F. Berntsson · L. Eldén

**Numerical Solution of Cauchy Problems for Elliptic Equations in “Rectangle-like” Geometries**

**Abstract** We consider two dimensional inverse steady state heat conduction problems in complex geometries. The coefficients of the elliptic equation are assumed to be non-constant. Cauchy data are given on one part of the boundary and we want to find the solution in the whole domain. The problem is ill–posed in the sense that the solution does not depend continuously on the data.

Using an orthogonal coordinate transformation the domain is mapped onto a rectangle. The Cauchy problem can then be solved by replacing one derivative by a bounded approximation. The resulting well–posed problem can then be solved by a method of lines.

A bounded approximation of the derivative can be obtained by differentiating a cubic spline, that approximate the function in the least squares sense. This particular approximation of the derivative is computationally efficient and flexible in the sense that its easy to handle different kinds of boundary conditions.

This inverse problem arises in iron production, where the walls of a melting furnace are subject to physical and chemical wear. Temperature and heat–flux data are collected by several thermocouples located inside the walls. The shape of the interface between the molten iron and the walls can then be determined by solving an inverse heat conduction problem.

In our work we make extensive use of Femlab for creating test problems. By using FEMLAB we solve relatively complex model problems for the purpose of creating numerical test data used for validating our methods. For the types of problems we are intrested in numerical artefacts appear, near corners in the domain, in the gradients that Femlab calculates. We demonstrate why this happen and also how we deal with the problem.

**Keywords** Ill–posed · Cauchy Problem · Elliptic Equation

1 Introduction

We consider the mathematical problem of determining the shape of an unknown boundary, motivated from the following industrial example: An illmenite iron furnace is operated continuously for several years. During this time the furnace material is worn out by the contact with the molten iron. To avoid accidents, where molten iron breaks through, it is necessary to monitor the furnace walls. The cross section of the furnace is illustrated in Figure 1.

The problem is to find the inner boundary of the furnace, based on temperature and heat–flux measurements along a curve inside the furnace wall. In an idealized setting the problem can be formulated as follows: The temperature, \( u = u(x, y) \), satisfies an elliptic partial differential equation,

\[
(a(u)u_x)_x + (a(u)u_y)_y = 0 \quad \text{in } \Omega,
\]

where \( \Omega \) is a 'rectangle-like' region, as seen in Figure 2, and with Cauchy data given on the lower boundary \( L_1 \),

\[
u = g, \quad \text{and, } \frac{\partial u}{\partial n} = h, \quad \text{on } L_1,
\]

and with no-flux conditions on the lateral boundaries,

\[
\frac{\partial u}{\partial n} = 0, \quad \text{on } L_2 \quad \text{and} \quad L_3,
\]

Since the thermal properties of the material depend on temperature the problem is non–linear. Our goal is to determine the location of the upper boundary \( L_4 \), given the additional knowledge that the temperature along the curve \( L_4 \) is constant, and equal to the temperature of the liquid iron. In our case 1450°C.
where the height, \( h \), of the rectangle depend on the geometry of the auxiliary domain, and the derivative \( \phi' \) is defined appropriately. The conformal mapping for polygonal regions is given explicitly by the Schwarz–Christoffel mapping formula. The mapping of an orthogonal grid on the rectangle onto the auxiliary domain (containing \( \Omega \)) is illustrated in Figure 3.

The Cauchy problem (4) is then reformulated as a system of ordinary differential equations, and solved using a standard ODE solver, e.g. the MATLAB routine `ode45`.

Cauchy problems for Elliptic equations are well–known to be ill–posed in the sense that the solution does not depend continuously on the data. Hence regularization is needed. This is discussed briefly in Section 2. In our work we use FEMLAB for simulating measurements, and thus creating numerical test data for validating our algorithm. This is discussed in Section 3. In Section 4 we apply our algorithm for solving the boundary identification problem for a test case.

### 2 Ill–posedness and Stabilization

The instability of the Cauchy problem for the Laplace equation was originally demonstrated by Hadamard, see e.g. [6]. The ill–posedness can be seen by studying the following model problem in the unit square:

\[
\begin{align*}
    u_{xx} + u_{yy} &= 0, & 0 < x < 1, 0 < y < 1 \\
    u(x, 0) &= g(x), & 0 < x < 1 \\
    u_y(x, 0) &= h(x), & 0 < x < 1 \\
    u(x, y) &= 0, & 0 < y < 1,
\end{align*}
\]

By separation of variables the solution is found to be,

\[
u(x, y) = A_0 y + B_0 + \sum_{k=1}^{\infty} \left( A_k e^{k\pi y} + B_k e^{-k\pi y}\right) \cos(k\pi x),
\]

where the sequences \( \{A_k\} \) and \( \{B_k\} \) are determined by the initial conditions. Since \( \exp(k\pi y) \to \infty \) as \( k \to \infty \), small errors in the coefficients \( A_k \) are magnified, and could completely dominate the solution. Numerically this means that very small errors in the data can result in large errors in the computed solutions. Since, in our case, the data is obtained through measurements errors are unavoidable.

One may introduce a “cut off” frequency \( \eta \) and instead compute an approximate solution by including only the terms, \( k < \eta \), in the series. This does stabilize the computations.

A more detailed analysis shows that the fundamental reason for this instability is that the derivative \( \partial/\partial \xi \) is an unbounded operator [1]. Thus if the \( x \)–derivative is replaced by a bounded approximation then the resulting problem is well–posed, i.e. stable with respect to measurement errors.

This conclusion remains valid also for more general equations, e.g. as in (4), where the coefficients in the equation are temperature dependent. This can be seen by rewriting the Cauchy problem as a system of ordinary differential equations,

\[
\begin{align*}
    \left( \frac{u}{au_y} \right)_y &= \left( \begin{array}{cc} 0 & a^{-1} \\
    -\frac{\partial}{\partial \xi} & \frac{\partial}{\partial \xi} \end{array} \right) \left( \begin{array}{c} u \\
    au_y \end{array} \right), & 0 \leq y \leq 1,
\end{align*}
\]

where as previously \( a = a(u) \). By replacing the derivative \( \partial/\partial \xi \) by a bounded approximation, a well–posed initial value problem is obtained. Note that although the original problem is non-linear the initial–value problem can easily be solved using an explicit method, e.g. a Runge–Kutta method.

We use cubic splines for computing derivatives. This has several advantages. Cubic splines are flexible in the sense that, unlike Fourier methods, they do not require functions to be periodic, also different types of boundary conditions can be imposed. The derivative \( u_\xi \) can be computed, approximately, by finding a cubic spline \( s(x) \),
that approximate \( u(x, y_0) \), for a fixed \( y = y_0 \), in the least-squares sense, and then setting \( u_i(x, y_0) \approx s'(x) \). In the next paragraph we describe this procedure in more detail.

Let \( \alpha_{-3} < \ldots < \alpha_{N+2} \) be a coarse grid. The set of all cubic splines defined on the interval \([\alpha_0, \alpha_{N-1}]\) can be expressed using \( B\)-spline basis functions \[3\]. More precisely

\[
s(x) = \sum_{j=1}^{N} c_j B_j(x - \alpha_j),
\]

where \( B_j(x) \) are the \( B\)-spline basis functions, illustrated in Figure 4. A cubic spline \( s(x) \) that approximates the function \( u(x, y_0) \), in the least squares sense, can be found by solving

\[
\min_{c \in \mathbb{R}^{N+2}} \| s(x) - u(x, y_0) \|_2.
\]

In a discrete setting the function \( u(x, y_0) \) is only known on a grid \( \{x_k\} \) and (9) is solved as an ordinary linear least squares problem.

The number of \( B\)-spline basis functions that are used controls the accuracy of the computed derivatives. If too many basis functions are used, so that the derivatives are computed very accurately, the computed solutions of the Cauchy problem will be extremely unstable, with respect to errors in the initial data. Thus, the size of the coarse grid \( \{\alpha_j\} \), i.e. the parameter \( N \), acts as a regularization parameter \[4, 5\], that stabilizes the computations.

3 A Numerically Generated Test Problem

Although this work is focused on solving the inverse problem of identifying the unknown inner boundary of the furnace the creation of good numerical test problems is equally important.

Our algorithm, for solving the Cauchy problem (4), could easily be used also for creating the numerical test data, i.e. for simulating the temperature measurements inside the furnace walls. However, doing so would mean that we risk that numerical errors become systematic. It could happen that the numerical errors done when solving the boundary identification problem are cancelled out by identical errors introduced when creating the test data. As a result our numerical algorithm, for indentifying the unknown boundary, might appear to be more accurate than is actually the case.

Thus it is important to use unrelated numerical methods for creating, and for solving, test problems. In our case we choose to use FEMLAB for creating test data.

The test data was computed as follows: A FEMLAB model of the direct problem, i.e. the equation (1), and boundary conditions (2)–(3), was created. The coefficient \( a(u) \) is the thermal conductivity of the material used inside the furnace walls. By solving the direct problem we obtain the heat-flux \( \partial u / \partial n \), along the lower bound-

Fig. 4 The basis functions \( B_j^3(x) \), for \( j = -1, \ldots, 5 \). Every cubic spline, defined on the interval \([\alpha_0, \alpha_4]\) can be written, uniquely, as a linear combination of the basis functions \( \{B_j^3\} \).

Fig. 5 The thermal conductivity for Magnesia brick, as a function of temperature.

Fig. 6 Test problem generated in FEMLAB. The upper boundary at 1450°C and the lower boundary at 800°C. The sides are insulated. We display the computational mesh (upper graph) and the calculated solution (lower graph). The mesh actually used in the computations were refined further.
ary $L_1$. This is the data we use for identifying the upper boundary of the domain $\Omega$. Measurement errors are simulated by adding random noise to test data. Both the FEMLAB model, and the computed solution, are illustrated in Figure 6.

In practice the boundary of the domain $\Omega$ would be piecewise smooth, i.e. the lower boundary $L_1$ is a smooth curve. In our test problem we assigned a constant temperature, $u=800^\circ C$, along the curve $L_1$. Mathematically this means that the normal derivative $\partial u/\partial n$ should be a smooth curve. However it turns out that this is not the case.

The normal derivative, along the lower boundary $L_1$, as calculated by FEMLAB, is displayed in Figure 7. The computed derivative contains several sharp spikes. These are artifacts that appear in the calculated normal derivative because the, originally piecewise smooth, domain $\Omega$ is approximated by a polygonal region.

Since, for the rectangular region the boundaries are piecewise smooth (i.e. straight lines), the normal derivative, $v_0(\xi,0)$, is a smooth curve. The heat–flux data, for the equations (1) and (4) respectively, are related by the conformal mapping [2],

$$\frac{\partial u}{\partial n}|_{L_1} = (\phi^{-1})'| \frac{\partial v}{\partial n}(\xi,0)$$  \hspace{1cm} (10)

The derivative $(\phi^{-1})'$ of the Schwarz–Christoffel mapping function contain a singularity, of the type $(r-r_0)^{-a}$, at every corner of the polygonal region. This explains the numerical artifacts seen in Figure 7.

Note that when we apply the conformal mapping for computing the initial data $v_0(\xi,0)$, for the rectangular region, the singularities present in the mapping function, and in the normal derivative, that FEMLAB computes, almost cancel out, and the result is a curve that is smooth; except for large computational errors at certain locations. This is because FEMLAB cannot resolve the singularities, close to the corners of the polygonal domain, with sufficient accuracy.

4 Numerical Results

In this section we apply our method for identifying the unknown boundary to the test problem described in the previous section. Given the temperature $u=800^\circ C$, and the heat–flux $h=\partial u/\partial n$ along the lower boundary curve $L_1$. Both the temperature and heat–flux were sampled at 320 equally spaced points on the curve $L_1$. In order to simulate measurement errors normally distributed noise with variance $\sigma^2=30^\circ C$, was added to these data vectors. Thus the noise level is realistic for the application we have in mind.

The identification is done as follows: First the heat–flux data, given at the boundary $L_1$ on polygonal domain, was mapped onto the rectangle using the conformal mapping. The data points closest to the corners were ignored, and finally the data vectors were resampled on an equidistant grid of size $n=200$, by using a least–squares approximating cubic spline. This gives us the initial–boundary conditions for the Cauchy problem (4).

The Cauchy problem is then rewritten as an initial-value problem for a system of ODE’s, as seen in (7), and solved, initially using a standard Runge–Kutta code, i.e. ode45 in MATLAB, and switching to the explicit Euler method in order to simplify the boundary identification. For the purpose of stabilizing the computations the derivatives, $u_0(x,y_0)$, were approximated using $B$–splines, and a course grid $\{\xi_k\}$, of an appropriate size.

The unknown boundary curve $L_4$ is identified by the condition that the temperature at the inner wall of the furnace is equal to $1450^\circ C$. The identified isotherm is then mapped back into the original domain using the conformal mapping.

The computed temperature curves, at levels $y=0.6$ and $y=0.8$, obtained using a coarse grid of size $N=8$, i.e. by using 11 $B$–splines, for approximating the derivatives, are displayed in Figure 8. The identified boundary is displayed in Figure 9.

Computed temperature curves using 19 $B$–splines, for approximating the derivatives, are displayed in Figure 10. This results in less stabilization beeing applied, and the result is an oscillating solution, that cannot be used for identifying the unknown boundary accurately.
Fig. 9 The identified boundary, using $N=8$, i.e. using 11 $B$–spline basis functions.

Fig. 10 Computed temperature curves for $y=0.62$ (left graph) and $y=0.81$ (right graph). Here 19 $B$–splines were used, resulting in a too accurate approximation of the derivative $\partial/\partial x$, and a wildly oscillating solution.

5 Conclusions

In this paper we have presented a numerical algorithm for determining an unknown part of the boundary for an elliptic equation. The proposed method is based on transforming the original “rectangle–like region” to a rectangle using a conformal mapping. The problem on the rectangle is then solved by rewriting the boundary value problem as an initial value problem for a system of ordinary differential equations, that can be solved using standard methods. Since we use an explicit method for integrating the initial value problem non–linear equations can be treated easily.

The coordinate transformation is cheap to compute for rather general geometries, that can be approximated by polygons.

The problem is severely ill–posed, and the solution does not depend continuously on the data. In our algorithm stability is restored by replacing a derivative in the differential equation by a bounded approximation, based on cubic splines.

The numerical experiments demonstrate that the proposed method works well. The parameters in the test were selected so that their values are realistic for the application of determining the thinkness of the furnace walls.

References