DISCRETE AND CONTINUOUS
DYNAMICAL SYSTEMS
Volume **33**, Number **11&12**, November & December **2013**

doi:10.3934/dcds.2013.33.5441

pp. **5441–5455**

# ESTIMATING EIGENVALUES OF AN ANISOTROPIC THERMAL TENSOR FROM TRANSIENT THERMAL PROBE MEASUREMENTS

Steve Rosencrans and Xuefeng Wang

Department of Mathematics
Tulane University
New Orleans, LA 70118, USA

Shan Zhao

Department of Mathematics
University of Alabama
Tuscaloosa, AL 35487-0350, USA

Abstract. We propose a new method for estimating the eigenvalues of the thermal tensor of an anisotropically heat-conducting material, from transient thermal probe measurements of a heated thin cylinder.

We assume the principal axes of the thermal tensor to have been identified, and that the cylinder is oriented parallel to one of these axes (but we outline what is needed to overcome this limitation). The method involves estimating the first two Dirichlet eigenvalues (exponential decay rates) from transient thermal probe data. These implicitly determine the thermal diffusion coefficients (thermal tensor eigenvalues) in the directions of the other two axes. The process is repeated two more times with cylinders parallel to each of the remaining axes.

The method is tested by simulating a temperature probe time-series (obtained by solving the anisotropic heat equation numerically) and comparing the computed thermal tensor eigenvalues with their true values. The results are generally accurate to less than 1% error.

1. **Introduction.** This paper proposes a new method for estimating the eigenvalues of the thermal tensor of an anisotropically conducting material (for example, a nanocomposite) from transient thermal decay data.

Techniques for estimating thermal diffusion coefficients have been largely of three types:

- Those making use of a steady state distribution of temperature (see, for example, [7]).
- Those making use of a transient heat flow (see, for example, [2, 3]).
- Exact analytic formulations, e.g., "final overdetermination" ([5, p. 218]).

The first category uses a "divided bar" technique: two samples ("bars") are exposed to the same heat source. One is the material with unknown thermal diffusion coefficient and the other ("reference bar") has known parameters. Comparison at equilibrium yields an estimate for the unknown coefficient. See [7].

2010 *Mathematics Subject Classification.* Primary: 35K05, 80-04, 80A23.

*Key words and phrases.* Thermal tensor, inverse problem.

Our approach is of the transient category. But previous procedures in that category have been confined to unbounded geometries for which there are explicit formulas (see [4]) for the transient temperature flow. An example is [2], which uses the half-space $x_3 \geq 0$ in 3D. These explicit formulas, when compared with the measured temperature, enable diffusion coefficients to be estimated. Our experimental domain will be a *bounded* region. Unbounded domains (such as a half-space) can of course only be approximated in a real experiment, and this truncation would be an additional source of error in such treatments. Our proposed method is new and remains to be tested, but we do provide evidence (in Section 6) that suggests it would provide accurate recovery of diffusion coefficients.

Consider a bounded 3D domain $\Omega$ composed of the material. Suppose homogeneous Dirichlet/Neumann boundary conditions and that at time $t = 0$ the temperature is assigned everywhere on $\Omega$, say $u = u_0(x_1, x_2, x_3)$. The temperature $u$ at later times is the solution to the anisotropic heat equation

$$u_t = \sum_{i,j} a_{ij} u_{x_i x_j} \tag{1}$$

in which the symmetric positive-definite matrix $A = (a_{ij})$ (assumed constant) is the so-called *thermal tensor*. After a rotation to its principal coordinate system, the matrix becomes diagonal, with entries equal to the eigenvalues of $A$. We assume that the three (mutually orthogonal) principal axes of the thermal tensor $A$ have been identified ([2] also begins with this assumption), and in effect

$$A \to \begin{pmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & c \end{pmatrix}$$

where the eigenvalues $a > 0$, $b > 0$, and $c > 0$ are unknown. The heat equation in these principal coordinates is

$$u_t = a u_{xx} + b u_{yy} + c u_{zz}. \tag{2}$$

For $t \geq 0$

$$u(x, y, z, t) = \sum_{n \geq 1} c_n(x, y, z) e^{-\lambda_n t} \tag{3}$$

for some coefficients $\{c_n\}$, and distinct exponential decay rates (distinct Dirichlet eigenvalues, see Section 2)

$$0 < \lambda_1 < \lambda_2 < \lambda_3 < \ldots \to \infty.^{[1]}$$

The eigenspace of $\lambda_1$ is simple. In this generality the eigenspace of $\lambda_2$ may not be simple, but in our main application it is. Provided the sampling point $(x_0, y_0, z_0) \in \Omega$ and initial data are *acceptable* (this will be clarified in Section 2), knowledge of $u(x_0, y_0, z_0, t)$ for $t \geq 0$ implicitly determines the decay rates $\lambda_1, \lambda_2, \ldots$. At any acceptable sampling point the time-series of temperature measurements

$$u(x_0, y_0, z_0, 0),\ u(x_0, y_0, z_0, t_1),\ u(x_0, y_0, z_0, t_2),\ \ldots,\ u(x_0, y_0, z_0, t_N) \tag{4}$$

$(0 < t_1 < t_2 < \ldots < t_N)$ can be used to estimate (several terms of) the $\lambda$-sequence, e.g., the first three terms.

[1] In the original $x_1, x_2, x_3$ coordinates we would have $u(x_1, x_2, x_3, t) = \sum_{n \geq 1} a_n(x_1, x_2, x_3) e^{-\lambda_n t}$. The $\lambda$'s are the same as in (3) because they are rotational invariants.

If $\lambda_1, \lambda_2, \lambda_3$ could be accurately estimated, the recovery of the thermal tensor eigenvalues $a, b, c$ would be a simple matter (see Appendix A). However, the curve-fitting

$$u(x_0, y_0, z_0, t) \approx \sum_{n=1}^{3} d_n e^{-\lambda_n t} \quad t \text{ large} \tag{5}$$

on the basis of the time-series data (4) is an ill-posed problem, see [8]. The accurate estimation of *three* $\lambda$'s seems to be out of the question. We take $\Omega$ to be a thin cylinder with axis parallel to one of the principal axes of the thermal tensor. We will then (in Section 2) outline a procedure requiring in effect only the estimation of $\lambda_1$ and $\lambda_2$, but for two different initial data, so that the estimation of only *one* dominant exponential decay coefficient (at a time) is required. This permits much greater precision in our estimates.

In Section 3 we carry out the estimation of $\lambda_1$ and $\lambda_2$ from the time-series data $u_A$ and $u_B$ specified in Section 2.

In Section 4 we show how the estimates of $\lambda_1$ and $\lambda_2$ lead to estimates of the eigenvalues $a$ and $b$ of the thermal tensor $A$, the first two coefficients in equation (2). The method requires knowledge of the functions

$$\epsilon \to \tilde{\lambda}_i(\epsilon), \quad 0 < \epsilon \leq 1, \quad i = 1, 2$$

where $\tilde{\lambda}_1(\epsilon)$ and $\tilde{\lambda}_2(\epsilon)$ are the principal and second Dirichlet eigenvalue of the elliptic operator $-\epsilon^2 \frac{\partial^2}{\partial x^2} - \frac{\partial^2}{\partial y^2}$ on the unit disk. These are tabulated in Appendix B.

Section 5 shows how to estimate $c$, the last of the three thermal tensor eigenvalues.

In Section 6 we verify the accuracy of our method by *simulating* experimental temperature probe data for a 2D disk with *given* diagonal thermal tensor

$$\begin{pmatrix} a & 0 \\ 0 & b \end{pmatrix}, \tag{6}$$

by numerically solving (using Matlab's Partial Differential Equation Toolbox) the anisotropic heat equation (2) with appropriate initial data and Dirichlet boundary conditions (and, optionally, adding small random noise to the data and solution). This gives us a time-series to which we apply the methods outlined above, obtaining estimates $a_{\text{est}}$ and $b_{\text{est}}$. These generally have error less than 1%. Matlab scripts for these calculations are included in Appendices C, D, E, and F.

## 2. **Thin cylinder. Initial data.**

Now we specify the domain $\Omega$ to be a cylinder whose axis is the $z$-axis:

$$\Omega = \left\{ (x_1, y_1, z) | x_1^2 + y_1^2 < R^2, \ 0 < z < h \right\}, \tag{7}$$

with Dirichlet conditions ($u = 0$) on the vertical sides and Neumann conditions ($\partial u / \partial z = 0$) on the top and the bottom.[2] Since the domain is the cartesian product of the 2D disk of radius $R$ and the 1D interval $0 < z < h$, the eigenfunctions of $\Omega$ are of the form $V \cdot W$, where (after a rotation about the $z$-axis) $V$ is a (normalized) Dirichlet eigenfunction of the diagonal operator $-a \frac{\partial^2}{\partial x^2} - b \frac{\partial^2}{\partial y^2}$ on the *same* disk

$$\begin{cases} aV_{xx} + bV_{yy} + \lambda V = 0 & x^2 + y^2 < R^2 \\ V = 0 & x^2 + y^2 = R^2, \end{cases}$$

[2] The Neumann condition can be achieved by applying a thin (thickness $\delta$) layer of (an insulating) material with thermal diffusion coefficient of order $\epsilon \to 0$, where $\delta \sim C\epsilon^{\beta}$, $0 \leq \beta < 1$. See [6].

(assume $a \leq b$) with eigenvalues $\lambda_1, \lambda_2, \ldots$ and $W$ is a Neumann eigenfunction of the interval $0 < z < h$,

$$W_j = \sqrt{\frac{2}{h}} \cos(j\pi z/h) \quad j = 0, 1, 2, \ldots,$$

with eigenvalues $cj^2\pi^2/h^2$. The eigenvalues of the *full* problem are of the form

$$\lambda_n + cj^2\pi^2/h^2, \quad n = 1, 2, \cdots \quad j = 0, 1, 2, \cdots .$$

Our intuition tells us that if the cylinder is thin enough ($h$ small enough) the situation is closer to 2D. Indeed, the following are eigenvalues *not* necessarily in numerical order:

$$\lambda_1, \lambda_2, \lambda_1 + c\pi^2/h^2, \lambda_2 + c\pi^2/h^2, \cdots . \tag{8}$$

Assume from now on that $h$ is small enough,

$$h < \sqrt{\frac{c\pi^2}{\lambda_2 - \lambda_1}}; \tag{9}$$

then

$$\lambda_2 < \lambda_1 + c\pi^2/h^2, \tag{10}$$

and consequently the first two (Dirichlet/Neumann) eigenvalues of the thin cylinder are the same as the first two Dirichlet eigenvalues of the 2D disk.[3] We emphasize that $\lambda_1$, $\lambda_2$, etc., are invariant under rotations about the $z$-axis, just as is the cylinder itself. The first two corresponding normalized eigenfunctions of the *cylinder* are $V_1/\sqrt{h}$ and $V_2/\sqrt{h}$.

With initial data independent of $z$ (which we assume), $c_1$ and $c_2$ in the expansion (3) are independent of $z$ and are given by the 2D formulas (in this situation the eigenspace of $\lambda_2$ is simple)

$$c_1(x, y) = V_1(x, y) \int_{\xi^2+\eta^2<R^2} u_0(\xi, \eta) V_1(\xi, \eta) d\xi\, d\eta \tag{11}$$

$$c_2(x, y) = V_2(x, y) \int_{\xi^2+\eta^2<R^2} u_0(\xi, \eta) V_2(\xi, \eta) d\xi\, d\eta. \tag{12}$$

The principal eigenfunction $V_1$ is positive in the interior of the disk, whereas (because $b > a$, *which we assume*) the $y$-axis is a *nodal line* for $V_2$, and in fact it can be shown that $V_2$ is an odd function of $x$ and $V_1$ is an even function of both $x$ and $y$.

We use *two* initial data (both for the initial-value problem to be used to simulate experimental data, and also for the experiment itself).

**Data** $u_{0A}$**:** initial data even in $x$ and hence orthogonal to $V_2$, $u_{0A} = R^2 - x^2 - y^2$. The solution to the heat equation $u_A$ is of the form $c_1 e^{-\lambda_1 t} + c_3 e^{-\lambda_3 t} + \mathrm{O}(e^{-\lambda_4 t}) \sim c_1 e^{-\lambda_1 t} \quad (t \to \infty)$.

**Data** $u_{0B}$**:** initial data odd in $x$ and hence orthogonal to $V_1$, $u_{0B} = x \cdot u_{0A}$. The solution to the heat equation $u_B$ is of the form $d_2 e^{-\lambda_2 t} + d_3 e^{-\lambda_3 t} + \mathrm{O}(e^{-\lambda_4 t}) \sim d_2 e^{-\lambda_2 t} \quad (t \to \infty)$.

[3] So some rough a priori estimates of $\lambda_2 - \lambda_1$, and $c$ would enable us to determine a lower bound for the right side of (9), and hence an explicit upper bound for $h$.

Thus to estimate $\lambda_1$ and $\lambda_2$ it is only necessary to estimate the *leading* exponential decay coefficient for each of the two time-series $u_A$ and $u_B$.

We have not yet discussed the choice of sampling point (for the temperature probe).

- It can be on the top or bottom of the cylinder.
- It should not be near the lateral boundary, for then all terms will be very small.
- In the case of $u_B$, it should not be near the $y$-axis. This would make $d_2$ small, where $u_B \approx d_2 e^{-\lambda_2 t} + d_3 e^{-\lambda_3 t} + \dots$ and this would delay the dominance of the first term.
- Apart from this, the method is not sensitive to the sampling point.

The knowledge of *all three* principal axes was only used in the specification of odd and even data, just above. If only *one* principal axis were known (call it the $z$-axis), we would not have knowledge of data satisfying the above orthogonality conditions and the estimation of $\lambda_1$ and $\lambda_2$ from thermal probe data would be much less accurate. After estimating $\lambda_1$ and $\lambda_2$, we would pass to Section 4.[4]

3. **Estimation of Dirichlet eigenvalues $\lambda_1$ and $\lambda_2$ from the two time-series of temperature probe data, $u_A$ and $u_B$.** Following a suggestion of Mac Hyman (personal communication), we consider the ratio $u(t+\delta)/u(t)$, $\delta \sim \mathrm{O}(1)$ not yet determined. From the above it easily follows that

$$\frac{u_A(t+\delta)}{u_A(t)} = e^{-\lambda_1 \delta} + \mathrm{O}(e^{-(\lambda_3-\lambda_1)t}) \quad t >> \frac{1}{\lambda_3-\lambda_1} \tag{13}$$

$$\frac{u_B(t+\delta)}{u_B(t)} = e^{-\lambda_2 \delta} + \mathrm{O}(e^{-(\lambda_3-\lambda_2)t}) \quad t >> \frac{1}{\lambda_3-\lambda_2}, \tag{14}$$

which suggests simple estimates for the leading exponential decay coefficients $\lambda_1$ and $\lambda_2$, from the $u_A$ and $u_B$ time-series respectively.

$$\lambda_1 \approx -\frac{1}{\delta}\log\left(\frac{u_A(t+\delta)}{u_A(t)}\right) \quad t >> \frac{1}{\lambda_3-\lambda_1} \tag{15}$$

$$\lambda_2 \approx -\frac{1}{\delta}\log\left(\frac{u_B(t+\delta)}{u_B(t)}\right) \quad t >> \frac{1}{\lambda_3-\lambda_2}. \tag{16}$$

But the time-series of temperature measurements is unlikely to be accurate for such large times; the temperature has decayed considerably. The same seems to be true of the simulated time-series discussed in Section 6: evaluating the right sides of the last two equations (15) and (16) for large $t$ does not yield correct results. We proceed therefore to use these estimates for smaller $t$, with no guarantee that they will work. We take $u_A(t+\delta)$ and $u_A(t)$ to be time-adjacent points in the $u_A$ time-series, and we do the same for $u_B$, i.e., $\delta$ is the time between two successive measurements. It may change from one pair of measurements to another. In the case of the simulated experimental data, there are generally 81 observations, hence as many as 80 separate estimates for each of $\lambda_1$ and $\lambda_2$.

In every calculation we have made (i.e., Section 6 with various values of the thermal tensor eigenvalues $a$ and $b$), we have found that the 80 estimates of the points $(\lambda_1, \lambda_2)$ display a pronounced clustering in the plane: many outliers but also

[4] It's not a problem that $\lambda_1$ and $\lambda_2$ are Dirichlet eigenvalues related to (1) whereas the Dirichlet eigenvalues in Section 4 are Dirichlet eigenvalues related to (2). As we have pointed out, the Dirichlet eigenvalues are rotational invariants, hence the same for both cases.

many points very close together. There is always one and only one cluster. See Figures 1, 2, 3.

Our algorithm (Matlab script presented in Appendices D and F) for eliminating outliers in the $(\lambda_1, \lambda_2)$ plane is:

- Drop all data points whose $\lambda_1$ or $\lambda_2$ z-score[5] is greater than 1 in absolute value.
- Repeat the process until only one or two data points remain.
  - If one point remains it is taken as best estimate.
  - If two points remain the mean of the two is taken as best estimate.

We compared the above procedure with

- Exponential curve-fitting using Matlab's Curve Fitting Toolbox (fitting 'exp2').
- Introduction of a *single* time-series $u_C$, arising from initial data not orthogonal to either the principal or second eigenfunction and application of the above Toolbox to find the best fit to $u_C$ of the form
$$e_1 e^{-\alpha_1 t} + e_2 e^{-\alpha_2 t}.$$
- Per Sundqvist's script 'exp2fit' [9].

None of these three approached the accuracy obtained by using data $u_{0A}$ and $u_{0B}$ and (13) and (14).

4. **Recovery of thermal tensor eigenvalues $a$ and $b$ from the Dirichlet eigenvalues $\lambda_1$ and $\lambda_2$.** Recall that the $z$-axis is a principal axis for the thermal tensor $A$, hence $A$ is similar to

$$A = \begin{pmatrix} b_{11} & b_{12} & 0 \\ b_{21} & b_{22} & 0 \\ 0 & 0 & c \end{pmatrix}$$

where $B = (b_{ij})$ is a $2\times 2$ positive-definite matrix, whose eigenvalues we have called $a$ and $b$, with $\epsilon = \sqrt{a/b} \in (0,1)$. A rotation to principal coordinates, $(x_1, x_2) \to (x, y)$ and a rescaling of the independent variables yields the following:

$$\boxed{\lambda \text{ is a Dirichlet eigenvalue of } -\sum_{i,j=1}^{2} b_{ij} \frac{\partial^2}{\partial x_i \partial x_j} \text{ on } x_1^2 + x_2^2 \le R^2}$$

$$\Updownarrow$$

$$\boxed{\begin{array}{l}\lambda \text{ is a Dirichlet eigenvalue of } -a\dfrac{\partial^2}{\partial x^2} - b\dfrac{\partial^2}{\partial y^2} \text{ on } x^2 + y^2 \le R^2 \quad \text{where } a \text{ and } b \\ \text{are the eigenvalues of } B = (b_{ij})\end{array}}$$

$$\Updownarrow$$

$$\boxed{\begin{array}{l}\lambda = b\tilde{\lambda}(\epsilon)/R^2 \text{ where } \tilde{\lambda}(\epsilon) \text{ is a Dirichlet eigenvalue of } -\epsilon^2\dfrac{\partial^2}{\partial x^2} - \dfrac{\partial^2}{\partial y^2} \text{ on } x^2 + y^2 \le \\ 1\end{array}}$$

[5] Compute the mean $\mu$ and standard deviation $\sigma$ of all the $\lambda_1$ data points. The *z-score* of a data point $\lambda_1$ is $(\lambda_1 - \mu)/\sigma$. Now do the same for all the $\lambda_2$ data points.

$$\Updownarrow$$

$$\boxed{\lambda = b\tilde{\lambda}(\epsilon)/R^2 \text{ where } \tilde{\lambda}(\epsilon) \text{ is a Dirichlet eigenvalue of } -\Delta \text{ on } \epsilon^2 x^2 + y^2 \leq 1.}$$

The last characterization (Dirichlet eigenvalues of minus Laplacian on an ellipse) will not be used, but is included for completeness.

Let $\lambda_1, \lambda_2$ and $\tilde{\lambda}_1(\epsilon), \tilde{\lambda}_2(\epsilon)$ be the principal and second Dirichlet eigenvalues for the respective situations. We have computed $\tilde{\lambda}_1(\epsilon), \tilde{\lambda}_2(\epsilon)$ in steps of $\Delta\epsilon = .025$. These results are included in Appendix B.[6] Assuming $\lambda_1, \lambda_2$ to have been estimated, the relation $\lambda = b\tilde{\lambda}(\epsilon)/R^2$ implies

$$\frac{\tilde{\lambda}_1(\epsilon)}{\tilde{\lambda}_2(\epsilon)} = \frac{\lambda_1}{\lambda_2}. \tag{17}$$

We solve this equation for $\epsilon$. For this step table-lookup and interpolation (Matlab's *interp1*) are used. Finally,

$$b = \frac{\lambda_1 R^2}{\tilde{\lambda}_1(\epsilon)}$$

(requiring another table-lookup) and

$$a = \epsilon^2 b = \lambda_1 R^2 \left[\frac{\epsilon^2}{\tilde{\lambda}_1(\epsilon)}\right].$$

This completes the recovery of the thermal tensor eigenvalues $a$ and $b$ from thermal decay measurements.

**Remark 1.** We have used a simple re-scaling. The argument does not assume any eigenspace dimensions and does not even use the assumption that $\lambda_1$ and $\lambda_2$ are the *first* and *second* distinct Dirichlet eigenvalues. They could be, for example, the fifth and eighth distinct Dirichlet eigenvalues. (Then $\tilde{\lambda}_1(\epsilon)$ and $\tilde{\lambda}_2(\epsilon)$ would be the fifth and eighth distinct Dirichlet eigenvalues in the scaled situation.)

The Matlab script for these steps is included in Appendix E. It uses the tabular data $\tilde{\lambda}_1(\epsilon), \tilde{\lambda}_2(\epsilon)$, *ratio12* in steps of $\Delta\epsilon = .025$. (These arrays are denoted *lam1shan*, *lam2shan* and *epsshan*. The array *ratio12* is *lam1shan./lam2shan*.) See the table in Appendix B.

The two table-lookups (requiring the publication and dissemination of tables!) and the interpolation can be avoided (at the expense of a slight increase in relative error) by explicit curve-fitting of the two implicitly defined functions $\lambda_1/\lambda_2 \to 1/\tilde{\lambda}_1(\epsilon)$ and $\lambda_1/\lambda_2 \to \epsilon^2/\tilde{\lambda}_1(\epsilon)$, where $\epsilon$ is the solution of Equation (17). Matlab's Curve-Fitting Toolbox gives **explicit approximations** for $a$ and $b$ in terms of $\lambda_1$ and $\gamma :\equiv \lambda_1/\lambda_2$:

$$b = \lambda_1 R^2 \cdot (-.4109\gamma^2 + .9284\gamma - .1182) \pm .011\lambda_1 R^2 \tag{18}$$

$$a = \lambda_1 R^2 \cdot (.04239 \cdot \gamma^{-1.799} - .04862) \pm .006\lambda_1 R^2. \tag{19}$$

[6] The row corresponding to $\epsilon = 0$ has been omitted, as there is no point spectrum in that case. There are known symbolic formulas (not needed here) in terms of Mathieu functions for the scaled eigenvalues $\tilde{\lambda}_i(\epsilon)$. They can be found in [1].

## 5. Recovery of the third thermal tensor eigenvalue $c$.

We now repeat the process with a new thin cylinder whose axis is parallel to, say, the $x$-axis. Our procedure will estimate the thermal tensor eigenvalues $b$ and $c$. Finally, take a last thin cylinder whose axis is parallel to the $y$-axis, and thus estimate $a$ and $c$. Altogether, we have two estimates for each of the three thermal tensor eigenvalues. They should agree.

## 6. Simulated temperature probe data; numerical results.

Matlab's PDE Toolbox was used to create the time-series

$$u(x_0, y_0, t_1), u(x_0, y_0, t_2), \ldots$$

at a sampling point $(x_0, y_0)$, given 'true' values of $a$ and $b$. We have taken $R = 1$. The PDE Toolbox applies the Finite Element Method to solve $u_t = au_{xx} + bu_{yy}$ subject to $u = 0$ on $x^2 + y^2 = 1$ and initially at $t = 0$ either $u = u_{0A} \equiv 1 - x^2 - y^2$ or $u = u_{0B} \equiv x(1 - x^2 - y^2)$.

The unit disk and its boundary are selected in the graphical user interface (gui). The mesh is initialized and refined twice. At this point the arrays $b$ (boundary), $p$ (nodes), $e$ (edges) and $t$ (triangles) are exported to the Matlab workspace, to allow the PDE to be solved using command-line functions rather than the gui. In the m-file q1 quoted in Appendix C, the command `load(DISK_BPET.mat)` loads all these arrays.

The command-line function *parabolic* outputs the solution at a finite sequence of times *tlist*, which we have taken to be $0, h, 2h, 3h, \ldots, 2$ plus a random vector[7] of the same length with mean $h/2$. We allow the solution and data to be perturbed by a small normal relative noise. These random effects are meant to simulate real experimental effects.

To select $h$ appropriately we must consider the order of magnitude of the thermal tensor eigenvalues $a$ and $b$. Recall $b > a$. If $\epsilon = \sqrt{a/b} << 1$, the problem is a singular perturbation, and can't generally be handled by our methods. (But even, for example, $b = 1, a = .1$ is successfully computed.) If $a$ and $b$ are roughly of order 1, we take $h = .025$. Much smaller or much larger values for $h$ would degrade the results. But the scaling of (2) shows that if $a$ and $b$ are (roughly) of order 10, we should replace $h \to h/10$. If $a$ and $b$ are roughly of order .1, then we should set $h \to 10h \approx .25$, etc. The *tlist* variable has to be altered accordingly. So in a real experiment, while $a$ and $b$ are unknown, we do need to have some rough a priori estimate of their order of magnitude.

The results, displayed in Table 1, confirm that our procedure adequately estimates thermal tensor eigenvalues from simulated temperature probe data.

**Acknowledgments.** We acknowledge fruitful discussions with Lev Kaplan, and we thank Mac Hyman and Jed Diem for valuable suggestions. We also thank the referee for pointing out several corrections. X.W. acknowledges support by NSF grant DMS-0707796 and S.Z. acknowledges support by NSF grant DMS-1016579.

## Appendix A. Recovery of all three thermal tensor eigenvalues in 3D.

We take $\Omega$ to be the ball $x_1^2 + x_2^2 + x_3^2 \leq R^2$; assume $c \geq b \geq a \geq 0$. Let $\epsilon = \sqrt{a/c}$,

[7] Components uniformly distributed on the unit interval $0 < x < 1$.

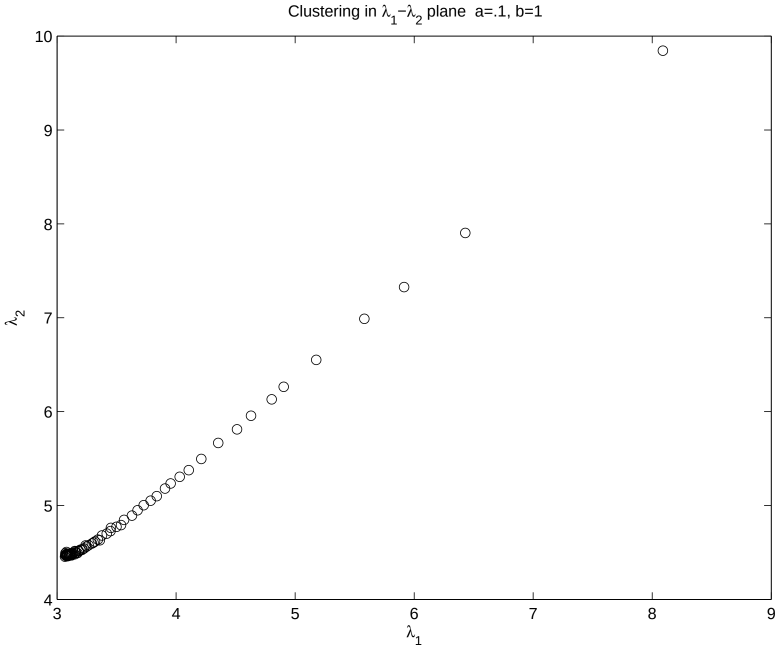

FIGURE 1. Showing clustering in the $\lambda_1$-$\lambda_2$ plane, for the case $a =$ .1, $b = 1$.

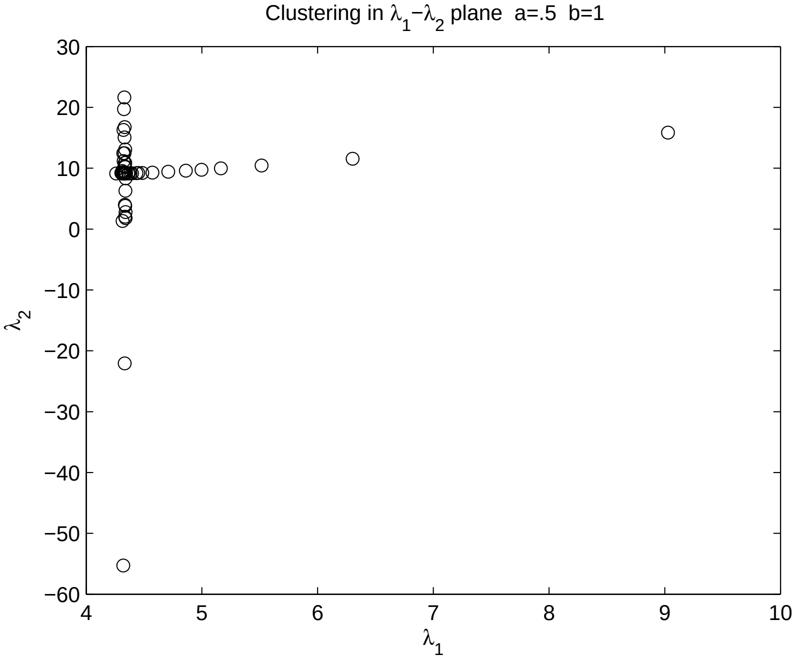

FIGURE 2. Showing clustering in the $\lambda_1$-$\lambda_2$ plane, for the case $a =$ .5, $b = 1$.

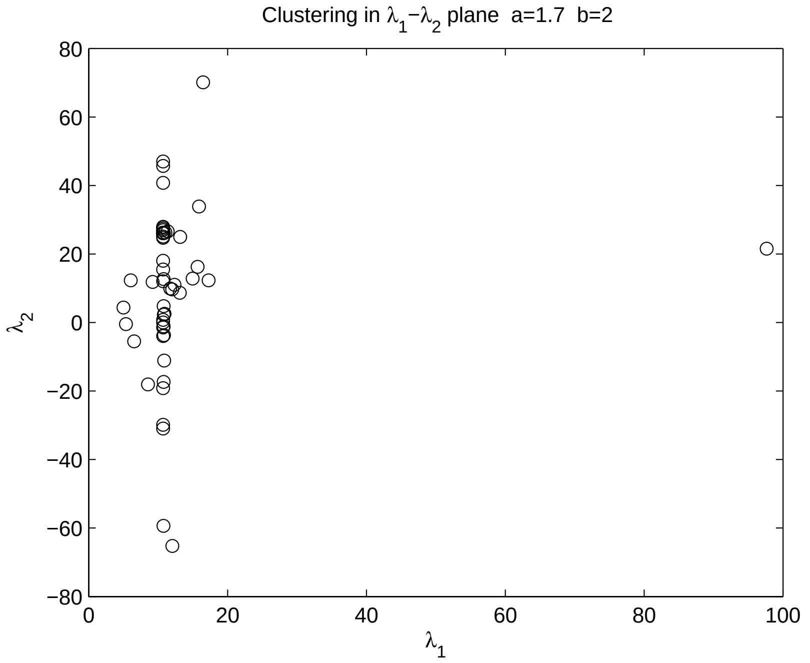

FIGURE 3. Showing clustering in the $\lambda_1$-$\lambda_2$ plane, for the case $a = 1.7$, $b = 2$.

| a | b | a (est.) | b (est.) | Rel. error |
|---|---|---|---|---|
| .1 | 1 | .0982 | 1.0157 | .0180 |
| .5 | 1 | .5060 | .9933 | .0119 |
| .9 | 1 | .9037 | .9965 | .0042 |
| .7 | 2 | .7019 | 1.9994 | .0027 |
| 1.7 | 2 | 1.7163 | 1.9863 | .0096 |
| 3 | 5 | 3.0181 | 4.9785 | .006 |
| 11 | 20 | 11.0864 | 19.8941 | .0079 |
| 90 | 100 | 90.5701 | 99.5474 | .0063 |
| .04 | .2 | .0403 | .2 | .0082 |

TABLE 1. Table listing the estimated values of $a$ and $b$ versus the *true* values of $a$ and $b$. In the the last three rows, the order of magnitude of $a$ and $b$ required adjustments of $h$ and *tlist*, as explained in Section 4. The values of $h$ were .0025, .00025, and .25. The upper limits of *tlist* were .2, .02. and 20. Relative error is defined as $max(abs((a - a_{est})/a), abs((b - b_{est})/b))$.

$\delta = \sqrt{b/c}$. Then $0 \le \epsilon \le \delta \le 1$, and since the Dirichlet eigenvalues are rotational invariants,

$$\boxed{\lambda \text{ is a Dirichlet eigenvalue of } -\sum_{i,j=1}^{3} a_{ij}\frac{\partial^2}{\partial x_i \partial x_j} \text{ on } x_1^2+x_2^2+x_3^2 \le R^2}$$

$$\Updownarrow$$

$$\boxed{\lambda \text{ is a Dirichlet eigenvalue of } -a\frac{\partial^2}{\partial x^2} - b\frac{\partial^2}{\partial y^2} - c\frac{\partial^2}{\partial z^2} \text{ on } x^2+y^2+z^2 \le R^2}$$

$$\Updownarrow$$

$$\boxed{\lambda = c\tilde{\lambda}(\epsilon,\delta)/R^2 \text{ where } \tilde{\lambda}(\epsilon,\delta) \text{ is a Dirichlet eigenvalue of } -\Delta \text{ on } \epsilon^2 x^2 + \delta^2 y^2 + z^2 \le 1.}$$

$$\Updownarrow$$

$$\boxed{\lambda = c\tilde{\lambda}(\epsilon,\delta)/R^2 \text{ where } \tilde{\lambda}(\epsilon,\delta) \text{ is a Dirichlet eigenvalue of } -\epsilon^2\frac{\partial^2}{\partial x^2} - \delta^2\frac{\partial^2}{\partial y^2} - \frac{\partial^2}{\partial z^2} \text{ on } x^2+y^2+z^2 \le 1.}$$

Let $\lambda_1, \lambda_2, \lambda_3$ and $\tilde{\lambda}_1, \tilde{\lambda}_2, \tilde{\lambda}_3$ be the first three distinct Dirichlet eigenvalues in the unscaled and scaled situations. We solve the 2D system of nonlinear equations

$$\frac{\tilde{\lambda}_1(\epsilon,\delta)}{\tilde{\lambda}_2(\epsilon,\delta)} = \frac{\lambda_1}{\lambda_2} \qquad \frac{\tilde{\lambda}_2(\epsilon,\delta)}{\tilde{\lambda}_3(\epsilon,\delta)} = \frac{\lambda_2}{\lambda_3} \tag{20}$$

for $\epsilon$ and $\delta$. (We have tabulated $\tilde{\lambda}_1(\epsilon,\delta), \tilde{\lambda}_2(\epsilon,\delta), \tilde{\lambda}_3(\epsilon,\delta)$ in steps of $\Delta\epsilon = .025$ and $\Delta\delta = .025$.)

The thermal tensor eigenvalues are then

$$\begin{aligned} a &= \lambda_1 R^2 \left[\frac{\epsilon^2}{\tilde{\lambda}_1(\epsilon,\delta)}\right] \\ b &= \lambda_1 R^2 \left[\frac{\delta^2}{\tilde{\lambda}_1(\epsilon,\delta)}\right] \\ c &= \lambda_1 R^2 \left[\frac{1}{\tilde{\lambda}_1(\epsilon,\delta)}\right]. \end{aligned}$$

The Remark in Section 4 also applies here.

## Appendix B. Table of 2D eigenvalues $\tilde{\lambda}_1(\epsilon), \tilde{\lambda}_2(\epsilon), \tilde{\lambda}_3(\epsilon)$ versus $\epsilon$.

## Appendix C. Matlab script q1 generating simulated time-series of temperature probe data.

```
%q1 simulates experimental time-series; thermal tensor (aa,bb)
%                    is known
%   adds small normal relative noise to data
%   timesteps can have a random component also
%
```

| $\epsilon$ | $\tilde{\lambda}_1(\epsilon)$ | $\tilde{\lambda}_2(\epsilon)$ | $\tilde{\lambda}_3(\epsilon)$ | ERR BOUND |
|---|---|---|---|---|
| 0.025 | 2.50715087763041 | 2.58759947042208 | 2.66995715980735 | 2.1949e-07 |
| 0.050 | 2.54790642812031 | 2.71276226397251 | 2.88539367042714 | 4.6746e-08 |
| 0.075 | 2.58974061687542 | 2.84318836801003 | 3.11445396687429 | 1.3979e-09 |
| 0.100 | 2.63273186105141 | 2.97920287996760 | 3.35794073201464 | 7.1599e-09 |
| 0.125 | 2.67696481476064 | 3.12115994039287 | 3.61672306989209 | 6.9907e-09 |
| 0.150 | 2.72253113453280 | 3.26944586691502 | 3.89174347036226 | 5.4634e-09 |
| 0.175 | 2.76953030513289 | 3.42448243335298 | 4.18402415201058 | 1.3831e-09 |
| 0.200 | 2.81807047193797 | 3.58672907551076 | 4.49466683099014 | 1.4141e-09 |
| 0.225 | 2.86826881782867 | 3.75668124643935 | 4.82483633735001 | 1.2707e-09 |
| 0.250 | 2.92025100640909 | 3.93486229580276 | 5.17572353990045 | 2.4295e-09 |
| 0.275 | 2.97414937295098 | 4.12180896073886 | 5.54849533636693 | 9.9854e-10 |
| 0.300 | 3.03010002673113 | 4.31805361161770 | 5.94424588654869 | 4.3397e-09 |
| 0.325 | 3.08823936597537 | 4.52410725438517 | 6.36395999162647 | 1.4534e-09 |
| 0.350 | 3.14870064530032 | 4.74044614138165 | 6.80849252597542 | 3.4963e-10 |
| 0.375 | 3.21161110910151 | 4.96750319420759 | 7.27856203704870 | 1.2470e-09 |
| 0.400 | 3.27708983516641 | 5.20566392292949 | 7.77475446909216 | 2.8263e-10 |
| 0.425 | 3.34524639879916 | 5.45526598093810 | 8.29753254722709 | 5.6024e-10 |
| 0.450 | 3.41618023247429 | 5.71660116729728 | 8.84724741408878 | 2.2520e-09 |
| 0.475 | 3.48998053452207 | 5.98991893744759 | 9.42415014073432 | 1.6618e-09 |
| 0.500 | 3.56672660349568 | 6.27543062013820 | 10.02840162075563 | 7.4453e-10 |
| 0.525 | 3.64648838144603 | 6.57331383765330 | 10.66008001274155 | 7.7266e-10 |
| 0.550 | 3.72932720238970 | 6.88371680052222 | 11.31918507915019 | 8.9693e-10 |
| 0.575 | 3.81529656413977 | 7.20676229126275 | 12.00563895276996 | 1.1218e-09 |
| 0.600 | 3.90444292550250 | 7.54255124654173 | 12.21767129046121 | 5.8978e-10 |
| 0.625 | 3.99680647841861 | 7.89116592283709 | 12.34556703623874 | 9.1338e-10 |
| 0.650 | 4.09242186527282 | 8.25267265161467 | 12.47668291103114 | 6.2281e-10 |
| 0.675 | 4.19131883176124 | 8.62712421665310 | 12.61109112756273 | 3.7693e-10 |
| 0.700 | 4.29352282398541 | 9.01456189325594 | 12.74885867597377 | 1.1344e-09 |
| 0.725 | 4.39905551477386 | 9.41501719233678 | 12.89004738170730 | 1.1132e-09 |
| 0.750 | 4.50793527279893 | 9.82851334306315 | 13.03471403553021 | 7.8444e-10 |
| 0.775 | 4.62017756656750 | 10.25506656058690 | 13.18291056020309 | 5.8575e-10 |
| 0.800 | 4.73579533720385 | 10.69468712876005 | 13.33468421721045 | 5.4681e-10 |
| 0.825 | 4.85479929791676 | 11.14738031232958 | 13.49007783503293 | 9.1072e-10 |
| 0.850 | 4.97719822172391 | 11.61314715238768 | 13.64913004293306 | 4.9958e-10 |
| 0.875 | 5.10299916456171 | 12.09198512605550 | 13.81187552835763 | 7.8995e-10 |
| 0.900 | 5.23220769544059 | 12.58388872981959 | 13.97834527339000 | 1.7316e-09 |
| 0.925 | 5.36482804969993 | 13.08884996711787 | 14.14856681346959 | 3.7777e-10 |
| 0.950 | 5.50086332022875 | 13.60685877627523 | 14.32256446118663 | 1.2242e-09 |
| 0.975 | 5.64031557014096 | 14.13790338807485 | 14.50035954903979 | 1.5125e-09 |
| 1.000 | 5.78318595917881 | 14.68197064164292 | 26.37461642712712 | 2.0045e-09 |

```
%After q1 run q2, which calculates lam1 and lam2 from the time-series
%Then run q3, which calculates aa1, bb1 (estimates of thermal tensor)

clear all
load('DISK_BPET.mat') % loads b p e t (Explanation in Section 6.)
index=68; %place to measure temperature time-series = [.77,.4]
aa=input('aa=');
bb=input('bb=');
stdev=input('stdev=');  %stdev of relative normal noise

% initial data
```

```
x = p(1,:);
y = p(2,:);
u0A = abs(1-x.^2-y.^2); %initial data A
u0B = x.*u0A;           %initial data B, orthog to 1st efn,
%                        note u0B not > 0

c = char(num2str(aa),‘0’,‘0’,num2str(bb)); %thermal tensor

h=.025
tlist = [0:h:2] +[0,h*rand(1,80)]; %mean timestep is 1.5 * h
vA=parabolic(u0A,tlist,b,p,e,t,c,0,0,1);
vA=vA(index,:);
vB=parabolic(u0B,tlist,b,p,e,t,c,0,0,1);
vB=vB(index,:);
vA = vA + stdev.*vA.*randn(size(vA));
vB = vB + stdev.*vB.*randn(size(vB));

%% NOTE:  if aa,bb O(10),  h=.0025, replace 0:h:2 by 0:h:.2
%%        if aa,bb O(100), h=.00025, replace 0:h:2 by 0:h:.02
%%        if aa,bb O(.1),  h=.25, replace 0:h:2 by 0:h:20
```

## Appendix D. Matlab script q2 using temperature probe data to estimate $\lambda_1$ and $\lambda_2$.

```
% Assumes q1  already executed. (So tlist, vA, vB have all
% been calculated.)
% Output is best lam1est and best lam2est.
%   Given a time-series vC asymptotic (large t) to
%   const*exp(-lambda t),
%   we here estimate lambda.
%   (Do this for vA to estimate lam1,
%   and vB, to estimate lam2.)
% After this script run q3 to find aa1, bb1
% (aa1 & bb1 are estimates of thermal tensor).
lam1vec=[];
lam2vec=[];

for k=1:numel(tlist)-2 % 2 could be bigger
    delta=tlist(k+1)-tlist(k);
    if vA(k)<0 || vA(k+1)<0, continue,end
    lam1est=-log(vA(k+1)/vA(k))/delta;
    if vB(k+1)/vB(k)<0, continue, end
    lam2est=-log(vB(k+1)/vB(k))/delta;
    lam1vec=[lam1vec;lam1est]; % col vec of lam1 values
    lam2vec=[lam2vec;lam2est]; % col vec of lam2 values
end

plot(lam1vec,lam2vec,‘ko’); % visualize cluster
C=[lam1vec,lam2vec]; % C has 2 columns, many rows of data
```

```
% Find best point in cluster by repeatedly removing
% outlier rows of C until one or two rows remain.
% If 2 rows, take mean and quit:

while not(isvector(C))
    M=mean(C);
    lam1est=M(1);lam2est=M(2);
    disp(M)
    if size(C,1)==2, return, end  %(C has two rows, can't go further)
    C=remoutliers(C,1);%drop rows having an entry with abs(zscore)>1
end
lam1est=C(1);lam2est=C(2);
disp(C)
```

## Appendix E. Matlab script q3 estimating thermal tensor eigenvalues $a$, $b$ from $\lambda_1$ and $\lambda_2$.

```
% Assumes q1 and q2 already executed.
% (So lam1est,lam2est have been calculated.)
% Output of this script is the estimate (aa1,bb1) of (aa,bb).

% The m-file shandata2d3.m simply defines arrays epsshan, lam1shan,
% and lam2shan as the first three columns of the table in
% Appendix \ref{A:2Dtable}. Then it defines ratio12 as
% lam1shan./lam2shan.

shandata2d3

epsest = interp1(ratio12,epsshan,lam1est/lam2est);
epssqest = epsest^2;
lam1tilde = interp1(epsshan,lam1shan,epsest);
bb1 = lam1est/lam1tilde
aa1 = epssqest*bb1
```

## Appendix F. Matlab script for the function *remoutliers*.

```
function B = remoutliers(A,NSD)
% NSD= number of st devs above which data is considered to be outlier
n=size(A,1); % A has two columns (aa column and bb column) and n rows
mu=mean(A);
sigma=std(A);
MeanMat=repmat(mu,n,1);
SigmaMat=repmat(sigma,n,1);
outliers=abs(A-MeanMat)>NSD*SigmaMat;
% nout=sum(outliers);
A(any(outliers,2),:)=[];
B=A;
end

(This script was adapted from the MathWorks documentation
http://www.mathworks.com/help/techdoc/data_analysis/f0-7275.html.)
```

## REFERENCES

[1] F. M. Arscott, "Periodic Differential Equations," Pergamon Press, 1964.

[2] G. Backstrom and J. Chaussy, *Determination of thermal conductivity tensor and heat capacity of insulating solids*, J. Phys. E: Sci. Instrum., **10** (1977), 767–769.

[3] U. Brydsten and G. Backstrom, *Hot strip determination of the thermal conductivity tensor and heat capacity of crystals*, Int'l. J. Thermophysics, **4** (1983), 369–387.

[4] H. S. Carslaw and J. C. Jaeger, "Conduction of Heat in Solids," 2nd edition, Oxford University Press, 1959.

[5] V. Isakov, "Inverse Problems for Partial Differential Equation," Springer, 1998.

[6] J. Li, S. Rosencrans, X. Wang and K. Zhang, *Asymptotic analysis of a Dirichlet problem for the heat equation on a coated body*, Proceedings of the American Math. Society, **137** (2009), 1711–1721.

[7] J. M. Macleod, *Instructions for operating the divided bar apparatus for thermal conductivity measurement at the GSC, calgary*, Open File **3444**, Geological Survey of Canada, (1997).

[8] V. Pereyra and G. Scherer, *Exponential data fitting*, in "Exponential Data Fitting and Its Applications," 1–26, Bentham, (2010).

[9] Per Sundqvist, *Exponential curve-fitting without start-guess*, http://www.mathworks.com/matlabcentral/fileexchange/21959.

Received August 2011; revised March 2012.

*E-mail address*: srosenc@tulane.edu
*E-mail address*: xdw@math.tulane.edu
*E-mail address*: szhao@bama.ua.edu