec{\nabla}) \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix} = \begin{pmatrix} \gamma_x \frac{\partial T}{\partial x} \\ \gamma_y \frac{\partial T}{\partial y} \\ \gamma_z \frac{\partial T}{\partial z} \end{pmatrix} \quad (8)$$

where the rank four tensor $\underline{\mathbf{C}}$ has elements:

$$\underline{\mathbf{C}} = \begin{pmatrix} \begin{pmatrix} c_{11} & 0 & 0 \\ 0 & c_{66} & 0 \\ 0 & 0 & c_{44} \end{pmatrix} & \begin{pmatrix} 0 & c_{12} & 0 \\ c_{66} & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & \begin{pmatrix} 0 & 0 & c_{13} \\ 0 & 0 & 0 \\ c_{44} & 0 & 0 \end{pmatrix} \\ \begin{pmatrix} 0 & c_{66} & 0 \\ c_{12} & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & \begin{pmatrix} c_{66} & 0 & 0 \\ 0 & c_{22} & 0 \\ 0 & 0 & c_{55} \end{pmatrix} & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & c_{23} \\ 0 & c_{55} & 0 \end{pmatrix} \\ \begin{pmatrix} 0 & 0 & c_{44} \\ 0 & 0 & 0 \\ c_{13} & 0 & 0 \end{pmatrix} & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & c_{55} \\ 0 & c_{23} & 0 \end{pmatrix} & \begin{pmatrix} c_{44} & 0 & 0 \\ 0 & c_{55} & 0 \\ 0 & 0 & c_{33} \end{pmatrix} \end{pmatrix}$$

4.1 Boundary Conditions

The normal forces on the boundaries must vanish, as well as components of shear that are parallel to the surface. This is equivalent to the Neumann boundary condition:

$$\vec{n} \cdot (\underline{\mathbf{C}} \otimes \vec{\nabla}) \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix} = (T - T_\infty) \begin{pmatrix} \gamma_x n_x \\ \gamma_y n_y \\ \gamma_z n_z \end{pmatrix} \quad (9)$$

where n_i is the component of the surface normal in the i th direction.

In general, the sum of all effective body forces ($\vec{\gamma} \cdot \vec{\nabla} T$) and effective boundary forces ($\gamma_n (T - T_\infty)$) is not identically zero when evaluated over the finite element mesh (although it might be very close to zero). Thus, to keep the model from “flying away” (i.e., it will iterate itself far away from of the original origin of coordinates), we must anchor the model at some point. To do this, we pick a small square patch of edge length $h/10$ on the optic’s edge, centered at $(x = R, y = 0, z = h/2)$, and impose the constraint $u_x = u_z = 0$. This anchors the optic in x and z directions. To anchor the y direction, we use the symmetry discussed below. To eliminate torques about the patch at $(x = R, y = 0, z = h/2)$, we impose the constraint $u_z = 0$ on identically sized patches at $(x = 0, y = R, z = h/2)$ and $(x = -R, y = 0, z = h/2)$. Refer to figure 1 for the locations of the constrained patches.

To minimize the effect of these mechanical constraints, as well as maximizing the accuracy of the computed solution, we compute the average temperature on the spatially constrained patch at $(x = R, y = 0, z = h/2)$ and use this temperature in equation 9 in place of T_∞ . The net temperature increase on and around the patch is then near zero, and the patch does not constrain expansion of the surfaces. This technique utilizes the fact that the equations governing thermal expansion (equations 8 and 9) are linear, as well as the fact that a DC temperature increase has the trivial solution of uniform thermal expansion. Thus, we may subtract a constant temperature off of the thermal load in equation 9 with no effect on the spatial profile of the resulting mechanical distortion.

4.2 Symmetry Considerations

Similarly as discussed in section 3.2, the solution $\vec{u}(x, y, z)$ will be symmetric about the $(x, y = 0, z)$ plane. So, we have the additional boundary condition:

$$u_y|_{y=0} = 0$$

and we escape with modeling only half of the optic.

4.3 Extracting the Surface Distortion

For each case of absorption (center surface, offcenter surface, and bulk), there are three relevant surface distortions which are extracted:

1. `psi`, the distortion at $z = 0$, defined as:

$$\psi(x', y') = u_z(x \cos(\theta_i), y, z = 0)$$

2. `psi_ib`, the distortion at $z = h$ centered on the main path's optical axis, defined as:

$$\psi_{ib}(x', y') = u_z((x - h \tan(\theta_p)) \cos(\theta_i), y, z = h)$$

3. `psi_ob`, the distortion at $z = h$ centered on the conjugate path's optical axis, defined as:

$$\psi_{ob}(x', y') = u_z((x + h \tan(\theta_p)) \cos(\theta_i), y, z = h)$$

4.3.1 Matrix Elements

The source matrix elements calculated in the Hermite-Gauss basis $|k\rangle$, for each of the 3 cases are:

1. Bulk Absorption

- $\text{Upsilon}_{s_i_f}(k', k) = \langle k' | \frac{2\pi}{\lambda} \text{psi} | k \rangle$
- $\text{Upsilon}_{s_i_b}(k', k) = \langle k' | \frac{2\pi}{\lambda} \text{psi_ib} | k \rangle$
- $\text{Upsilon}_{s_o_b}(k', k) = \langle k' | \frac{2\pi}{\lambda} \text{psi_ob} | k \rangle$

2. Center Surface Absorption

- $\text{Upsilon}_{c_f_f}(k', k) = \langle k' | \frac{2\pi}{\lambda} \text{psi} | k \rangle$
- $\text{Upsilon}_{c_f_b}(k', k) = \langle k' | \frac{2\pi}{\lambda} \text{psi_ib} | k \rangle$

3. Offcenter Surface Absorption

- $\text{Upsilon}_{c_ib_f}(k', k) = \langle k' | \frac{2\pi}{\lambda} \text{psi} | k \rangle$
- $\text{Upsilon}_{c_ib_b}(k', k) = \langle k' | \frac{2\pi}{\lambda} \text{psi_ib} | k \rangle$

- $\text{Upsilon}_{\text{c_ob_b}}(k', k) = \langle k' | \frac{2\pi}{\lambda} \text{psi_ob} | k \rangle$

Note that we do not take the angle of propagation relative to the surface into account. For example, to properly calculate the scattering matrix for immediate reflection off of the $z = 0$ face due to power P_a of bulk absorption:

$$\underline{S} = \text{Expm}(i2P_a \cos(\theta_i) \text{Upsilon}_{\text{s_i_f}})$$

where the factor of 2 comes from reflection. Similarly, to calculate the scattering matrix for transmission through the optic due to power P_a of bulk absorption:

$$\underline{S} = \text{Expm}(iP_a(n-1) \cos(\theta_p)(\text{Upsilon}_{\text{s_i_b}} - \text{Upsilon}_{\text{s_i_f}}))$$

where the factor $n - 1$ comes from transmission through additional substrate material.

5 Tests Against Analytical Models

The main limitation we expect in this finite element model are situations where we expect relatively large temperature gradients (i.e., heating beams very small compared to the size of the optic), or relatively large physical distortions (i.e., materials with large thermal expansion coefficients). We consider three cases to test this model against existing analytical solutions for cylindrically symmetric heating in an isotropic material [1, 2, 3, 4] (dashed curves represent the analytical solution, solid curves represent the model's output):

5.1 Case 1: Large beam, fused silica optic.

| Parameter | Value | Units |
|----------------|-----------------------|-------------------|
| radius | 0.125 | m |
| height | 0.1 | m |
| waist | 0.0365 | m |
| iangle | 0 | - |
| rho | 2202 | Kg/m ³ |
| c | 746 | J/Kg/°K |
| k_th | 1.38 | W/m/°K |
| dndT | 8.7×10^{-6} | 1/°K |
| alpha | 5.1×10^{-7} | 1/°K |
| youngs_modulus | 7.26×10^{10} | Pascal |
| poisson_ratio | 0.16 | - |
| emissivity | 0.76 | - |
| T0 | 293 | °K |

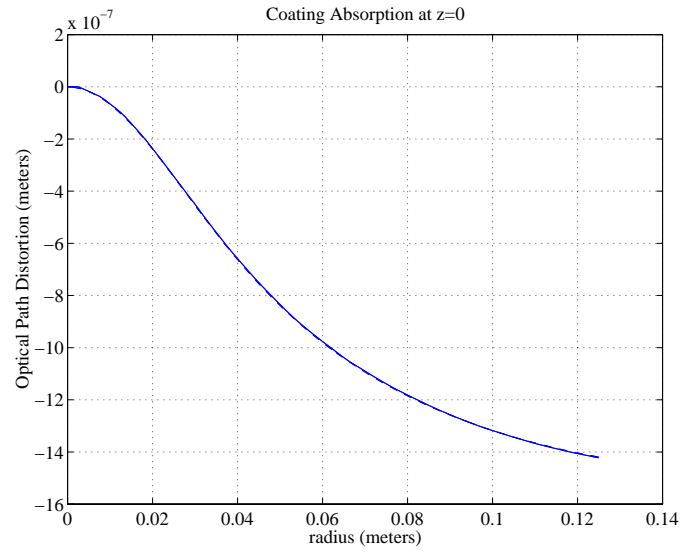
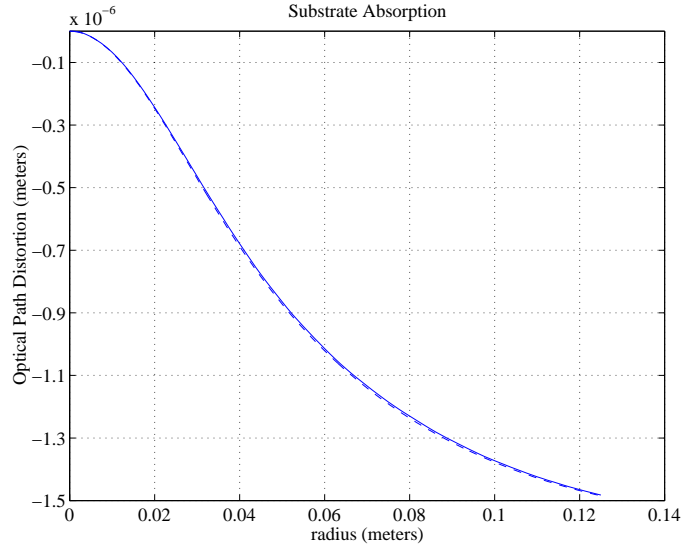


Figure 2: Large beam, fused silica optic, thermal lensing.

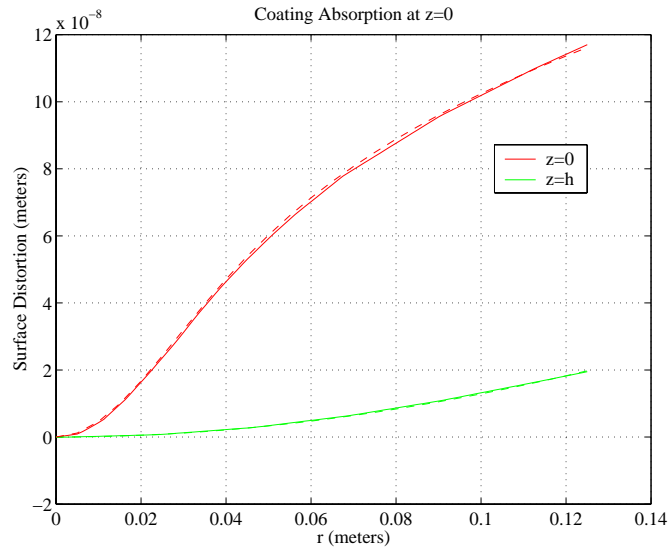
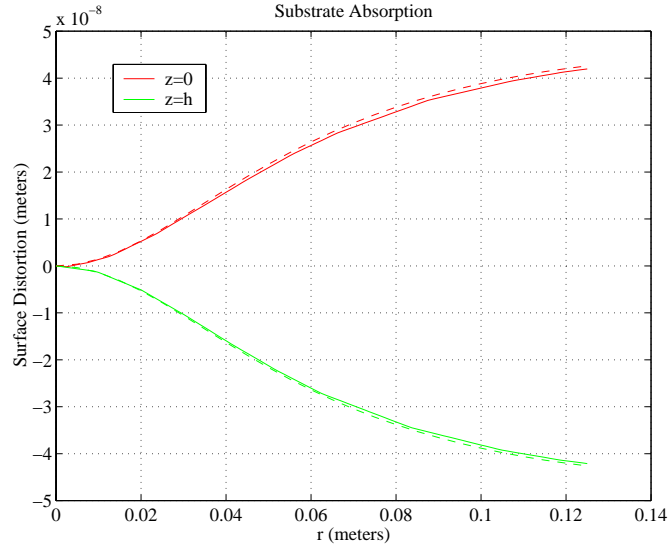


Figure 3: Large beam, fused silica optic, thermoelastic deformation.

5.2 Case 2: Large beam, sapphire optic.

| Parameter | Value | Units |
|----------------|-----------------------|-------------------|
| radius | 0.125 | m |
| height | 0.1 | m |
| waist | 0.0365 | m |
| iangle | 0 | - |
| rho | 3980 | Kg/m ³ |
| c | 770 | J/Kg/°K |
| k_th | 33 | W/m/°K |
| dndT | 13×10^{-6} | 1/°K |
| alpha | 5.1×10^{-6} | 1/°K |
| youngs_modulus | 3.35×10^{11} | Pascal |
| poisson_ratio | 0.25 | - |
| emissivity | 0.535 | - |
| T0 | 293 | °K |

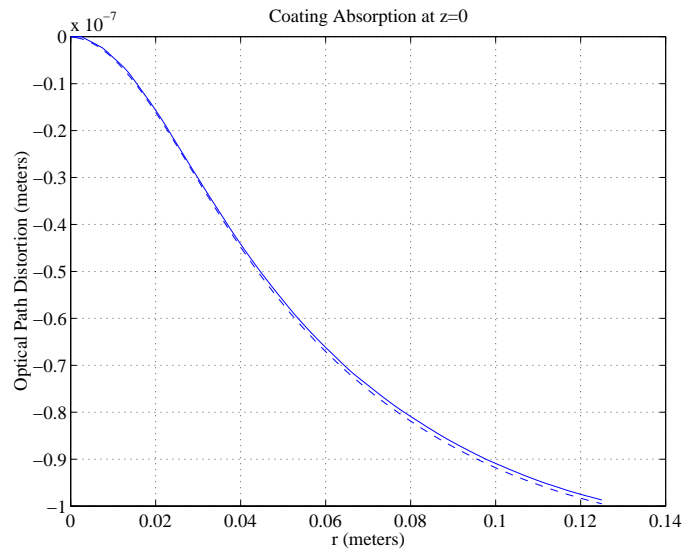
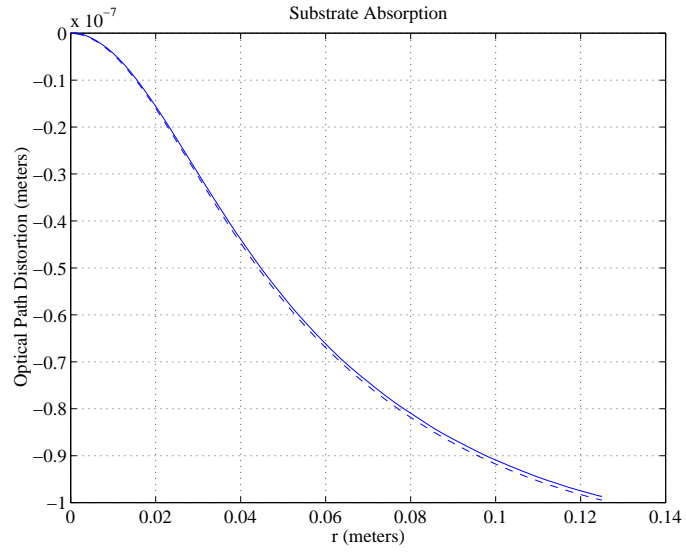


Figure 4: Large beam, sapphire optic, thermal lensing.

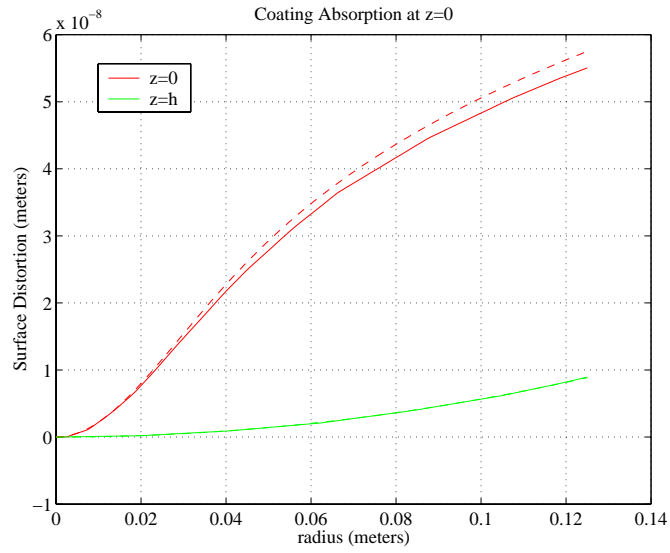
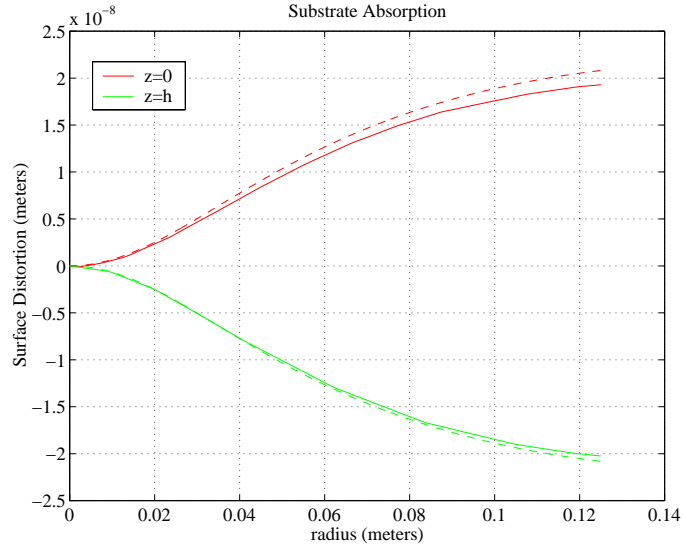


Figure 5: Large beam, sapphire optic, thermoelastic deformation.

5.3 Case 3: Small beam, sapphire optic.

| Parameter | Value | Units |
|----------------|-----------------------|-------------------|
| radius | 0.125 | m |
| height | 0.1 | m |
| waist | 0.0365 | m |
| iangle | 0 | - |
| rho | 3980 | Kg/m ³ |
| c | 770 | J/Kg/°K |
| k_th | 33 | W/m/°K |
| dndT | 13×10^{-6} | 1/°K |
| alpha | 5.1×10^{-6} | 1/°K |
| youngs_modulus | 3.35×10^{11} | Pascal |
| poisson_ratio | 0.25 | - |
| emissivity | 0.535 | - |
| T0 | 293 | °K |

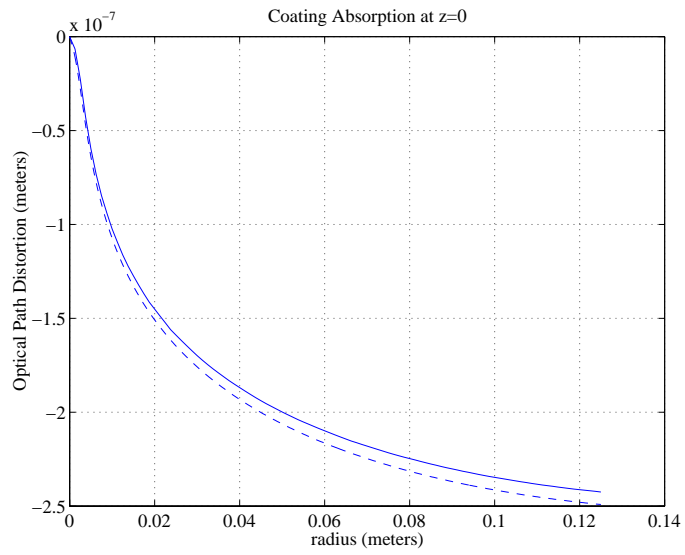
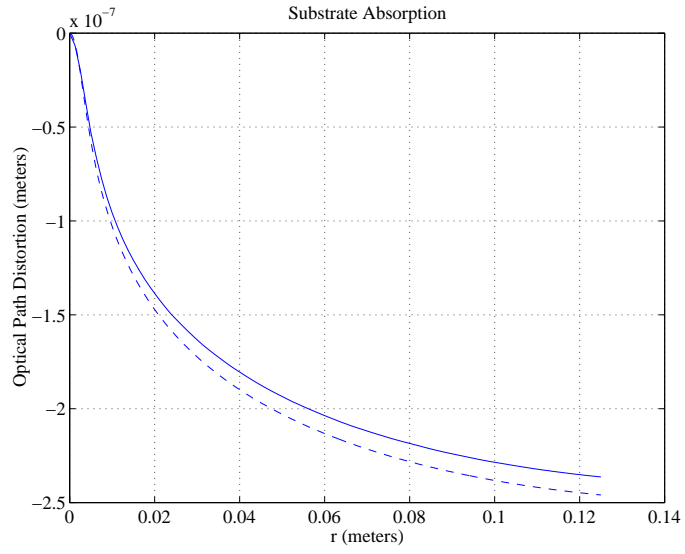


Figure 6: Small beam, sapphire optic, thermal lensing.

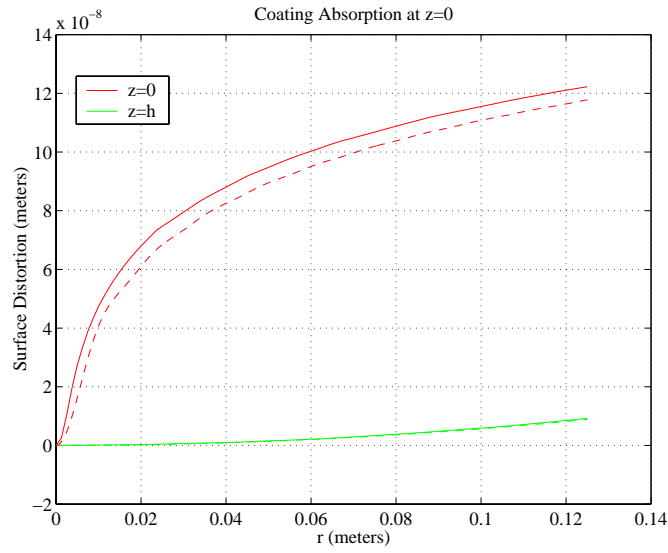
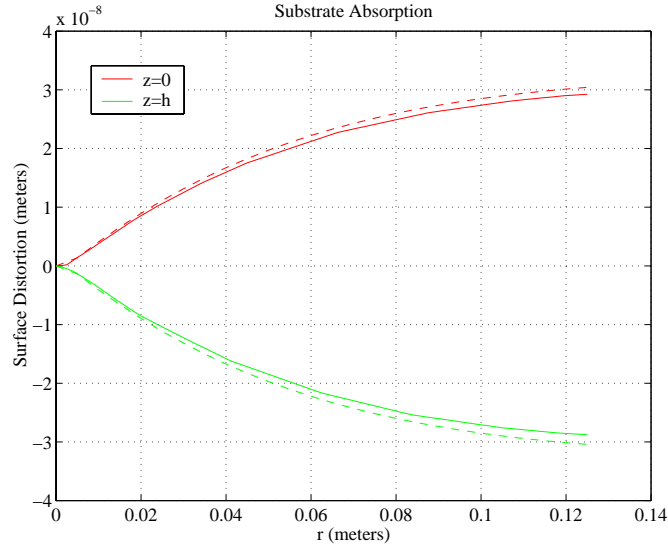


Figure 7: Small beam, sapphire optic, thermoelastic deformation.

References

- [1] Hello P. and Vinet J., *Analytical models of thermal aberrations in massive mirrors heated by high power laser beams*, J. Phys. France **51** (1990) 1267-1282
- [2] Hello P. and Vinet J., *Analytical models of transient thermoelastic deformations of mirrors heated by high power cw laser beams*, J. Phys. France **51** (1990) 2243-2261
- [3] Beausoleil R., *Melody/Matlab: Object-Oriented Model of Gravitational-Wave Interferometers Using Matlab*, software available at the URL:
<http://www.phys.ufl.edu/LIGO/LIGO/STAIC/SOFT/>
- [4] Beausoleil R., *Melody/Matlab: Object-Oriented Model of Gravitational-Wave Interferometers Using Matlab*, LIGO internal document (LIGO document number LIGO-G010301-00-Z).

Appendix A: Notation

| | |
|-----------------------------|--|
| <code>mfile</code> | The Matlab script, function, or variable named “mfile” |
| \vec{v} | Row vector |
| \mathbf{v} | Column vector |
| \underline{M} | Matrix (rank 2 tensor) |
| $\underline{\underline{M}}$ | Rank 4 tensor |
| $ n\rangle$ | Basis state vector (i.e., eigenket) |

Appendix B: Description of the M-files

.1 `beam.m`

Purpose Gaussian profile function.

Syntax `I=beam(a,w,x)`

Description Returns the value I over x (which can be a matrix or vector) according to the equation:

$$I(x) = \frac{2a}{\pi w^2} \exp\left(-2\frac{x^2}{w^2}\right)$$

.2 `calculate_matrices.m`

Purpose Calculates the scattering matrices given raw thermal distortions and saves to file.

Syntax `calculate_matrices(m,l,bf,cf,ocf,out)`

Description After the finite element model solves for both the temperature profile and thermoelastic distortion, and the solution is interpolated onto a rectangular grid, `calculate_matrices` is run to calculate the scattering matrices in the Hermite Gauss basis. Input parameters are:

- `m` (integer) the total number of modes over which the calculation is performed.
- `l` (double) the wavelength of the scattering beam (1064 nm for LIGO).
- `bf` (string) mat file where distortions due to bulk absorption may be found.
- `cf` (string) mat file where distortions due to center surface absorption may be found.
- `ocf` (string) mat file where distortions due to offcenter surface absorption may be found.
- `out` (string) mat file where the output of the calculation will be saved.

.3 `find_distortions.m`

Purpose Interpolates to find raw thermal distortions and saves to file.

Syntax `find_distortions(fem,Xres,Zres,out)`

Description After the finite element solves for both the temperature profile and thermoelastic distortion, `find_distortions` is run to interpolate the solution saved in the object `fem` to a rectangular grid of resolution `Xres` in x and y , and resolution `Zres` in z . The raw surface distortions are simply recentered according to their location and interpolated in 2D over x and y . The thermal lenses are calculated by interpolating the temperature solution over the entire volume, and then integrating the solution over z . The raw distortions are saved in the file `out`. Input parameters are:

- `fem` (FEMLAB object) object which contains the entire finite element model
- `Xres` (double) spacing for the grid in x and y (optional)
- `Zres` (double) spacing for the grid in z (optional)
- `out` (string) output file

.4 get_variable.m

Purpose Retrieves variables stored in a FEMLAB object.

Syntax [varargout]=get_variable(fem,varargin)

Description All user-specified properties are saved in the FEMLAB object fem, and may be extracted using get_variable. The number of output arguments must match the number of input arguments plus one (the object fem). Parameters available for output are:

| | |
|---|---|
| 'rho' | Density (Kg/m ³) |
| 'c' | Heat capacity (J/Kg/°K) |
| 'k_th_x', 'k_th_y', and 'k_th_z' | Components of the thermal conductivity vector (W/m/°K) |
| 'sigma' | Stefan-Boltzmann constant times emissivity (W/m ² /°K ⁴) |
| 'T0' | Ambient temperature (°K) |
| 'dndT' | Refractive index derivative with respect to temperature (1/°K) |
| 'depth' | Thickness of the optic (m) |
| 'radius' | Radius of the optic (m) |
| 'index' | Index of refraction (dimensionless) |
| 'waist' | Waist of the heating beam (m) |
| 'iangle' | Incident angle of the heating beam (radians) |
| 'angle' | Substrate propagation angle of the heating beam (radians) |
| 'alpha_x', 'alpha_y', and 'alpha_z' | Components of the thermal expansion coefficient vector (1/°K) |
| 'c11', 'c12', 'c13', 'c22', 'c23', 'c33', 'c44', 'c55', and 'c66' | Components of the elastic modulus matrix (Pascal) |
| 'gammax', 'gammay', and 'gammaz' | Components of the stress temperature modulus vector (Pascal/°K) |
| 'abs_cent' | Absorption coefficient for center absorption (dimensionless) |
| 'abs_offcent' | Absorption coefficient for offcenter absorption (dimensionless) |
| 'abs_bulk' | Absorption coefficient for bulk absorption (1/m) |
| 'meanT' | The DC temperature increase subtracted before calculating u |

.5 numerical_optic.m

Purpose The model engine.

Syntax numerical_optic

Description Run from the command prompt, numerical_optic takes the material and geometry parameters from optic_params, as well as the solution options, and solves for the desired cases of absorption. It then calls find_distortions to find the raw optical distortions, and then calls calculate_matrices to calculate all the relevant scattering matrices.

.6 optic_params.m

Purpose Material and geometry parameters.

Syntax `optic_params`

Description Material and geometry parameters that define the model are input to `numerical_optic` by this script. Parameters available for user input are:

| | |
|--|--|
| <code>rho</code> | Density (Kg/m ³) |
| <code>c</code> | Heat capacity (J/Kg/°K) |
| <code>k_th_x, k_th_y, and k_th_z</code> | Components of the thermal conductivity vector (W/m/°K) |
| <code>T0</code> | Ambient temperature (°K) |
| <code>dndT</code> | Refractive index derivative with respect to temperature (1/°K) |
| <code>depth</code> | Thickness of the optic (m) |
| <code>radius</code> | Radius of the optic (m) |
| <code>index</code> | Index of refraction (dimensionless) |
| <code>waist</code> | Waist of the heating beam (m) |
| <code>iangle</code> | Incident angle of the heating beam (radians) |
| <code>alpha_x, alpha_y, and alpha_z</code> | Components of the thermal expansion coefficient vector (1/°K) |
| <code>c11, c12, c13, c22, c23, c33, c44, c55, and c66</code> | Components of the elastic modulus matrix (Pascal) |
| <code>lambda</code> | Wavelength of of the probe beam |
| <code>spatial_mode_count</code> | Number of modes <code>calculate_matrices.m</code> calculates over. |

Solution options available are:

| | |
|---------------------------------|--|
| <code>solve_bulk</code> | (logical) flag to solve for bulk absorption |
| <code>bulkout</code> | (string) mat file to save the solution for bulk absorption |
| <code>solve_cent</code> | (logical) flag to solve for center surface absorption |
| <code>centout</code> | (string) mat file to save the solution for center surface absorption |
| <code>solve_offcent</code> | (logical) flag to solve for offcenter surface absorption |
| <code>offcentout</code> | (string) mat file to save the solution for offcenter surface absorption |
| <code>matrices_calculate</code> | (logical) flag to calculate all associated scattering matrices |
| <code>matrixout</code> | (string) mat file to save the scattering matrices |
| <code>dc_temp_subtract</code> | (logical) flag to calculate distortions with DC temperature increase removed |

.7 numerical_optic_plot.m

Purpose Plots the solution saved in a FEMLAB object.

Syntax `numerical_optic_plot(fem)`

Description Plots the solution saved in a FEMLAB object `fem` in 3 dimensions. Color represents temperature, and surfaces are plotted distorted proportionally to the calculated physical distortion.