ARGONNE NATIONAL LABORATORY
9700 South Cass Avenue
Argonne, Illinois 60439

THE MINPACK-2 TEST PROBLEM COLLECTION

Brett M. Averick[†], Richard G. Carter[†], Jorge J. Moré, and Guo-Liang Xue[†]

Mathematics and Computer Science Division

Preprint MCS-P153-0692

June 1992

Work supported by the Applied Mathematical Sciences subprogram of the Office of Energy Research, U.S. Department of Energy, under Contract W-31-109-Eng-38.

[†] Army High Performance Computing Research Center, University of Minnesota, 1100 Washington Avenue South, Minneapolis, Minnesota 55415. Work supported in part by the United States Army under contract DAAL03-89-C-0038

ABSTRACT

Optimization software has often been developed without any specific application in mind. This generic approach has worked well in many cases, but as we seek the solution of larger and more complex optimization problems on high-performance computers, the development of optimization software should take into account specific optimization problems that arise in a wide range of applications. This observation was the motivation for the development of the MINPACK-2 test problem collection. Each of the problems in this collection comes from a real application and is representative of other commonly encountered problems. There are problems from such diverse fields as fluid dynamics, medicine, elasticity, combustion, molecular conformation, nondestructive testing, chemical kinetics, lubrication, and superconductivity.

Contents

THE MINPACK-2 TEST PROBLEM COLLECTION

Brett M. Averick*, Richard G. Carter*, Jorge J. Moré, and Guo-Liang Xue*

1 Introduction

The Army High Performance Computing Research Center at the University of Minnesota and the Mathematics and Computer Science Division at Argonne National Laboratory have collaborated on the development of the software package MINPACK-2. As part of the MINPACK-2 project, we have developed a collection of significant optimization problems to serve as test problems for the package. This report describes the problems in this collection.

Optimization software has often been developed without any specific application in mind. This generic approach has worked well in many cases, but as we seek the solution of larger and more complex optimization problems on high-performance computers, the development of optimization software should take into account specific optimization problems that arise in a wide range of applications. This observation was the motivation for the main requirement for inclusion in this collection: each problem must come from a real application and be representative of other commonly encountered problems. Problems in the collection come from such diverse fields as fluid dynamics, medicine, elasticity, combustion, molecular conformation, nondestructive testing, chemical kinetics, lubrication, and superconductivity.

Our interest in high-performance computers was the reason for the second requirement for inclusion in this collection: each application selected must lead to a large-scale optimization problem. Many of the problems in the collection are finite dimensional approximations to problems that are naturally expressed in an infinite dimensional setting. Thus, the solution of these problems usually requires the solution of an optimization problem with a large number of variables.

We have also included in this collection small dimensional application problems with interesting features. In particular, we have included difficult problems that are of especial use in testing the robustness of an optimization algorithm. Although such problems may not be large-scale with respect to problem dimension, they can be computationally intensive and difficult to parallelize.

The optimization problems in this collection are divided into three broad categories: systems of nonlinear equations, nonlinear least squares, and general minimization problems. Most of the problems in the collection either are unconstrained or have only upper and lower

Work supported in part by the Applied Mathematical Sciences subprogram of the Office of Energy Research of the U.S. Department of Energy under Contract W-31-109-Eng-38.

*Army High Performance Computing Research Center, University of Minnesota, 1100 Washington Avenue South, Minneapolis, Minnesota 55415. Work supported in part by the United States Army under contract DAAL03-89-C-0038

bounds on the variables. We are also interested in large-scale optimization problems with more general constraints, but our efforts have focused on bound constrained problems, in view of our current work in developing software to solve these problems.

The effort needed to develop a large-scale problem for this collection can be considerable. Inclusion of a problem in the collection requires code for the evaluation of the functions associated with the application, and verification that the code actually represents the specified application. Our experience in developing these problems has shown that the verification process is important, because in several cases this process has unveiled errors in the description of the application. We have tried to minimize the effort needed to include a problem in this collection by concentrating on model applications that can be described (at least superficially) in two pages.

We also emphasize that we are developing code for the evaluation of the functions and the associated derivatives in Fortran 77 to enhance portability. This is an important part of our effort. There are several collections of interesting optimization problems, but in many cases software for these problems is either not available or is available in a restricted format.

The primary purpose of this collection is to provide difficult test cases for MINPACK-2. We are interested in examining the following issues:

> How robust is the software with respect to poor initial approximations?
>
> How does the software perform on badly scaled problems?
>
> How robust is the code with respect to noise in the user-supplied software?
>
> How does the software perform on large-scale problems?
>
> How does the software perform on diverse vector and parallel architectures?

A complete discussion of these issues is not in the scope of this report, but we mention that with these problems it is entirely appropriate to use computing time as a measure of efficiency: the computational expense (as measured by the number of floating-point operations) of evaluating the functions in this collection is relatively small, roughly the same order as the number of variables in the problem. For optimization software designed for cases in which the expense of the user-supplied software is dominant, these same test problems can be used to evaluate the software by using the number of calls to the user-supplied software as a measure of efficiency.

In the remainder of this report we describe the problems in this collection. We have not attempted to provide a detailed description of the applications. The emphasis is on the mathematical formulation of the application as an optimization problem. We provide background information on the application; details that are not needed to understand the formulation of the application have been omitted.

2 Systems of Nonlinear Equations

The solution to a system of nonlinear equations specified by a mapping $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is a vector $x \in \mathbb{R}^n$ such that $f(x) = 0$. Algorithms for systems of nonlinear equations usually approach this problem by seeking a local minimizer to the problem

$$\min\{\|f(x)\| : x_l \leq x \leq x_u\},$$

where x_l and x_u are bounds on the solution x, and $\|\cdot\|$ is some norm on $\mathbb{R}^n$. Most algorithms use the l_2 norm. Interestingly enough, codes for systems of nonlinear equations do not tend to have provisions for handling bounds (or more general constraints) on the variables.

2.1 Flow in a Channel

The problem of fluid injection through one side of a long vertical channel leads to the boundary value problem

$$u'''' = R[u'u'' - uu'''], \qquad 0 \leq t \leq 1,$$

$$u(0) = u'(0) = 0, \quad u(1) = 1, \quad u'(1) = 0,$$

where u is the potential function, u' is the tangential velocity of the fluid, and R is the Reynolds number. This problem is interesting because it is easy to solve for small Reynolds numbers but becomes increasingly difficult to solve A plot of the computed tangential velocity u' for several values of R appears in Figure 2.1.

This problem was formulated by Huang [21]. In our formulation we have followed Ascher, Mattheij, and Russell [3, p. 7].

We solve this nonlinear boundary value problem by a k-stage collocation method. Consider the general boundary value problem

$$u^{(m)}(t) = F(t, u(t), u'(t), \cdots, u^{(m-1)}(t)), \qquad t \in (a, b),$$

with m total boundary conditions given at $t = a$ and $t = b$, and let

$$a = t_1 < t_2 < \cdots < t_{n_0} < t_{n_0+1} = b$$

be a partitioning of $[a, b]$, with $h_i = t_{i+1} - t_i$. A k-stage collocation method is defined in terms of k points

$$0 < \rho_1 < \rho_2 < \cdots < \rho_k < 1.$$

We choose the collocation points ρ_i as the roots of the Legendre polynomial of order $k \geq m$. This choice guarantees that superconvergence occurs at the mesh points t_i. The k-stage collocation method approximates the solution to the boundary value problem by a piecewise

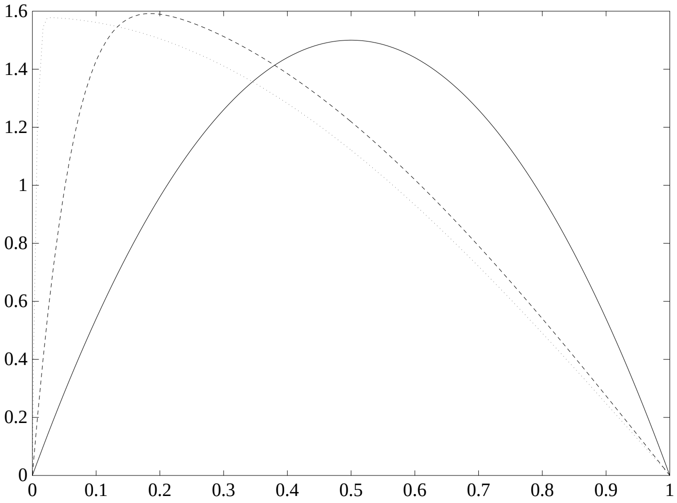

Figure 2.1: Tangential velocity u' for $R = 0, 10^2, 10^4$ (solid, dashed, dotted)

polynomial u_π, where u_π is a polynomial of order $m+k$ in each subinterval $[t_i, t_{i+1}]$. Thus, u_π is defined in terms of $n_0(m+k)$ parameters. We specify these parameters by requiring that $u_\pi \in C^{m-1}[a,b]$, that u_π satisfy the m given boundary conditions, and that u_π satisfy the differential equation at the collocation points

$$\xi_{ij} = t_i + h_i\rho_j, \qquad 1 \le i \le n_0, \quad 1 \le j \le k.$$

The piecewise polynomial approximation u_π in the interval $[t_i, t_{i+1}]$ is of the form

$$\sum_{j=1}^{m} \frac{(t-t_i)^{j-1}}{(j-1)!} v_{ij} + h_i^m \sum_{j=1}^{k} \phi_j(\frac{t-t_i}{h_i}) w_{ij}, \qquad 1 \le i \le n_0,$$

where we choose the basis representation (Ascher, Mattheij, and Russell [3, pp. 247–249])

$$\phi_j(t) = \frac{t^{m+j-1}}{(m+j-1)!}, \qquad 1 \le j \le k.$$

A simple computation shows that in this representation

$$v_{ij} = u_\pi^{(j-1)}(t_i), \qquad 1 \le i \le n_0, \quad 1 \le j \le m,$$

and that

$$w_{ij} = h_i^{j-1} u_\pi^{(m+j-1)}(t_i), \qquad 1 \le i \le n_0, \quad 1 \le j \le k.$$

Thus, bounds on a derivative of u_π at t_i can be specified by bounding v_{ij} or w_{ij}.

We guarantee that $u_\pi \in C^{m-1}[a,b]$ by enforcing continuity at the interior grid points. The continuity equations are thus given by

$$u_\pi^{(l-1)}(t_i^-) = u_\pi^{(l-1)}(t_i^+), \qquad 1 \le l \le m, \quad 1 < i \le n_0.$$

The collocation equations are then

$$u_\pi^{(m)}(\xi_{ij}) = F(\xi_{ij}, u_\pi(\xi_{ij}), u'_\pi(\xi_{ij}), \ldots, u_\pi^{(m-1)}(\xi_{ij})), \qquad 1 \le j \le k, \quad 1 \le i \le n_0.$$

These equations, together with the m boundary conditions, lead to a system of $n_0(m+k)$ equations in the $n_0(m+k)$ unknowns v_{ij} and w_{ij}.

We choose $k = 4$ and $n_0 = 40$ in our numerical results. This choice leads to a system of nonlinear equations with 320 variables. A plot of the computed tangential velocity u' for several values of R appears in Figure 2.1; similar results were obtained by Ascher [2] with $k = 5$. Note, in particular, the steep gradient near $t = 0$ as R increases.

The three plots in Figure 2.1 were generated by solving a sequence of five problems with $R_0 = 0$, $R_1 = 10^2$, $R_2 = 10^3$, $R_3 = \frac{1}{2}10^4$, and $R_4 = 10^4$. The problem with $R_0 = 0$ is linear; its solution is used as the initial approximation to the problem for R_1. The continuation process continues in this manner, with the initial approximation to the problem for R_{i+1} being the solution to the problem for R_i. The whole continuation process requires 26 function evaluations. An interesting observation is that with the continuation process we were able to obtain the solution for $R = 10^4$, but this solution was not obtainable if we started from the solution for $R = 0$.

2.2 Swirling Flow between Disks

The steady flow of a viscous, incompressible, axisymmetric fluid between two rotating, infinite coaxial disks, located at $t = 0$ and $t = 1$, yields the boundary value problem

$$\epsilon f'''' + f f''' + g g' = 0, \quad \epsilon g'' + f g' + f' g = 0, \qquad 0 \le t \le 1,$$

$$f(0) = f'(0) = f(1) = f'(1) = 0, \quad g(0) = \Omega_0, \quad g(1) = \Omega_1$$

where f' is radial velocity, g is angular velocity (Ω_0 and Ω_1 are the angular velocities of the infinite disks), and $0 \le \epsilon \ll 1$ is a viscosity parameter. This problem is interesting since it is easy to solve for ϵ close to 1 but becomes increasingly difficult to solve as ϵ decreases. Plots of the computed radial velocity f' and the computed angular velocity g appear in Figures 2.2 and 2.3, respectively. In these plots, $\Omega_0 = -1$ and $\Omega_1 = 1$.

In our formulation of the swirling flow problem we have followed Parter [36]. (We note that there is a typographical error in the formulation of Ascher, Mattheij, and Russell [3, p. 23]; in this reference the first equation is $\epsilon f'''' + f''' + g' = 0$.)

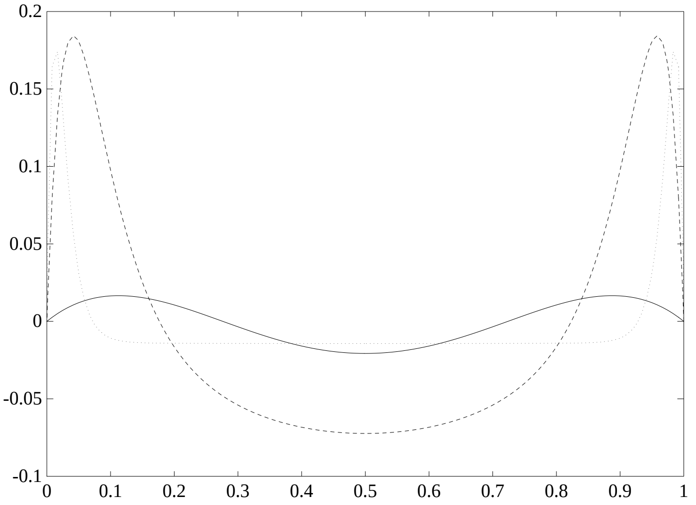

Figure 2.2: Radial velocity f' for $\epsilon = 10^{-1}, 10^{-3}, 10^{-4}$ (solid, dashed, dotted)

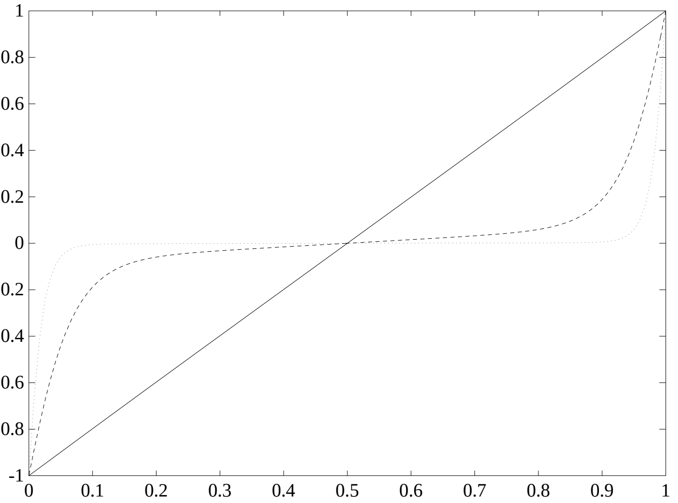

Figure 2.3: Angular velocity g for $\epsilon = 10^{-1}, 10^{-3}, 10^{-4}$ (solid, dashed, dotted)

The swirling flow problem is described by a (coupled) system of boundary value problems. Systems of this type can be solved by a natural extension of the k-stage collocation method discussed above. A k-stage collocation method approximates the solution to a system of p boundary value problems by a vector-valued function $u_\pi : [a,b] \to \mathbb{R}^p$, where the j-th component of u_π is a polynomial of order $m_j + k$ in each subinterval $[t_i, t_{i+1}]$, and m_j is the degree of the j-th boundary value problem. Thus u_π is defined in terms of $n_0(pk + m_0)$ parameters, where m_0 is the sum of all the degrees. In many cases all the boundary value problems have the same degree (for example, in initial value problems), and then u_π is defined in terms of $n_0 p(k+m)$ parameters, where m is the common degree.

The parameters that define u_π are determined by the continuity and collocation equations for each boundary value problem, together with the m_0 boundary conditions. If all the boundary value problem have the same degree, the continuity equations are

$$u_\pi^{(l-1)}(t_i^-) = u_\pi^{(l-1)}(t_i^+), \qquad 1 \le l \le m, \quad 1 < i \le n_0.$$

Note that in this case these are vector equations. Similarly, the collocation equations are

$$u_\pi^{(m)}(\xi_{ij}) = F(\xi_{ij}, u_\pi(\xi_{ij}), u_\pi'(\xi_{ij}), \ldots, u_\pi^{(m-1)}(\xi_{ij})), \qquad 1 \le j \le k, \quad 1 \le i \le n_0.$$

If the boundary value problems have different degrees (as in the swirling flow problem), then these equations have to be modified in an obvious manner since m depends on the component of F.

For the swirling flow problem, $p = 2$ and $m_0 = 6$. For our numerical results, we choose $k = 4$ and $n_0 = 40$. This leads to a system of nonlinear equations with 560 variables. The plots of the computed radial velocity f' and the computed angular velocity g appear in Figures 2.2 and 2.3, respectively. In these plots, $\Omega_0 = -1$ and $\Omega_1 = 1$. For these values of Ω_0 and Ω_1, McLeod and Parter [26] have shown that there is a solution to the swirling flow problem such that the functions f and g are odd functions about $t = \frac{1}{2}$, the function g is strictly monotone on $(0, \frac{1}{2})$, and there is a t_1 in $(0, \frac{1}{2})$ with $f'(t) > 0$ on $(0, t_1)$ and $f'(t) < 0$ on $(t_1, \frac{1}{2})$. These results are confirmed by the plots.

Continuation was used to solve these problems with $\epsilon_1 = 10^{-1}$, $\epsilon_2 = 10^{-3}$, and $\epsilon_3 = 10^{-4}$. The initial approximation to the problem with $\epsilon_1 = 10^{-1}$ was the solution of the boundary value problem with $\epsilon = \infty$. The continuation process required 24 function evaluations.

2.3 Incompressible Elastic Rods

The shape of a thin incompressible elastic rod, or elastica, clamped at the origin and acted on by a vertical force Q, a horizontal force P, and torque M is described by the solution of the boundary value problem

$$\theta'(s) = Qx(s) - Py(s) + M, \quad x'(s) = \cos[\theta(s)], \quad y'(s) = \sin[\theta(s)]$$

subject to the boundary conditions

$$x(0) = y(0) = \theta(0) = 0,$$

where θ is the local angle of inclination, and s is the arc length along the elastica.

The inverse elastic rod problem is to determine Q, P, and M such that $x(\cdot), y(\cdot)$ and $\theta(\cdot)$ solve this boundary value problem and satisfies the boundary conditions

$$x(1) = a, \quad y(1) = b, \quad \theta(1) = c,$$

for specified values of a, b and c. This formulation implies that $x'(s)^2 + y'(s)^2 = 1$, and thus we must have $x(s)^2 + y(s)^2 \leq 1$ for all $s \in [0, 1]$. In particular, if the inverse elastic rod problem has a solution then $a^2 + b^2 \leq 1$. Also note that since θ is the local angle of inclination, setting $\theta(1) = c \geq 2\pi$ yields a rod with one or more loops. See, for example, the solutions shown in Figures 2.4 and 2.5.

The problem can be attacked as a system of three nonlinear equations in the three unknowns Q, P and M defined by

$$f(Q, P, M) = \begin{pmatrix} x(1) - a \\ y(1) - b \\ \theta(1) - c \end{pmatrix}.$$

Given values of Q, P and M, the function can be evaluated by using a differential equation solver to obtain values for $x(1)$, $y(1)$, and $\theta(1)$. This approach can be quite challenging because the elastica is sensitive to end conditions, especially for more complicated shapes which require large forces and torques. See the discussion by Watson and Wang [40].

We solve the inverse elastic rod problem with a k-stage collocation method. Recall that the k-stage collocation method is defined in terms of a partition π:

$$a = t_1 < t_2 < \cdots < t_{n_0} < t_{n_0+1} = b$$

of $[a, b]$, and a set of k collocation points in each interval $[t_i, t_{i+1}]$. The collocation method approximates the solution of the system of differential equations by a vector valued function $u_\pi : [a, b] \rightarrow R^p$, where each component of u_π is a polynomial of order $k + 1$ in each subinterval $[t_i, t_{i+1}]$. Thus u_π is defined in terms of $n_0 p(k + 1)$ parameters, where p is the number of equations. For the inverse elastic rod problem $p = 3$.

In our formulation of the inverse elastic rod problem, we use $x \in R^n$ as the vector which defines u_π with its first $n - 3$ components, and defines Q, P and M with its last 3 components, where $n = 3n_0(k + 1) + 3$. We have used $k = 4$ so that we have $n = 15n_0 + 3$ equations in n unknowns.

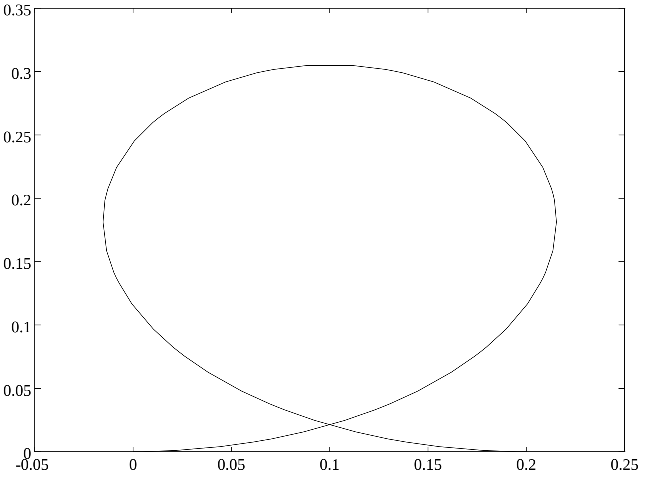

Figure 2.4: Solution to the elastic rod problem with $(a, b, c) = (0.2, 0.0, 2\pi)$.

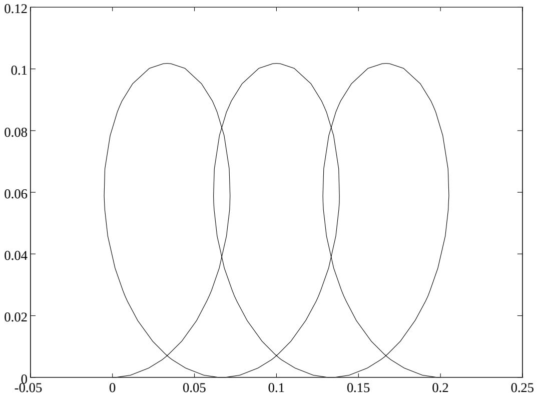

Figure 2.5: Solution to the elastic rod problem with $(a, b, c) = (0.2, 0.0, 6\pi)$.

Solutions to the elastic rod problem can be obtained by using continuation. For example, Figure 2.4 is obtained by a two phase continuation process. In the first phase we solve a sequence of problems with

$$a = \sin(c)/c, \quad b = (1 - \cos(c))/c, \quad c \in (0, 2\pi).$$

Once we obtain the solution to $(a, b, c) = (0, 0, 2\pi)$, we fix b and c, and gradually increase a to obtain Figure 2.4. A similar process yields Figure 2.5. For this figure the initial continuation phase is continued until $c = 6\pi$; the second phase proceeds as before. This continuation process can be interpreted as gradually bending the rod to the desired shape.

In general, the elastic rod problem does not have a unique solution. This is certainly the case if $c = 0$; in this case the solution $(x(\cdot), -y(\cdot), -\theta(\cdot))$ can be obtained by reflection about the y-axis. There can also be lack of uniqueness if $c \neq 0$. Indeed, an interesting aspect of the elastic rod problem is that different solutions can often be obtained by selecting different continuation paths. Figure 2.6 presents two different solutions for $(a, b, c) = (0.2, 0.3, 0.1\pi)$. The solid line is a solution corresponding to

$$(Q, P, M) = (55.3\ldots, 31.6\ldots, 9.22\ldots),$$

while the dotted line corresponds to

$$(Q, P, M) = (43.9\ldots, 33.4\ldots, -9.06\ldots).$$

Although the two solutions in Figure 2.6 seem to be similar, a careful examination of the two solution curves shows that they are not related in any obvious way.

2.4 Solid Fuel Ignition

A steady-state model of solid fuel ignition can be described in terms of the solution u_λ of the boundary value problem

$$-\Delta u(x) = \lambda \exp[u(x)], \quad x \in \mathcal{D}, \qquad u(x) = 0, \quad x \in \partial\mathcal{D},$$

where Δ is the Laplace operator, $\mathcal{D}$ is a domain in $\mathbb{R}^2$ with boundary $\partial\mathcal{D}$, and $\lambda \in \mathbb{R}$. The model simulates a thermal reaction process in a rigid material, where the process depends on a balance between chemically generated heat addition and heat transfer by conduction. Aris [1, pages-292-299] and Bebernes and Eberly [4], discuss this problem in the context of combustion problems.

This problem is usually called the Bratu problem in the literature. The main result on the existence and multiplicity of solutions to the Bratu problem shows that there is a $\lambda_{FK} > 0$ (the Frank-Kamenetskii parameter) such that the steady state model has two solutions for $0 < \lambda < \lambda_{FK}$, but that there are no solutions for $\lambda > \lambda_{FK}$.

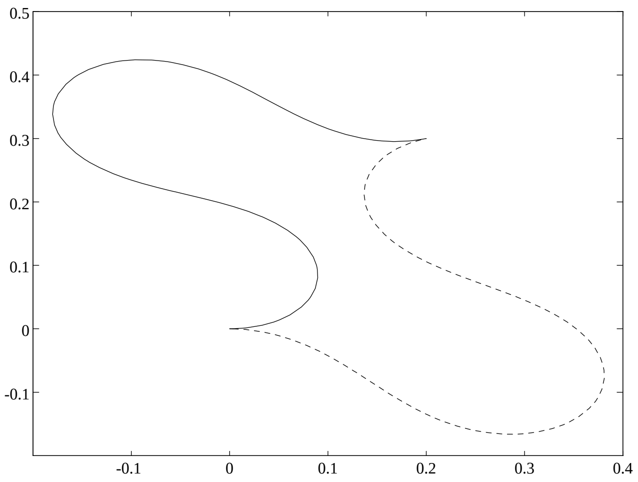

Figure 2.6: Solutions to the elastic rod problem with $(a, b, c) = (0.2, 0.3, 0.1\pi)$.

Figure 2.7 illustrates the lack of uniqueness in the Bratu problem. This figure is a plot of $\|u_\lambda\|_\infty$ when $\mathcal{D}$ is the unit square. Note that λ increases along the solution curve until a limit point is reached at $\lambda = \lambda_{FK} \approx 6.81$. At this point $\|u_\lambda\|_\infty = u(\frac{1}{2}, \frac{1}{2}) \approx 1.39$.

A finite dimensional version of the Bratu problem on a rectangle $\mathcal{D} = (l_1, u_1) \times (l_2, u_2)$ can be obtained via a finite-difference formulation. Vertices $z_{i,j} \in \mathcal{D}$ are determined by choosing grid spacings h_x and h_y and defining

$$z_{i,j} = (l_1 + ih_x, l_2 + jh_y), \qquad 0 \leq i \leq n_x + 1, \quad 0 \leq j \leq n_y + 1,$$

such that $z_{n_x+1, n_y+1} = (u_1, u_2)$. Approximations $u_{i,j}$ to $u(z_{i,j})$ can be obtained by using central differences to approximate the Laplacian operator. This leads to the system of $n = n_x n_y$ nonlinear equations

$$\frac{h_y}{h_x}(2u_{i,j} - u_{i+1,j} - u_{i-1,j}) + \frac{h_x}{h_y}(2u_{i,j} - u_{i,j+1} - u_{i,j-1}) = \lambda h_x h_y \exp[u_{i,j}]$$

where $1 \leq i \leq n_x$ and $1 \leq j \leq n_y$.

The solution of the Bratu problem requires the determination of a path $u(\cdot)$ such that $f(u(\lambda), \lambda) = 0$ where $f : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$ is the mapping that defines the finite dimensional version of the Bratu problem. The most interesting feature of the Bratu problem is that the path $u(\cdot)$ has a turning point at λ_{FK}. We also note that solutions on the lower part of the branch in Figure 2.7 are relatively easy to obtain from the starting point $u_0 = 0$, but solutions on the upper part of the branch are difficult to obtain without a close starting point.

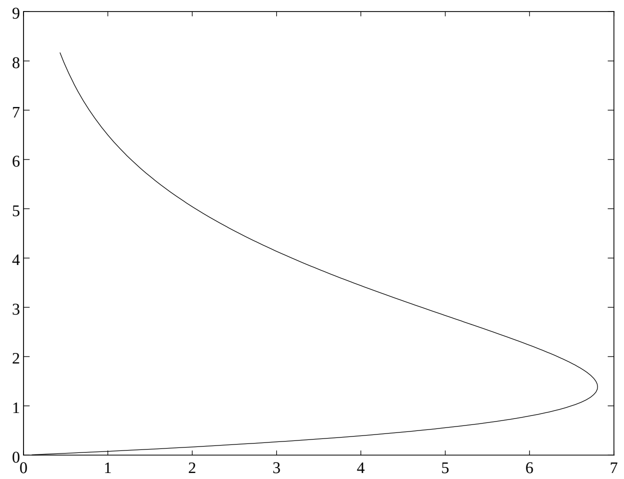

Figure 2.7: $\|u_\lambda\|_\infty$ for the solid fuel ignition problem

The classical approach to the solution of a sequence of nonlinear systems of the form $f(u, \lambda) = 0$ is to solve $f(u, \lambda_k) = 0$ for an increasing sequence of parameters λ_k. This approach fails for the Bratu problem when $\lambda_k > \lambda_{FK}$.

Modern continuation methods do not assume that u is a function of λ. Given a point (u_0, λ_0) on the solution curve, these methods generate a sequence $\{(u_k, \lambda_k)\}$ where (u_{k+1}, λ_{k+1}) solves the augmented system

$$\begin{pmatrix} f(u,\lambda) \\ a_k(u,\lambda) \end{pmatrix} = 0,$$

and the additional equation $a_k(u, \lambda) = 0$ is chosen so that the augmented system has a nonsingular Jacobian matrix. In the arclength continuation method used by Glowinski, Keller, and Reinhart [18], the parameter λ is expressed in terms of arclength s, and the additional equation

$$\frac{(u - u_k)^T (u_k - u_{k-1})}{(s_k - s_{k-1})^2} + \frac{(\lambda - \lambda_k)(\lambda_k - \lambda_{k-1})}{(s_k - s_{k-1})^2} = 1.$$

is imposed. Thus, given a sequence $0 = s_0 < s_1 < \ldots < s_k$, and points (u_0, λ_0), (u_1, λ_1) on the solution curve, the solution path can be obtained by solving a sequence of augmented systems.

Motivation for this approach can be obtained by first noting that $a_k(u, \lambda) = 1$ is a hyperplane in $\mathbb{R}^{n+1}$, orthogonal to the line segment that passes through the points (u_k, λ_k)

and (u_{k-1}, λ_{k-1}). Moreover, this line segment intersects the hyperplane at

$$\Big(u_k + \alpha_k[u_k - u_{k-1}], \lambda_k + \alpha_k[\lambda_k - \lambda_{k-1}]\Big)$$

for $\alpha_k = (s_k - s_{k-1})^2 > 0$. This argument suggests that if $s_k - s_{k-1}$ is sufficiently small, then the augmented system will have a solution.

The solution curve in Figure 2.7 was obtained by this arclength method. Glowinski, Keller, and Reinhart [18] use

$$(2u_k - u_{k-1}, 2\lambda_k - \lambda_{k-1})$$

as the starting point to obtain (u_{k+1}, λ_{k+1}). Our experience indicates that this choice of starting point is not crucial; we have used (u_k, λ_k). However, the continuation process is sensitive to the distance between the initial values (u_0, λ_0) and (u_1, λ_1).

Glowinski, Keller, and Reinhart [18] contains a thorough discussion of continuation methods for the Bratu problem. For additional numerical results on this problem see Kikuchi [22], and Brown and Saad [8].

2.5 Flow in a Driven Cavity

The steady flow of a viscous incompressible fluid in a planar region $\mathcal{D}$ is described by the Navier-Stokes equations

$$-\nu\Delta v + (v \cdot \nabla)v + \nabla p = 0, \qquad \nabla \cdot v = 0$$

where $v : \mathcal{D} \to \mathbb{R}^2$ is the velocity field for the fluid, $p : \mathcal{D} \to \mathbb{R}$ is the pressure, and ν is the viscosity parameter (the reciprocal of the Reynolds number R). In the classical driven cavity problem the region $\mathcal{D}$ is the unit square in $\mathbb{R}^2$, and the boundary conditions are

$$v(\xi_1, \xi_2) = \begin{cases} (0,1) & \text{if } \xi_2 = 1 \\ (0,0) & \text{if } 0 \le \xi_2 < 1 \end{cases}$$

One of the main difficulties associated with this formulation is that there are no boundary conditions on the pressure p. In the stream function-vorticity formulation, the pressure is eliminated, and the problem is expressed in terms of the stream function ψ and vorticity ω. The components v_1 and v_2 of the velocity vector v are expressed in terms of the stream function ψ by

$$v_1 = \partial_y \psi, \qquad v_2 = -\partial_x \psi.$$

Thus, the incompressibility condition $\nabla \cdot v = 0$ is automatically satisfied. The vorticity $\omega \equiv \partial_x v_2 - \partial_y v_1$ is thus given by $-\Delta\psi = \omega$. The stream function-vorticity formulation of the driven cavity problem is then

$$\nu\Delta^2\psi + \Big[(\partial_y\psi)(\partial_x\omega) - (\partial_x\psi)(\partial_y\omega)\Big] = 0, \qquad -\Delta\psi = \omega$$

The formulation of the driven cavity problem in terms of the stream function requires the solution of the boundary value problem

$$\Delta^2\psi - R\Big[(\partial_y\psi)(\partial_x\Delta\psi) - (\partial_x\psi)(\partial_y\Delta\psi)\Big] = 0.$$

with boundary conditions

$$\psi(\xi_1,\xi_2) = \partial_x\psi(\xi_1,\xi_2) = 0, \qquad \partial_y\psi(\xi_1,\xi_2) = \begin{cases} 1 & \text{if } \xi_2 = 1 \\ 0 & \text{if } 0 \le \xi_2 < 1 \end{cases}$$

The finite dimensional formulation of the driven cavity problem follows Schreiber and Keller [37]. Vertices $z_{i,j} \in \mathcal{D}$ are determined by choosing grid spacings h_x and h_y and defining

$$z_{i,j} = (ih_x, jh_y), \qquad 0 \le i \le n_x + 1, \quad 0 \le j \le n_y + 1$$

such that $z_{n_x+1,n_y+1} = (1,1)$. Approximations $u_{i,j}$ to $\psi(z_{i,j})$ are obtained by using central differences to approximate the Laplacian operator and the partial derivatives $\partial_y\psi$ and $\partial_x\psi$. These approximations lead to a system of $n = n_x n_y$ equations in the n unknown $u_{i,j}$ of the form

$$f(u) = Au + b - R\Phi(u),$$

where A is the discrete biharmonic operator, b contains boundary information, and Φ is the discrete representation of the nonlinear term.

An interesting feature of this formulation is that the discrete biharmonic approximation becomes poorly conditioned as the dimension n increases. Thus, the problems becomes difficult to solve for even moderate values of the Reynolds number R. This difficulty can be overcome by preconditioning the problem.

We have followed Brown and Saad [7] and preconditioned f with the fast biharmonic solver `bihar` of Björstad [5] (available from `netlib`), and solved the preconditioned problem

$$A^{-1}f(u) = 0$$

with $n = 10,000$ variables and for various values of the Reynolds number R. Figures 2.8 and 2.9 show the streamlines for $R = 400$ and $R = 1000$, respectively, while Figures 2.10 and 2.11 are the equivorticity contours. These solutions were obtained by the simple continuation process $R = [0, 400, 1000]$.

These figures show a primary vortex in the center of the cavity and two secondary vortices in the corners. The stream function contour lines have ψ values of

$$-.11,\ -.1,\ -.08,\ -.06,\ -.04,\ -.02,\ -.01,$$

$$-.00001,\ .000001,\ .00001,\ .0001,\ .001,\ .002,$$

while the equivorticity contours have $\omega = -\Delta\psi$ values of

$$-5.0,\ -3.0,\ -1.0,\ 1.0,\ 3.0,\ 5.0$$

These are the values used by Schreiber and Keller [37].

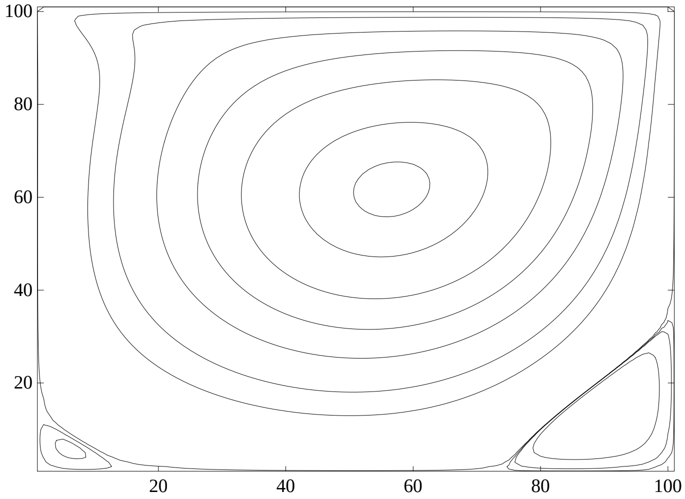

Figure 2.8: Streamlines for $R = 400$

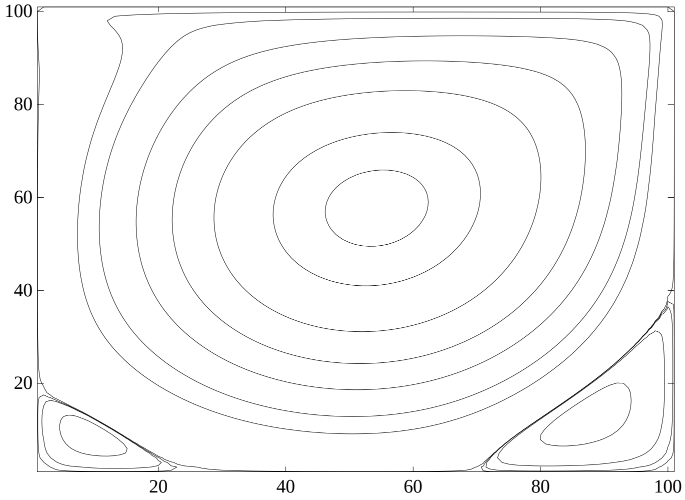

Figure 2.9: Streamlines for $R = 1000$

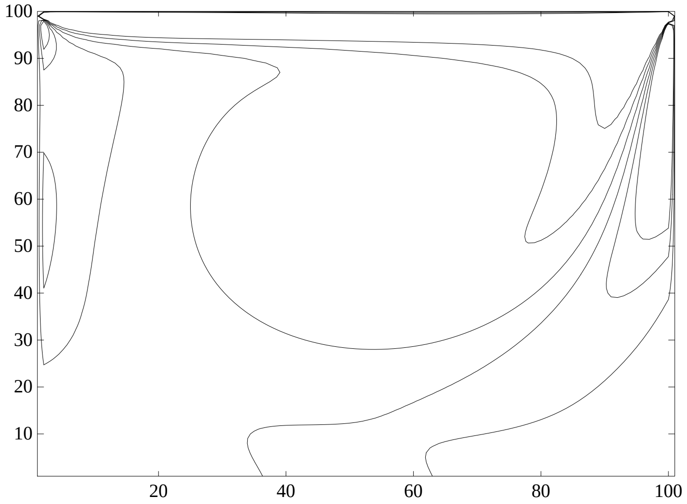

Figure 2.10: Equivorticity lines for $R = 400$

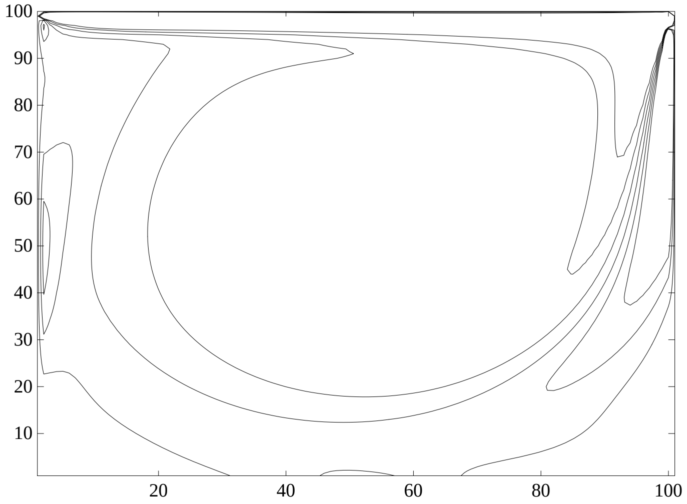

Figure 2.11: Equivorticity lines for $R = 1000$

2.6 Human Heart Dipole

The human heart dipole problem arises in the experimental electrolytic determination of the resultant dipole moment in the human heart. The problem is of the form

$$\begin{aligned}
f_1(x) &= x_1 + x_2 - \sigma_{mx} \\
f_2(x) &= x_3 + x_4 - \sigma_{my} \\
f_3(x) &= x_5 x_1 + x_6 x_2 - x_7 x_3 - x_8 x_4 - \sigma_A \\
f_4(x) &= x_7 x_1 + x_8 x_2 + x_5 x_3 + x_6 x_4 - \sigma_B \\
f_5(x) &= x_1(x_5^2 - x_7^2) - 2x_3 x_5 x_7 + x_2(x_6^2 - x_8^2) - 2x_4 x_6 x_8 - \sigma_C \\
f_6(x) &= x_3(x_5^2 - x_7^2) + 2x_1 x_5 x_7 + x_4(x_6^2 - x_8^2) + 2x_2 x_6 x_8 - \sigma_D \\
f_7(x) &= x_1 x_5(x_5^2 - 3x_7^2) + x_3 x_7(x_7^2 - 3x_5^2) + x_2 x_6(x_6^2 - 3x_8^2) + x_4 x_8(x_8^2 - 3x_6^2) - \sigma_E \\
f_8(x) &= x_3 x_5(x_5^2 - 3x_7^2) - x_1 x_7(x_7^2 - 3x_5^2) + x_4 x_6(x_6^2 - 3x_8^2) - x_2 x_8(x_8^2 - 3x_6^2) - \sigma_F
\end{aligned}$$

with data $\sigma_{mx}, \sigma_{my}, \sigma_A, \sigma_B, \sigma_C, \sigma_D, \sigma_E, \sigma_F$. In this problem $n = 8$.

This problem was formulated by Nelson and Hodgkin [31]. In our formulation we have followed Dennis, Gay, and Vu [11]. They also propose a six-variable version of this problem, obtained by using the first two equations to eliminate two of the first four variables. Further reductions are possible. Indeed, Morgan, Sommese, and Watson [30] showed that the human heart problem can be reduced to the solution of one quadratic equation in one unknown. We have not implemented any of these reduced problems.

There are five versions of the eight-variable problem depending on the data and the starting point. The last three versions can be difficult to solve, even from the standard starting point.

An interesting aspect of this problem is that the system is unchanged if the variables are permuted according to the

$$(1, 2, 3, 4, 5, 6, 7, 8) \rightarrow (2, 1, 4, 3, 6, 5, 8, 7)$$

permutation. This implies that solutions appear in pairs. In all of these problems, we have only found one pair of solutions. If a unique solution is desired, the bound $x_1 \leq 0$ can be imposed. There are no natural bounds associated with this problem, but the components of the solution lie in the interval $[-20, 20]$.

2.7 Combustion of Propane – Full Formulation

This chemical equilibrium problem describes the combustion of propane in air. Each unknown represents the number of moles of a given product formed for each mole of propane;

ten products are considered in this reaction. The problem is of the form

$$
\begin{aligned}
f_1(x) &= x_1 + x_4 - 3 \\
f_2(x) &= 2x_1 + x_2 + x_4 + x_7 + x_8 + x_9 + 2x_{10} - R \\
f_3(x) &= 2x_2 + 2x_5 + x_6 + x_7 - 8 \\
f_4(x) &= 2x_3 + x_9 - 4R \\
f_5(x) &= K_5 x_2 x_4 - x_1 x_5 \\
f_6(x) &= K_6 x_2^{1/2} x_4^{1/2} - x_1^{1/2} x_6 \left(\frac{p}{x_{11}}\right)^{1/2} \\
f_7(x) &= K_7 x_1^{1/2} x_2^{1/2} - x_4^{1/2} x_7 \left(\frac{p}{x_{11}}\right)^{1/2} \\
f_8(x) &= K_8 x_1 - x_4 x_8 \left(\frac{p}{x_{11}}\right) \\
f_9(x) &= K_9 x_1 x_3^{1/2} - x_4 x_9 \left(\frac{p}{x_{11}}\right)^{1/2} \\
f_{10}(x) &= K_{10} x_1^2 - x_4^2 x_{10} \left(\frac{p}{x_{11}}\right) \\
f_{11}(x) &= x_{11} - \textstyle\sum_{j=1}^{10} x_j,
\end{aligned}
$$

with data $K_5, \ldots, K_{10}$, and parameters p and R. The parameter p is the pressure in atmospheres and R expresses the relative amounts of air and fuel. In this problem $n = 11$, $p = 40$, and $R = 10$. In our formulation of this problem we have followed Meintjes and Morgan [27].

This problem may be difficult to solve because of the presence of square roots in the function components and the possibility of generating an iterate with a negative component. There are no difficulties in solving this problem from the standard starting point x_s, but an unconstrained algorithm is likely to generate an iterate with a negative component from the starting point $10x_s$. The bounds $x_j \geq 0$ can be used to solve this problem.

2.8 Combustion of Propane – Reduced Formulation

This chemical equilibrium problem, like the preceding one, describes the combustion of propane in air. This formulation, however, uses the element variables of Meintjes and Morgan [27] to avoid the square roots in the function evaluations. The formulation of the problem in terms of element variables also reduces the problem to a system of the form

$$
\begin{aligned}
f_1(x) &= x_1x_2 + x_1 - 3x_5 \\
f_2(x) &= 2x_1x_2 + x_1 + 2R_{10}x_2^2 + x_2x_3^2 + R_7x_2x_3 + R_9x_2x_4 + R_8x_2 - Rx_5 \\
f_3(x) &= 2x_2x_3^2 + R_7x_2x_3 + 2R_5x_3^2 + R_6x_3 - 8x_5 \\
f_4(x) &= R_9x_2x_4 + 2x_4^2 - 4Rx_5 \\
f_5(x) &= x_1x_2 + x_1 + R_{10}x_2^2 + x_2x_3^2 + R_7x_2x_3 + R_9x_2x_4 + R_8x_2 + R_5x_3^2 + R_6x_3 + x_4^2 - 1
\end{aligned}
$$

with data $R_5, \ldots, R_{10}$ which depends on the parameters p and R described previously. In this problem $n = 5$, $p = 40$, and $R = 10$. (We note that there is a typographical error in the paper of Meintjes and Morgan [27]; the last term in the equation defining f_4 should be $-4Rx_5$ and not $+4Rx_5$.)

This system of equations has four solutions with real components for $p = 40$ and $R = 10$. There is only one solution with all positive components; this is the desired solution to the physical problem.

This problem is not difficult to solve, but unless bounds are imposed, the physical solution may not be found. An unconstrained algorithm usually finds the physical solution from the standard starting point x_s but tends to converges to non-physical solutions from the starting points $10x_s$ and $100x_s$. The bounds $x_j \geq 0$ can be used to obtain the physical solution.

3 Least Squares Problems

Solutions to a nonlinear least squares problem subject to equality and inequality constraints are local minimizers of the problem

$$\min\{\|f(x)\|_2^2 : c_l \le c(x) \le c_u\},$$

where $f : \mathbb{R}^n \to \mathbb{R}^m$ defines the residuals of the least squares problem, $c : \mathbb{R}^n \to \mathbb{R}^p$ is the constraint function, and c_l and c_u are bounds. Equality constraints are obtained when components of c_l and c_u have the same value. Problems in this section include bound-constrained problems where $c(x) = x$, and equality constrained problems where $c_l = c_u$.

3.1 Isomerization of α-pinene – Direct Formulation

This problem requires the determination of the reaction coefficients in the thermal isomerization of α-pinene. The linear kinetic model proposed for this problem is of the form

$$\begin{aligned} y_1' &= -(\theta_1 + \theta_2) y_1 \\ y_2' &= \theta_1 y_1 \\ y_3' &= \theta_2 y_1 - (\theta_3 + \theta_4) y_3 + \theta_5 y_5 \\ y_4' &= \theta_3 y_3 \\ y_5' &= \theta_4 y_3 - \theta_5 y_5 \end{aligned} \tag{3.1}$$

where $\theta_1, \ldots, \theta_5$ are the unknown coefficients. Initial conditions for the differential equation are known. In this problem the relative concentrations of α-pinene and three by-products are measured at eight time points, while the relative concentration of a fourth by-product is derived from the other concentrations. Thus, at a set of eight time points $\tau_1, \ldots, \tau_8$, vectors of concentration measurements z_j are given for y at τ_j, where y is the solution to the system of differential equations which governs the reaction. The α-pinene problem is to minimize

$$\sum_{j=1}^{8} \|y(\tau_j; \theta) - z_j\|^2, \tag{3.2}$$

where θ is the vector with components $\theta_1, \ldots, \theta_5$ of unknown reaction coefficients. This formulation of the α-pinene problem is based on the work of Box, Hunter, MacGregor, and Erjavac [6].

The α-pinene problem is a typical example of inverse problems involving differential equations that arise in chemical kinetics. In the general case the reaction is governed by a system of p differential equations

$$y'(t) = F(t, y(t), \theta), \qquad a \le t \le b,$$

which depend on a vector $\theta \in \mathbb{R}^q$ of unknown parameters. Initial conditions for $y \in \mathbb{R}^p$ may also be provided, and may also depend on θ. In the α-pinene problem $p = q$ and F is bilinear in θ and y, but these conditions do not hold in general.

We formulate the α-pinene problem as an unconstrained nonlinear least squares problem involving a numerical approximation $u(t;\theta)$ to $y(t;\theta)$ obtained from a fourth-order Runge-Kutta scheme over n_0 time intervals. The optimization problem is then to determine a parameter vector $\theta \in \mathbb{R}^5$ that solves the problem

$$\min \left\{ \sum_{j=1}^{8} \|u(\tau_j;\theta) - z_j\|^2 : \ \theta_i \geq 0, \ i = 1, \ldots, 5 \right\}. \tag{3.3}$$

This is a nonlinear least squares problem with $m = 40$ equations and $n = 5$ variables. The constraints $\theta_i \geq 0$ arise from physical considerations. For sufficiently large n_0, we expect that solutions to problem (3.3) will be close to a solution of problem (3.2).

An approximation to the Jacobian matrix for this formulation of the α-pinene problem can be obtained by solving a system of coupled ordinary differential equations consisting of the original α-pinene equations and 25 additional equations. The additional equations are obtained by differentiating each of the α-pinene equations (3.1) by θ_j for $j = 1, \ldots, 5$, and noting that if

$$w_{i,j}(\tau;\theta) \equiv \partial_{\theta_j} y_i(\tau;\theta),$$

then

$$w'_{i,j}(\tau;\theta) = \partial_{\theta_j} y'_i(\tau;\theta).$$

The approximation to the Jacobian matrix obtained by this method is more accurate than an approximation based on differences of function values.

In our numerical results we used a Runge-Kutta method with $n_0 = 80$ time intervals. The solution of this version of the α-pinene problem is not difficult to obtain from the standard start θ_s, but becomes increasingly difficult to solve from remote starting points. The approximation $u(t, \cdot)$ for the optimal θ, shown in Figure 3.1, is an excellent fit to the data.

Numerical difficulties in solving this version of the α-pinene problem are mainly due to the result that with fixed-steplength Runge-Kutta techniques and sufficiently large values of θ, the approximation $u(t;\theta)$ becomes unbounded as t increases. In contrast, the true solution $y(t;\theta)$ of the differential equations problem remains bounded for $t \geq 0$ for any nonnegative choice of $\theta_1, \ldots, \theta_5$. Hence, our test problem becomes difficult to solve given poor initial estimates for θ. We illustrate this remark by noting that while the initial residual norm computed at the standard starting point was 7.03, the residual computed at 50 times this starting point was 7.22×10^{110}. Hence, this problem is quite challenging with respect to choice of initial estimates.

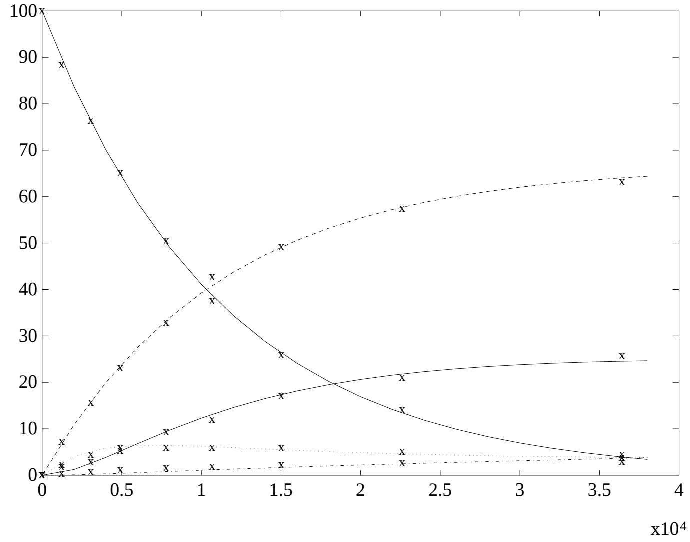

Figure 3.1: The five components of $u(t, \theta)$ for the α-pinene problem with the optimal θ

In general, extreme sensitivity to initial θ estimates is a hallmark of problems of this type. As with the α-pinene example, integration may be done using a fixed-steplength numerical solver of initial value problems. Decreasing the steplength does not affect the size of the optimization problem and may enlarge the set of possible θ for which the solver is stable, but of course adds to the computational expense of residual and Jacobian evaluations. A more interesting approach is to use an adaptive solver, which automatically adjusts steplengths to correspond to the level of accuracy requested by the user, and can be expected to be more stable than fixed steplength solvers. The resultant approximation $u(t; \theta)$ may not be smooth with respect to small variations in θ (Lyness [25] provides an interesting discussion of this point), but this approach allows the possibility of using optimization software that makes use of variable-accuracy evaluations to avoid the expense of full accuracy solutions whenever possible. This challenging category of test problems has been neglected in the literature.

3.2 Isomerization of α-pinene – Collocation Formulation

The second α-pinene problem requires the determination of the reaction coefficients in the thermal isomerization of α-pinene; in this problem, however, collocation is used to approximate the solution of the differential equations that define the kinetics of the problem. This formulation of the α-pinene problem is based on the work of Tjoa and Biegler [38].

The collocation method for initial value problems is a special case of the collocation method for boundary value problems that was used in Sections 2.1 and 2.2. Recall that the

k-stage collocation method is defined in terms of a partition

$$a = t_1 < t_2 < \cdots < t_{n_0} < t_{n_0+1} = b$$

of $[a, b]$, and a set of k collocation points in each interval $[t_i, t_{i+1}]$. The collocation method approximates the solution of the system of differential equations by a vector-valued function $u_\pi : [a, b] \to \mathbb{R}^p$, where each component of u_π is a polynomial of order $k + 1$ in each subinterval $[t_i, t_{i+1}]$. Thus u_π is defined in terms of $n_0 p(k+1)$ parameters. In the collocation formulation these parameters are determined by requiring that $u_\pi \in C[a, b]$ and that u_π satisfy the differential equation at the collocation points. In the usual case we are given p initial values; these initial values together with the continuity and collocation equations lead to a system of $n_0 p(k + 1)$ equations in the $n_0 p(k + 1)$ parameters that define u_π.

We now formulate the α-pinene problem as a minimization problem subject to equality constraints. Let $x \in \mathbb{R}^n$ be the vector that defines u_π, with $n = n_0 p(k + 1)$. We define $u_\pi(\tau; x) \equiv u_\pi(\tau)$ to make explicit the dependence of u_π on x. If we write the initial value, continuity, and collocation equations as constraints of the form

$$c(x, \theta) = 0,$$

where $c : \mathbb{R}^{n+q} \to \mathbb{R}^n$, then the optimization problem is

$$\min \left\{ \sum_{j=1}^{m} \|u_\pi(\tau_j; x) - z_j\|^2 : c(x, \theta) = 0 \right\}.$$

The l_2 penalty approach to the solution of this problem leads to a least squares problem of the form

$$\min \left\{ \sum_{j=1}^{m} \|u_\pi(\tau_j; x) - z_j\|^2 + \tfrac{1}{2} \sum_{i=1}^{n} \sigma_i c_i(x, \theta)^2 \right\},$$

where $\sigma_i > 0$, while the augmented Lagrangian approach leads to a problem of the form

$$\min \left\{ \sum_{j=1}^{m} \|u_\pi(\tau_j; x) - z_j\|^2 + \tfrac{1}{2} \sum_{i=1}^{n} \sigma_i \left[c_i(x, \theta) + \frac{\lambda_i}{\sigma_i} \right]^2 \right\},$$

where $\sigma_i > 0$ and $\lambda_i \in \mathbb{R}$ is an estimate of the Lagrange multiplier for the constraint c_i. Both approaches lead to least squares problem with $mp + n$ equations and $n + q$ variables. Recall that $n = n_0 p(k+1)$ and that n_0 is the number of subintervals, $k+1$ is the order of the polynomials that define u_π in each subinterval, p is the number of differential equations in the model, q is the number of components in the parameter vector θ, and m is the number of data points. Note that n_0 and k can be specified, while the other parameters are dependent on the problem. Arbitrarily large-dimensional test problems can be generated by selecting larger values of n_0.

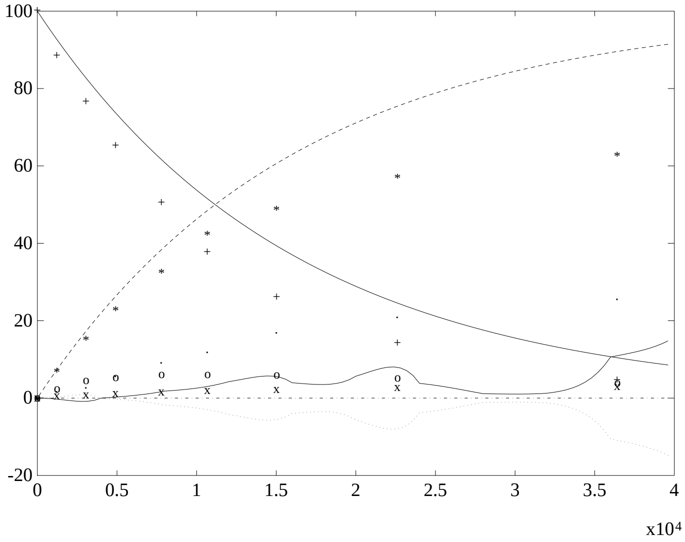

Figure 3.2: Spurious solution $u(t;\theta)$ to the α-pinene problem

For the α-pinene problem, $p = q = 5$, and $m = 8$. We selected $k = 4$ and $n_0 = 10$, so our least squares problem has 290 equations and 255 variables. Convergence from the standard starting point is not difficult. The final approximation obtained for y was identical to that shown in Figure 3.1.

The collocation approach demonstrates the value of using simple constraints on the variables to enforce the known nonnegativity of the components of θ. For sufficiently remote starting points, unconstrained least squares algorithms applied to this problem may converge to a different local minimizer with some components of θ negative, with u as shown in Figure 3.2. A bound version would not have encountered this spurious local solution. Simple bound constraints on the variables defining the approximation u_π can also be used to enforce $u_\pi(t_j) \geq 0$ at the node points t_j — a desirable feature in chemical engineering problems where negative concentrations have no physical meaning. Similarly, if physical considerations dictate that a given component of u_π be increasing or decreasing, simple constraints on the variables can ensure that a given component of $u'_\pi(t_j)$ is of the correct sign. These capabilities for constraining u are not present in the direct approach.

3.3 Coating Thickness Standardization

The coating thickness problem arises from the need to nondestructively determine any nonuniformity in the lead-tin coating on samples of standard reference materials. This is a multiple-response data-fitting problem communicated by Janet Rogers of the National Institute of Standards and Technology.

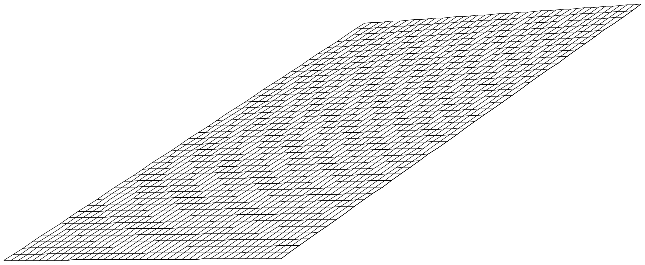

Figure 3.3: Coating thickness standards model $z_1(\xi_1, \xi_2)$ for optimal $x_1, \ldots, x_4$

At each of n_0 isolated points on the surface, we have measurements y_i for the coating thickness, the relative abundance y_{i+n_0} of lead to tin, and the surface coordinates $(\xi_1, \xi_2)_i$ at which the measurements were made. All four of these values are subject to error. We model the thickness of the coating and the relative abundance of tin to lead using simple bilinear tensor-product functions

$$\begin{aligned} z_1(\xi_1, \xi_2) &= x_1 + x_2\xi_1 + x_3\xi_2 + x_4\xi_1\xi_2, \\ z_2(\xi_1, \xi_2) &= x_5 + x_6\xi_1 + x_7\xi_2 + x_8\xi_1\xi_2. \end{aligned}$$

We seek values of the parameters $x_1, \ldots, x_8$, and small perturbations $x_9, \ldots, x_{8+2n_0}$ to the measured coordinates $(\xi_1, \xi_2)_i$ which fit the data in a least squares sense. This formulation leads to a least squares problem with residuals of the form

$$\begin{aligned} f_i(x) &= z_1(\xi_{1,i} + x_{8+i}\,, \xi_{2,i} + x_{8+i+n_0}) - y_i, \qquad 1 \le i \le n_0, \\ f_{i+n_0}(x) &= z_2(\xi_{1,i} + x_{8+i}\,, \xi_{2,i} + x_{8+i+n_0}) - y_{i+n_0}, \qquad 1 \le i \le n_0, \end{aligned}$$

and

$$f_{i+2n_0}(x) = w_i \; x_{8+i}, \qquad 1 \le i \le 2n_0,$$

where y_i and $\xi_{1,i}$, $\xi_{2,i}$ are the measured data and w_i is a set of weights. These residuals define a least squares problem with $n = 8 + 2n_0$ variables and $m = 4n_0$ equations. In the data supplied by Susannah Shiller of the National Institute of Standards and Technology, $n_0 = 63$, so that $n = 134$ and $m = 252$.

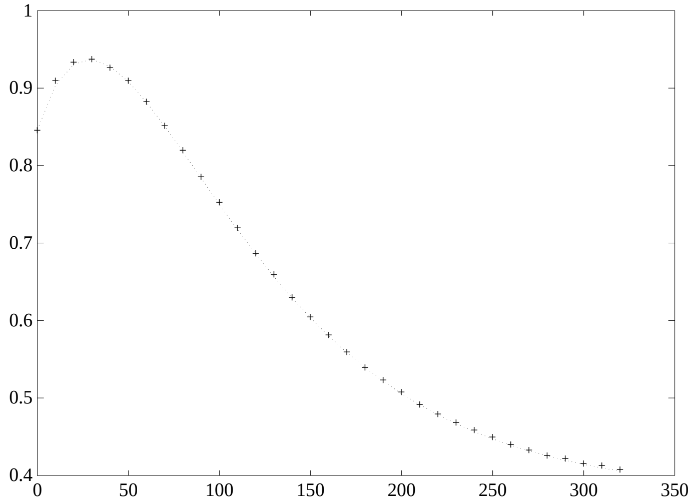

Figure 3.4: Exponential data fitting problem I

This problem is not difficult to solve from any of the starting values. Convergence always takes place to the same minimizer. At the solution x^* we have

$$\|f(x^*)\| = 0.7109842,$$

which reveals an excellent fit of the model to the data, since y_i is of order 10.

A plot of the model z_1 as a function of (ξ_1, ξ_2), with $x_1, \ldots, x_4$ set to the optimal values for the parameters, appears in Figure 3.3. This plot suggests that the model is almost linear in the region of interest. However, the relationship is not *exactly* linear because a calculation shows that in this region the first component of the gradient of the model varies over the interval $[16.4, 18.6]$, while the second component varies over the interval $[1.17, 1.33]$.

We have omitted a plot of the model z_2 with optimal $x_5, \ldots, x_8$ because the plot is similar. Moreover, for this model we also conclude that the relationship between the model and the independent parameters is almost linear in the region of interest.

3.4 Exponential Data Fitting I

This is an exponential data fitting problem using data supplied by A. M. Sargeson from the Research School of Chemistry of the Australian National University. The problem is of the form

$$f_i(x) = y_i - \Big(x_1 + x_2 \exp(-t_i x_4) + x_3 \exp(-t_i x_5)\Big),$$

with $t_i = 10(i-1)$ and data y_i. In this problem $m = 33$ and $n = 5$. The formulation of this problem is due to Osborne [35].

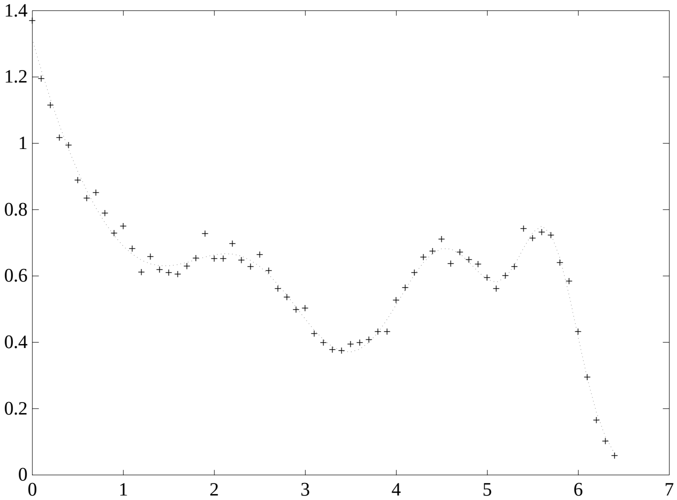

Figure 3.5: Exponential data fitting problem II

This problem is not difficult to solve from the standard starting point x_s and from the starting point $10x_s$, but underflows occur from the starting point $100x_s$. The bounds $-10 \le x_j \le 10$ for $1 \le j \le 5$ can be used to solve this problem. At the solution x^* we have

$$\|f(x^*)\| = 0.7392493 \times 10^{-2}.$$

A plot of the data and the model with the optimal parameters appears in Figure 3.4.

3.5 Exponential Data Fitting II

This is an exponential data fitting problem using data supplied by W. J. Caelli from the Research School of Physical Sciences of the Australian National University. The problem is of the form

$$\begin{aligned} f_i(x) = y_i - \Big(&x_1 \exp(-t_i x_5) + x_2 \exp(-(t_i - x_9)^2 x_6) \\ &+ x_3 \exp(-(t_i - x_{10})^2 x_7) + x_4 \exp(-(t_i - x_{11})^2 x_8)\Big), \end{aligned}$$

with $t_i = (i-1)/10$ and data y_i. In this problem $m = 65$ and $n = 9$. The formulation of this problem is due to Osborne [35].

This problem is not difficult to solve from the standard starting point x_s, but underflows occur from other starting points. The bounds $0 \le x_j \le 10$ for $1 \le j \le 10$ can be used to solve this problem. At the solution x^* we have

$$\|f(x^*)\| = 0.2003440.$$

A plot of the data and the model with the optimal parameters appears in Figure 3.5.

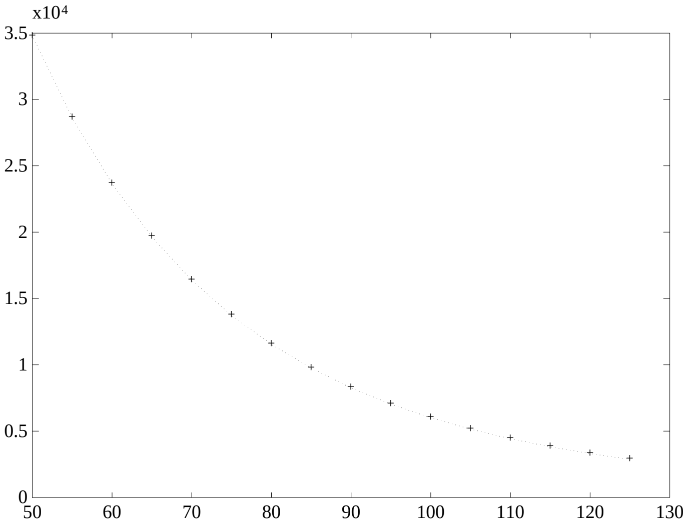

Figure 3.6: The thermistor resistance problem

3.6 Analysis of Thermistor Resistance

The thermister resistance problem arises in the analysis of the resistance of a thermistor as a function of the temperature. The data was supplied by J. H. Hadley of the Shell Development Company. The problem is of the form

$$f_i(x) = x_1 \exp\left(\frac{x_2}{(t_i + x_3)}\right) - y_i,$$

with data y_i at the time points $t_i = 5 + 45i$ for $i = 1, \ldots, m$. In this problem $m = 16$ and $n = 3$. The formulation of this problem is due to Meyer [28].

This problem is difficult to solve even from the standard starting point x_s. At the solution x^* we have

$$\|f(x^*)\| = 9.377945 \ .$$

An algorithm can fail from other starting points because the dependence of this problem on t_i is lost for large values of x_3. In particular, if x_3 is sufficiently large, this problem reduces to a one-variable problem in the variable

$$z = x_1 \exp\left(\frac{x_2}{x_3}\right) .$$

This situation can be avoided by imposing the bound $x_3 \leq 10^3$. This bound is fairly tight since at the solution, $x_3 = 345.22$.

A plot of the data and the model with the optimal parameters appears in Figure 3.6.

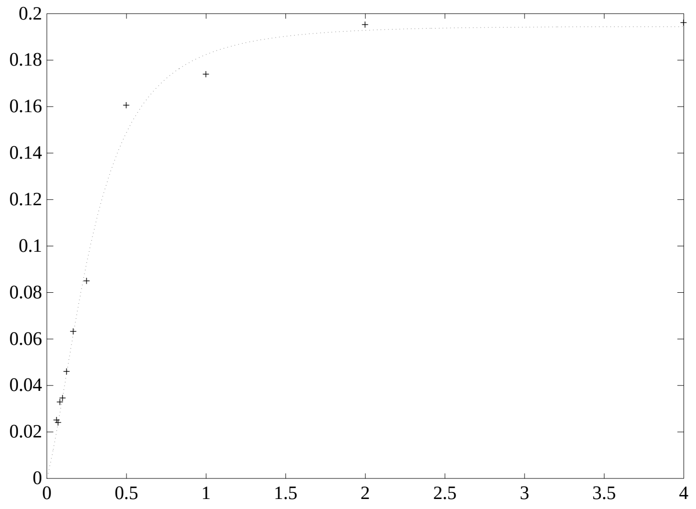

Figure 3.7: The enzyme problem

3.7 Analysis of an Enzyme Reaction

This problem arises in the analysis of the kinetic data for an enzyme reaction. The problem is of the form

$$f_i(x) = y_i - \frac{x_1(u_i^2 + u_i x_2)}{u_i^2 + u_i x_3 + x_4},$$

with data y_i and u_i. In this problem $m = 11$ and $n = 4$. This problem was formulated by Kowalik and Morrison [23].

This problem can be solved from the standard starting point x_s without difficulty. At the solution x^* we have

$$\|f(x^*)\| = 0.1753584 \times 10^{-1}.$$

From the starting point $10x_s$, algorithms may be attracted to a local minimizer at infinity with $x_2 = -14.075$. For this minimizer

$$\|f(x^*)\| = 0.3205219 \times 10^{-1}.$$

There are no bounds in this problem, but imposing the bounds $x_3 \geq 0$ and $x_4 \geq 0$ guarantees that the function is well defined, since $u_i \geq 0$ for all i. The local minimizer at infinity can be avoided by imposing the bounds $x_j \leq 100$ for $1 \leq j \leq 4$.

A plot of the data and the model with the optimal parameters appears in Figure 3.7.

3.8 Chebyshev Quadrature

The Chebyshev problem arises from the determination of the nodes of a quadrature formula with equal weights. The problem is of the form

$$f_i(x) = \frac{1}{n}\sum_{j=1}^{n} T_i(x_j) - \int_0^1 T_i(\xi)\, d\xi,$$

where T_i is the i-th Chebyshev polynomial shifted to the interval $[0,1]$. In this problem any $m \geq n$ is allowed, but in the discussion below it is assumed that $m = n$. This problem was formulated by Fletcher [14].

The Chebyshev problem has a zero residual solution for $1 \leq n \leq 7$ and for $n = 9$. Note that the solution is not unique because any permutation of the variables also yields a solution. Thus, there are $n!$ distinct zero residual solutions. Such a zero residual solution can be obtained without difficulty from the standard starting point, but the problem becomes difficult to solve from the starting point $10x_s$ unless bounds are imposed. Since the nodes are required to be in the interval $[0,1]$, the bounds $0 \leq x_j \leq 1$ for $1 \leq j \leq n$ are natural.

There seems to be a unique minimum of the least squares problem for $n = 8$ and $n = 11$. They are given by

$$\|f(x^*)\| = 0.5930324 \times 10^{-1}, \qquad n = 8,$$
$$\|f(x^*)\| = 0.5291277 \times 10^{-1}, \qquad n = 11.$$

We have found two local minima for $n = 10$. They are given by

$$\|f(x^*)\| = 0.6908483 \times 10^{-1}, \qquad \|f(x^*)\| = 0.8064710 \times 10^{-1}.$$

If an algorithm is started from the standard starting point, it will usually converge to the second minimum given above. There are multiple local minima for $n > 11$.

4 Minimization Problems

The problem of minimizing a function $f : \mathbb{R}^n \to \mathbb{R}$ subject to equality and inequality constraints can be expressed in the form

$$\min\{f(x) : c_l \le c(x) \le c_u\},$$

where $c : \mathbb{R}^n \to \mathbb{R}^p$ is the constraint function, and c_l and c_u are bounds. Equality constraints are obtained when components of c_l and c_u have the same value. Many of the problems in this section are bound-constrained problems; in this case $c(x) = x$.

4.1 Elastic-Plastic Torsion

The elastic plastic torsion problem arises from the determination of the stress field on an infinitely long cylindrical bar. The infinite-dimensional version of this problem is of the form

$$\min\{q(v) : v \in K\},$$

where $q : K \to \mathbb{R}$ is the quadratic

$$q(v) = \int_{\mathcal{D}} \left\{ \tfrac{1}{2} \|\nabla v(x)\|^2 - cv(x) \right\} dx$$

for some constant c, and $\mathcal{D}$ is a bounded domain with smooth boundary. The convex set K is defined by

$$K = \{v \in H_0^1(\mathcal{D}) : |v(x)| \le dist(x, \partial\mathcal{D}),\ x \in \mathcal{D}\},$$

where $dist(\cdot, \partial\mathcal{D})$ is the distance function to the boundary of $\mathcal{D}$, and $H_0^1(\mathcal{D})$ is the Hilbert space of all functions with compact support in $\mathcal{D}$ such that v and $\|\nabla v\|^2$ belong to $L^2(\mathcal{D})$.

This formulation and the physical interpretation of the torsion problem are discussed, for example, in Glowinski [17, pp. 41–55]. A plot of the solution to the torsion problem with $\mathcal{D} = (0,1) \times (0,1)$ and $c = 25$ appears in Figure 4.1.

A finite element approximation to the torsion problem is obtained by triangulating $\mathcal{D}$ and replacing the minimization of q over $H_0^1(\mathcal{D})$ by the minimization of q over the set of piecewise linear functions that satisfy the constraints specified by K. The finite element approximation thus gives rise to a finite-dimensional minimization problem whose variables are the values of the piecewise linear function at the vertices of the triangulation.

We develop a finite element approximation to a minimization problem with a quadratic q of the general form

$$q(v) = \int_{\mathcal{D}} \left\{ \tfrac{1}{2} w_q(x) \|\nabla v(x)\|^2 - w_l(x) v(x) \right\} dx, \tag{4.1}$$

where $w_q : \mathcal{D} \to \mathbb{R}$ and $w_l : \mathcal{D} \to \mathbb{R}$ are functions defined on the rectangle $\mathcal{D}$. In the torsion problem $w_q \equiv 1$ and $w_l \equiv c$.

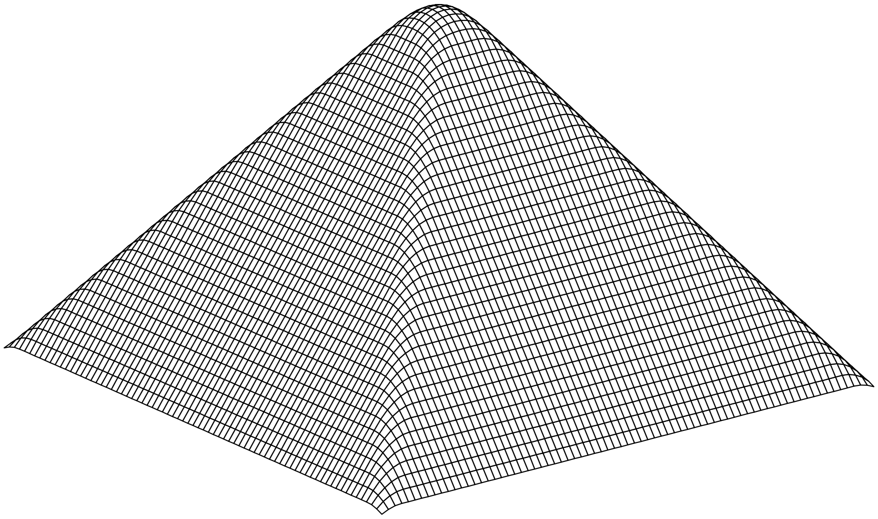

Figure 4.1: Torsion problem with $c = 25$

Let $\mathcal{D} = (l_1, u_1) \times (l_2, u_2)$ be a rectangle in $\mathbb{R}^2$. Vertices $z_{i,j} \in \mathcal{D}$ for a triangulation of $\mathcal{D}$ are obtained by choosing grid spacings h_x and h_y and defining grid points

$$z_{i,j} = (l_1 + ih_x, l_2 + jh_y), \qquad 0 \le i \le n_x + 1, \quad 0 \le j \le n_y + 1,$$

such that $z_{n_x+1,n_y+1} = (u_1, u_2)$. The triangulation consists of triangular elements T_L with vertices at $z_{i,j}, z_{i+1,j}$, and $z_{i,j+1}$ and elements T_U with vertices at $z_{i,j}, z_{i-1,j}$, and $z_{i,j-1}$.

A finite element approximation to the torsion problem is obtained by minimizing q over the space of piecewise linear functions v with values $v_{i,j}$ at $z_{i,j}$. The approximation to the integral

$$\int_{\mathcal{D}} w_q(x) \|\nabla v(x)\|^2 \, dx$$

over the element T_L is the quadratic $q^L_{i,j}$, where

$$q^L_{i,j}(v) = \mu^L_{i,j} \left\{ \left(\frac{v_{i+1,j} - v_{i,j}}{h_x} \right)^2 + \left(\frac{v_{i,j+1} - v_{i,j}}{h_y} \right)^2 \right\},$$

$$\mu^L_{i,j} = \frac{h_x h_y}{6} \left\{ w_q(z_{i,j}) + w_q(z_{i+1,j}) + w_q(z_{i,j+1}) \right\}.$$

Similarly, the approximation over the element T_U is the quadratic $q^U_{i,j}$, where

$$q^U_{i,j}(v) = \mu^U_{i,j} \left\{ \left(\frac{v_{i-1,j} - v_{i,j}}{h_x} \right)^2 + \left(\frac{v_{i,j-1} - v_{i,j}}{h_y} \right)^2 \right\},$$

$$\mu^U_{i,j} = \frac{h_x h_y}{6}\left\{ w_q(z_{i,j}) + w_q(z_{i-1,j}) + w_q(z_{i,j-1}) \right\}.$$

These calculations show that the finite element approximation to the quadratic (4.1) leads to a quadratic programming problem of the form

$$\min\{q(v) : v \in \Omega\}, \tag{4.2}$$

where q is the quadratic

$$q(v) = \frac{1}{2}\sum \left(q^L_{i,j}(v) + q^U_{i,j}(v) \right) - h_x h_y \sum w_l(z_{i,j}) v_{i,j}. \tag{4.3}$$

Note that in this formulation the quadratic $q^L_{i,j}$ is defined only when $0 \le i \le n_x$ and $0 \le j \le n_y$, while $q^U_{i,j}$ is defined when $1 \le i \le n_x + 1$ and $1 \le j \le n_y + 1$. Also note that for the torsion problem $w_q \equiv 1$ and $w_l \equiv c$ and that the feasible set Ω is

$$\Omega = \{ v \in \mathbb{R}^{n_x n_y} : |v_{i,j}| \le d_{i,j} \},$$

where $d_{i,j}$ is the value of $dist(\cdot, \partial\mathcal{D})$ at $z_{i,j}$.

In general, the problem becomes easier to solve as c increases because then the linear term in q dominates. Numerical results for the elastic-plastic torsion problem are presented, for example, by O'Leary and Yang [34], Elliott and Ockendon [13, pp. 124–125], and Moré and Toraldo [29].

4.2 Pressure Distribution in a Journal Bearing

The journal bearing problem arises in the determination of the pressure distribution in a thin film of lubricant between two circular cylinders. The infinite-dimensional version of this problem is of the form

$$\min\{q(v) : v \in K\},$$

where $q : K \to \mathbb{R}$ is the quadratic (4.1) with

$$w_q(\xi_1, \xi_2) = (1 + \epsilon \cos \xi_1)^3, \quad w_l(\xi_1, \xi_2) = \epsilon \sin \xi_1$$

for some constant ϵ in $(0, 1)$, and $\mathcal{D} = (0, 2\pi) \times (0, 2b)$ for some constant $b > 0$. The convex set K is defined by

$$K = \{ v \in H^1_0(\mathcal{D}) : v \ge 0 \text{ on } \mathcal{D} \}.$$

In the formulation of Cimatti [9], all functions in K were required to be periodic in the first argument with period 2π; in our formulation we have neglected the periodicity conditions.

A finite element approximation to the journal bearing problem is obtained as in the torsion problem. The result is a quadratic programming problem of the form (4.2), where q

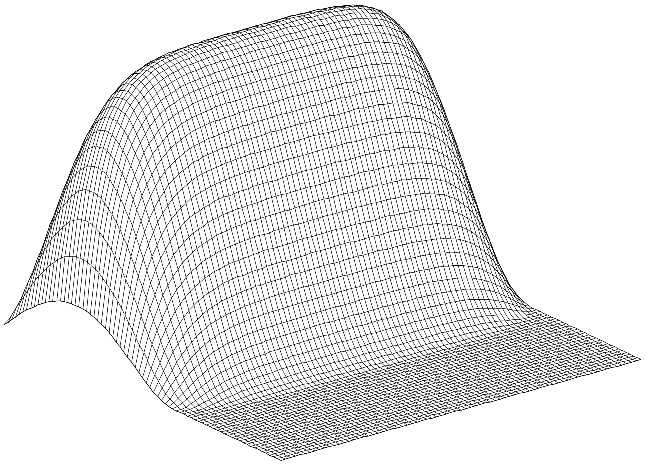

Figure 4.2: Journal bearing problem with $b = 10$ and $\epsilon = 0.1$

is the quadratic defined by (4.3). For the journal bearing problem $w_q(\xi_1, \xi_2) = (1 + \epsilon \cos \xi_1)^3$ and $w_l(\xi_1, \xi_2) = \epsilon \sin \xi_1$, and the feasible set Ω is

$$\Omega = \{ v \in \mathbb{R}^{n_x n_y} : v_{i,j} \geq 0 \}.$$

A plot of the solution to the finite-dimensional approximation to the journal bearing problem with $b = 10$ and $\epsilon = 0.1$ appears in Figure 4.2. This problem is harder to solve than the elastic-plastic torsion problem unless the problem is scaled so that the diagonal elements in the matrix that represents q are unity. Numerical results for the journal bearing problem are presented, for example, by Lin and Cryer [24], Cimatti and Menchi [10], and Moré and Toraldo [29].

4.3 Minimal Surfaces

The determination of the surface with minimal area and given boundary values in a convex domain $\mathcal{D}$ is an infinite-dimensional optimization problem of the form

$$\min\{ f(v) : v \in K \},$$

where $f : K \to \mathbb{R}$ is the functional

$$f(v) = \int_{\mathcal{D}} \left(1 + \|\nabla v(x)\|^2 \right)^{1/2} dx,$$

and the set K is defined by

$$K = \left\{ v \in H^1(\mathcal{D}) : v(x) = v_D(x) \text{ for } x \in \partial\mathcal{D} \right\}$$

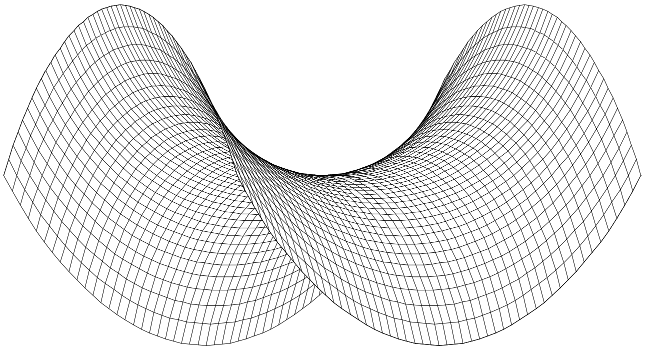

Figure 4.3: Enneper's minimal surface

for some boundary data function $v_D : \partial\mathcal{D} \to \mathbb{R}$. The boundary function v_D uniquely defines the solution to the minimal surface problem.

An interesting minimal surface discovered by A. Enneper is obtained by defining v_D on $\mathcal{D} = (-\frac{1}{2}, \frac{1}{2}) \times (-\frac{1}{2}, \frac{1}{2})$ by

$$v_D(\xi_1, \xi_2) = u^2 - v^2,$$

where u and v are the unique solutions to the equations

$$\xi_1 = u + uv^2 - \frac{1}{3}u^3, \quad \xi_2 = -v - u^2 v + \frac{1}{3}v^3.$$

For more information on this minimal surface, see Nitsche [32, pp. 80–85]. A plot of this minimal surface appears in Figure 4.3.

A finite element approximation to the minimal surface problem is obtained by minimizing f over the space of piecewise linear functions v with values $v_{i,j}$ at $z_{i,j}$, where $z_{i,j} \in \mathbb{R}^2$ are the vertices of a triangulation of $\mathcal{D}$ with grid spacings h_x and h_y. The values $v_{i,j}$ are obtained by solving the minimization problem

$$\min\{\sum \left(f^L_{i,j}(v) + f^U_{i,j}(v)\right) : v \in \mathbb{R}^n\},$$

where the functions $f^L_{i,j}$ and $f^U_{i,j}$ are defined by

$$f^L_{i,j}(v) = \frac{h_x h_y}{2}\left\{1 + \left(\frac{v_{i+1,j} - v_{i,j}}{h_x}\right)^2 + \left(\frac{v_{i,j+1} - v_{i,j}}{h_y}\right)^2\right\}^{1/2}$$

$$f_{i,j}^U(v) = \frac{h_x h_y}{2} \left\{ 1 + \left(\frac{v_{i-1,j} - v_{i,j}}{h_x} \right)^2 + \left(\frac{v_{i,j-1} - v_{i,j}}{h_y} \right)^2 \right\}^{1/2} .$$

Note that in this formulation $f_{i,j}^L$ is defined only when $0 \le i \le n_x$ and $0 \le j \le n_y$, while $f_{i,j}^U$ is defined when $1 \le i \le n_x + 1$ and $1 \le j \le n_y + 1$.

4.4 Optimal Design with Composite Materials

This optimal design problem requires determining the placement of two elastic materials in the cross-section of a rod with maximal torsional rigidity. Our formulation follows the approach of Goodman, Kohn, and Reyna [19].

Let $\mathcal{D}$ in $\mathbb{R}^2$ be a bounded domain, and let $w < |\mathcal{D}|$, where $|\mathcal{D}|$ denotes the area of $\mathcal{D}$. The solution of the optimal design problem is a subset Ω of $\mathcal{D}$ that solves the problem

$$\min\{\mathcal{F}(v,\Omega) : v \in H_0^1(\mathcal{D}),\ |\Omega| = w\}$$

where

$$\mathcal{F}(v,\Omega) = \int_{\mathcal{D}} \left\{ \tfrac{1}{2}\mu(x)\|\nabla v(x)\|^2 + v(x) \right\} dx,$$

and

$$\mu(x) = \mu_1,\ x \in \Omega, \quad \mu(x) = \mu_2,\ x \notin \Omega.$$

The reciprocals of the constants μ_1 and μ_2 are the shear moduli of the elastic materials in the rod. We assume that $\mu_1 < \mu_2$.

Goodman, Kohn, and Reyna [19] formulate the optimal design problem in terms of a family of problems of the form

$$\min\{f_\lambda(v) : v \in H_0^1(\mathcal{D})\},$$

where $f_\lambda : H_0^1(\mathcal{D}) \to \mathbb{R}$ is the functional

$$f_\lambda(v) = \int_{\mathcal{D}} \left\{ \psi_\lambda\left(\|\nabla v(x)\|\right) + v(x) \right\} dx$$

and $\psi_\lambda : \mathbb{R} \to \mathbb{R}$ is a piecewise quadratic. In this formulation λ is a Lagrange multiplier associated with the optimal design problem, and the piecewise quadratic $\psi_\lambda : \mathbb{R} \to \mathbb{R}$ is of the form

$$\psi_\lambda(t) = \begin{cases} \frac{1}{2}\mu_2 t^2, & 0 \le t \le t_1, \\ \mu_2 t_1 (t - \frac{1}{2}t_1), & t_1 \le t \le t_2, \\ \frac{1}{2}\mu_1 (t^2 - t_2^2) + \mu_2 t_1 (t_2 - \frac{1}{2}t_1), & t_2 \le t, \end{cases}$$

with the breakpoints t_1 and t_2 defined by

$$t_1 = \left(2\lambda \frac{\mu_1}{\mu_2} \right)^{\frac{1}{2}}, \qquad t_2 = \left(2\lambda \frac{\mu_2}{\mu_1} \right)^{\frac{1}{2}}.$$

The definition of the breakpoints implies that $\mu_1 t_2 = \mu_2 t_1$, and thus ψ is continuously differentiable. The solution of the optimum design requires determining a λ that maximizes the mapping $\Phi : \mathbb{R} \rightarrow \mathbb{R}$ defined by

$$\Phi(\lambda) = \phi(\lambda) + [(\mu_1 - \mu_2)w]\lambda,$$

where $\phi(\lambda)$ is the minimum value of f_λ. Goodman, Kohn, and Reyna [19] describe how the solution of the problem defined by Φ can be used to generate a minimizing sequence for the optimal design problem.

In the sequel we consider only the problem of minimizing f_λ for a fixed value of λ. In our numerical results we used $\mu_1 = 1$ and $\mu_2 = 2$, so that $t_1^2 = \lambda$ and $t_2^2 = 2\lambda$. A plot of the norm $\|\nabla v\|$ of the gradient of the stress field v with $\mathcal{D} = (0,1) \times (0,1)$ and $\lambda = 0.008$ appears in Figure 4.4. Figure 4.5 is the contour plot for this surface. In both figures we have used $n_x = n_y = 100$ so that $n = 10^4$.

A finite element approximation to this problem is obtained by minimizing f_λ over the space of piecewise linear functions v with values $v_{i,j}$ at $z_{i,j}$, where $z_{i,j} \in \mathbb{R}^2$ are the vertices of a triangulation of $\mathcal{D}$ with grid spacings h_x and h_y. The values $v_{i,j}$ are obtained by solving the minimization problem

$$\min\{\sum \left(f_{i,j}^L(v) + f_{i,j}^U(v) + h_x h_y v_{i,j} \right) : v \in \mathbb{R}^n\},$$

where the functions $f_{i,j}^L$ and $f_{i,j}^U$ are defined by

$$f_{i,j}^L(v) = \frac{h_x h_y}{2} \psi_\lambda \left(d_{i,j}^+(v) \right), \quad f_{i,j}^U(v) = \frac{h_x h_y}{2} \psi_\lambda \left(d_{i,j}^-(v) \right)$$

with

$$d_{i,j}^\pm(v) = \left\{ \left(\frac{v_{i\pm1,j} - v_{i,j}}{h_x} \right)^2 + \left(\frac{v_{i,j\pm1} - v_{i,j}}{h_y} \right)^2 \right\}^{1/2}$$

Note that in this formulation $f_{i,j}^L$ is defined only when $0 \le i \le n_x$ and $0 \le j \le n_y$, while $f_{i,j}^U$ is defined when $1 \le i \le n_x + 1$ and $1 \le j \le n_y + 1$.

Figures 4.4 and 4.5 show that $\|\nabla v\|$ is changing slowly in the center of $\mathcal{D}$, where $\|\nabla v(x)\| \le t_1$. On the other hand, the gradient changes quite rapidly in the region where $t_1 \le \|\nabla v(x)\| \le t_2$. We are not even guaranteed a continuous gradient ∇v in this region. Thus, approximation by piecewise linear elements seems to be fully justified in this problem.

The rod has the material with greater shear modulus $1/\mu_1$ where the shear $\|\nabla v(x)\| > t_2$, and the weaker material where $\|\nabla v(x)\| < t_1$. Figure 4.4 shows that in the optimal design, the weaker material is placed in the center and corners of the rod.

The region where $t_1 \le \|\nabla v(x)\| \le t_2$ is the homogenized region. The geometry and placement of the region of homogenization are of interest in the optimal design. It is

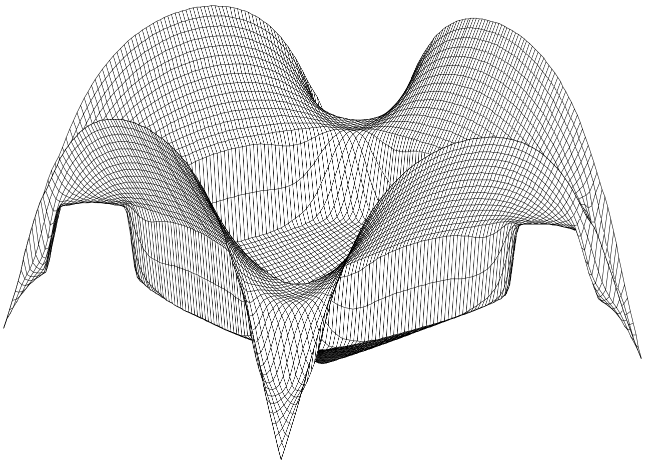

Figure 4.4: Norm $\|\nabla v\|$ for the stress field v in a design with composite materials

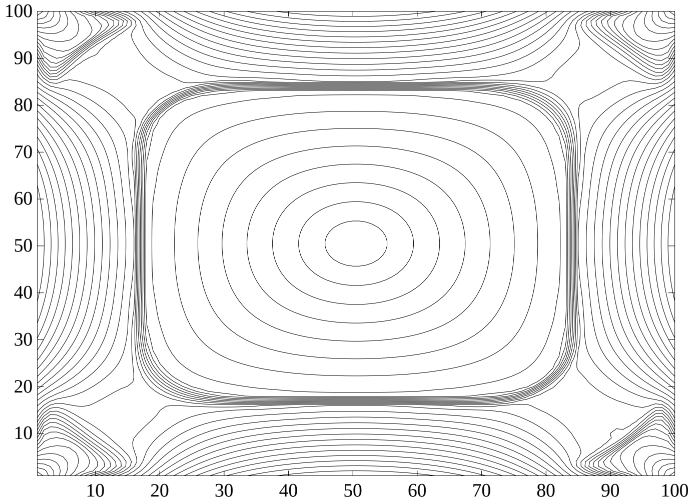

Figure 4.5: Contours of $\|\nabla v\|$ for the stress field v in a design with composite materials

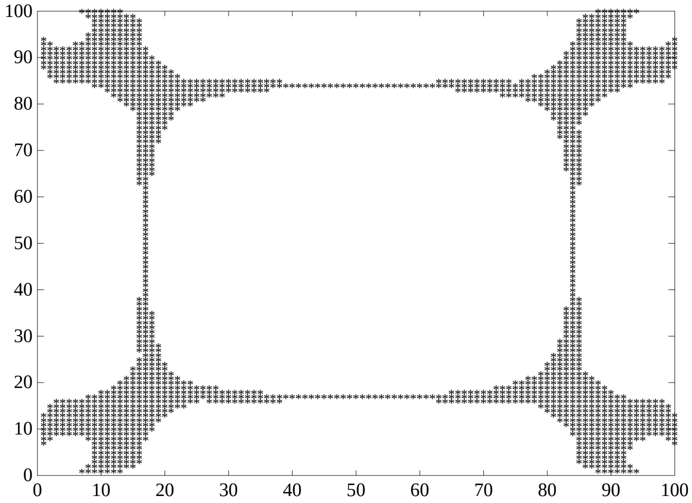

Figure 4.6: Region of homogenization in a design with composite materials ($n = 10^4$)

known, for example, that in general the boundary of this region is not smooth. The plot of this region in Figure 4.6 indicates the unusual nature of this region.

The contour plot in Figure 4.5 and the plots of the region of homogenization in Figure 4.6 are similar to those obtained by Goodman, Kohn, and Reyna [19]. Note, however, that in these plots we used $\lambda = 0.008$, while Goodman, Kohn, and Reyna [19] used $\lambda = 0.002$. Another difference is that the homogenized region in these plots is connected, while this is not the case in the results of Goodman, Kohn, and Reyna [19].

4.5 Inhomogeneous Superconductors: 1-D Ginzburg-Landau

This problem arises in the solution of the Ginzburg-Landau equations for inhomogeneous superconductors in the absence of a magnetic field. The one-dimensional system under consideration consists of alternating layers of lead and tin. Our formulation is based on the work of Garner and Benedek [15].

The optimization problem is to minimize the Gibbs free energy as a function of the temperature. The infinite-dimensional version of this problem is of the form

$$\min\{f(v) : v(-d) = v(d),\ v \in C^1[-d,d]\},$$

where $2d$ is the width of the material, and f is the Gibbs free energy function. In this problem

$$f(v) = \frac{1}{2d}\int_{-d}^{d} \left\{ \alpha(\xi)|v(\xi)|^2 + \tfrac{1}{2}\beta(\xi)|v(\xi)|^4 + \frac{\hbar^2}{4m}|v'(\xi)|^2 \right\} \, d\xi,$$

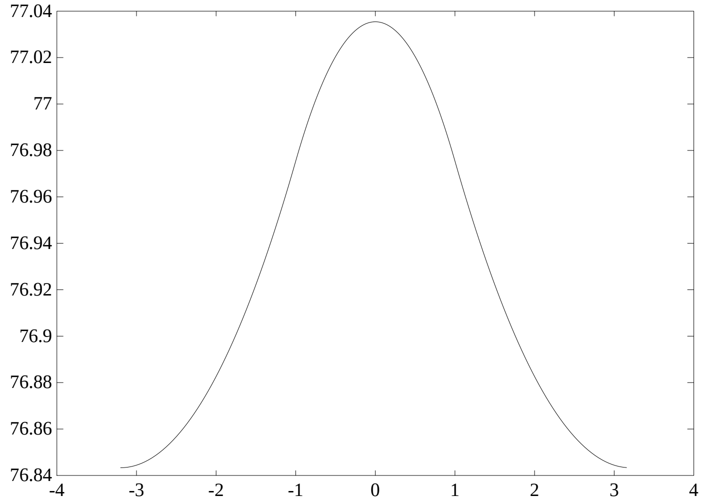

Figure 4.7: Solution to the superconductivity problem with $t = 5$

the functions α and β are piecewise constant for a fixed value of the temperature, $\hbar$ is Planck's constant, and m is the mass of the electron.

The functions α and β are constant in the intervals that correspond to the lead and the tin. Since in this problem the lead in the material corresponds to the interval $[-d_s, d_s]$ and tin in the remaining part of the interval $[-d, d]$, the function α is defined by

$$\alpha(\xi) = \begin{cases} \alpha_N, & -d \leq \xi \leq -d_S \\ \alpha_S, & -d_S < \xi \leq d_S \\ \alpha_N, & d_S < \xi \leq d \end{cases} .$$

Similarly, the function β is defined by

$$\beta(\xi) = \begin{cases} \beta_N, & -d \leq \xi \leq -d_S \\ \beta_S, & -d_S < \xi \leq d_S \\ \beta_N, & d_S < \xi \leq d \end{cases} .$$

The constants α_S and α_N are negative, but β_S and β_N are positive.

As noted above the functions α and β are piecewise constant for a fixed value of the temperature, Thus, the solution to the one-dimensional Ginzbug-Landau problem depends on the temperature t. Values of interest are $t \in [3.73, 7.32]$; a typical value is $t = 5$. A plot of the solution to the superconductivity problem with $t = 5$ is shown in Figure 4.7.

A finite element approximation to the superconductivity problem is obtained by minimizing f over the space of piecewise linear functions v with values v_i at t_i, where

$$-d = t_1 < t_2 < \ldots < t_n < t_{n+1} = d.$$

We assume that there are indices n_1 and n_2 such that $t_{n_1} = -d_S$ and $t_{n_2} = d_S$, where $1 < n_1 < n_2 < n$. This guarantees that the t_i do not straddle a point of discontinuity of the functions α and β. The values v_i are obtained by solving the minimization problem

$$\min\{\frac{1}{2d}\sum_{i=1}^{n} f_i(v) : v \in \mathbb{R}^n\},$$

where

$$f_i(v) = h_i \left\{ \frac{\alpha_i}{3}\frac{v_{i+1}^3 - v_i^3}{v_{i+1} - v_i} + \frac{\beta_i}{10}\frac{v_{i+1}^5 - v_i^5}{v_{i+1} - v_i} + \frac{\hbar^2}{4m}\left(\frac{v_{i+1} - v_i}{h_i}\right)^2 \right\},$$

with $h_i = t_{i+1} - t_i$ the length of the i-th interval, and the constants α_i and β_i the values of the functions α and β in the interval $[t_i, t_{i+1}]$. The constraint that $v(-d) = v(d)$ is taken into account by requiring that $v_{n+1} = v_1$.

4.6 Lennard-Jones Clusters

The determination of the minimum energy configuration of a cluster of atoms or molecules is known as the molecular conformation problem. This is a central problem in the study of cluster statics. For additional background on this problem, see Hoare [20].

Given the positions $p_1, p_2, \ldots, p_n$ of n molecules (points) in $\mathbb{R}^d$, the energy potential function is defined as

$$V_d(p) = \sum_{j=2}^{n}\sum_{i=1}^{j-1} v(||p_j - p_i||_2),$$

where $v : \mathbb{R} \to \mathbb{R}$ is the potential function between pairs of atoms. We consider the Lennard-Jones potential function defined by

$$v(r) = r^{-12} - 2r^{-6}.$$

The molecular conformation problem is to determine a configuration (positions for the n points) such that the energy function V_d is minimized.

In the sequel we consider both the 2-dimensional ($d = 2$) and the 3-dimensional ($d = 3$) case of the total energy potential function. In both cases we determine configurations that are minimal in the sense that they correspond to a local minimizer of the molecular conformation problem with the least known value of the energy function; we do not claim that these minimizers produce the global minimum.

A minimal configuration for the 2-dimensional case with $n = 1000$ is shown in Figure 4.8, while Figure 4.9 is a minimal configuration for the 3-dimensional case with $n = 400$ atoms. The configuration in Figure 4.8 clearly shows that minimal onfigurations are obtained by packing the atoms into a structured tight configuration. The same remark applies to the configuration in Figure 4.9, but now the structure is not clear; this is an advantage of the 2-dimensional case. Note that the bonding that is shown in Figure 4.9 is created by the program used to produce this figure; the molecular conformation problem does not specify any bonding.

The potential function V_d is invariant with respect to permutations, translations, and rotations of the n molecules. Invariance with respect to translations can be eliminated, for example, by translating the cluster so that the center of gravity is at the origin. Also note that the Lennard-Jones potential function $v(\cdot)$ is a non-convex function that is bounded below. Indeed, $v(r)$ approaches $+\infty$ as r approaches 0, while $v(r)$ converges to zero as r approaches $+\infty$. The global minimum of v occurs at $r = 1$.

The minimization of V_d is an unconstrained optimization problem in dn variables. Finding the global minimum

$$E_d = \min \left\{ V_d(p) : p \in \mathbb{R}^{dn} \right\}$$

of V_d is a difficult problem because the number of local minima grows exponentially with the number of atoms n. Algorithms that use random multiple starts or simulated annealing techniques often fail to find the best known solution.

Two-Dimensional Lennard-Jones Clusters

Table 4.1 lists the best known minima of the potential function V_d for the 2-dimensional $(d = 2)$ case. We have only listed values for $n \geq 100$ because of our interest in large problems, but other values are of interest. Details on the algorithm used to generate these results will be published at a later date.

The computational results in Table 4.1 show that the relationship between the best known minima of V_2 and n is nearly linear. A plot of the data clearly shows this linear behavior. Also note that the linear function

$$v_2(x) = \alpha_1 x - \alpha_2, \quad \alpha_1 = 3.30, \quad \alpha_2 = 41.48,$$

fits the data with less than a 2% error; the largest relative error between the values predicted by v_2 and the data in Table 4.1 is 0.017.

There is partial theoretical justification for the conjecture that the relationship between the global minimum of V_2 and n is nearly linear because Xue, Maier, and Rosen [41] have shown that the minimum energy function value has a linear lower bound of the form $-\alpha n$ for some positive constant α if the minimum pair distance is bounded away from zero. If the assumption that the minimum pair distance is bounded away from zero is dropped then the lower bound is of the form $-\alpha n^p$ for some positive constant α and $p = 17/13$.

The minimal configuration for $n = 1000$ is shown in Figure 4.8. Note the regular arrangement of the minimal configuration. This arrangement is fairly typical of minimal

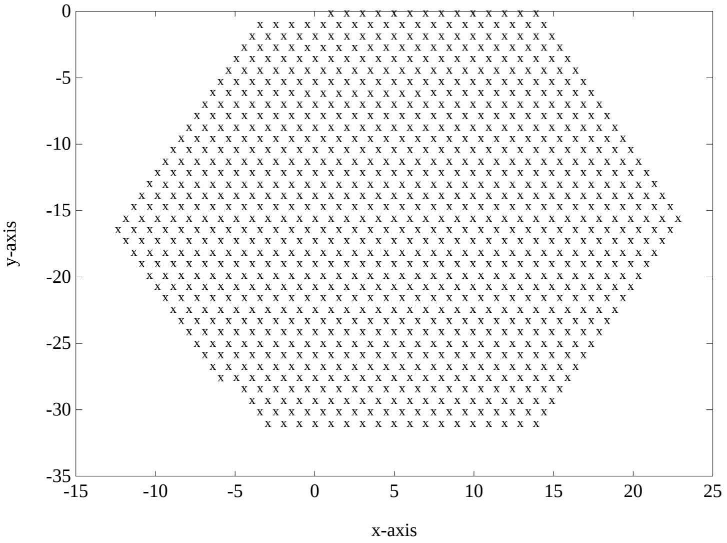

Figure 4.8: Minimal configuration for a 1000-cluster

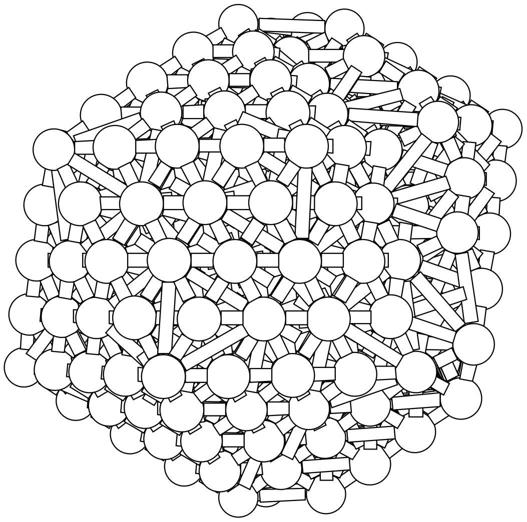

Figure 4.9: Minimal configuration for a 400-cluster

n	$-\min\{V_2\}$	n	$-\min\{V_2\}$
100	293.697	1000	3240.966
200	613.700	2000	6565.062
300	937.704	3000	9902.100
400	1263.247	4000	13245.562
500	1591.147	5000	16595.435
600	1920.311	6000	19946.654
700	2249.524	7000	23301.707
800	2579.787	8000	26658.053
900	2910.363	9000	30015.692
1000	3240.966	10000	33374.616

Table 4.1: Minima for the potential function V_d when $d = 2$

configurations. Figure 4.8 suggests that in a minimal configuration the atoms are arranged by rows where the rows in the middle have k of atoms and the extreme rows have $(k+1)/2$ atoms. A configuration with k atoms in the middle row and $(k+1)/2$ atoms in the extreme rows is possible if k is odd and the number of atoms is $(3k^2+1)/4$. In this configuration the number of atoms in the middle row is $((4n-1)/3)^{1/2}$.

The above argument suggests that in a minimal configuration the number of atoms in the middle rows should be approximately $((4n-1)/3)^{1/2}$. Note, in particular, that if $n = 10^3$, the number of atoms in the middle row should be 36.5. A careful inspection of Figure 4.8 shows that the rows in the middle have 36 atoms.

Three-Dimensional Lennard-Jones Clusters

Northby [33] has made an extensive study of the 3-dimensional molecular conformation problem. Northby used an isocahedral configuration search to find a lattice configuration with lowest possible function value, followed by a local minimization procedure from that configuration. With this approach Northby [33] was able to compute minimal configurations for $n \leq 150$; a table with the results can be found in his paper.

Table 4.2 lists the best known minima of the potential function V_d for the 3-dimensional $(d = 3)$ case. The algorithm used to generate these results relies on some of the techniques used by Northby; the details of this algorithm will be published at a later date.

The computational results in Table 4.2 show that the relationship between the best

n	100	200	300	400	500	600
$-\min\{V_3\}$	557.040	1229.185	1942.107	2650.432	3382.693	4119.244

Table 4.2: Minima for the potential function V_d when $d = 3$

know minima of V_3 and n is nearly linear. In this case the linear function

$$v_3(x) = \alpha_1 x - \alpha_2, \quad \alpha_1 = 7.01, \quad \alpha_2 = 149.12,$$

fits the data with less than a 2% error; the largest relative error between the values predicted by v_3 and the data is Table 4.2 is 0.019. Note that linear function v_3 can be used to predict the best known minimum for $100 \leq n \leq 600$. For example, the minimal configuration for $n = 309$ is -2007.21, while the global minimum predicted by v_3 is -2016.97.

We also note that the computational results in Table 4.2 conform to the result of Xue, Maier, and Rosen [41] that the minimum energy function value has a linear lower bound of the form $-\alpha n$ for some positive constant α if the minimum pair distance is bounded away from zero. The assumption that the the minimum pair distance is bounded away from zero can be relaxed, but then the lower bound is of the form $-\alpha n^p$ for some positive constant α and $p = 19/13$.

4.7 Steady-State Combustion

The study of the steady-state in solid fuel ignition models leads to the infinite-dimensional optimization problem

$$\min\{f_\lambda(v) : v \in H_0^1(\mathcal{D})\},$$

where $f_\lambda : H_0^1(\mathcal{D}) \to \mathbb{R}$ is the functional

$$f_\lambda(v) = \int_{\mathcal{D}} \left\{ \tfrac{1}{2}\|\nabla v(x)\|^2 - \lambda \exp[v(x)] \right\} dx,$$

and $\lambda \geq 0$ is a parameter. This problem is the variational formulation of the boundary value problem

$$-\Delta v(x) = \lambda \exp[v(x)], \ x \in \mathcal{D}, \qquad v(x) = 0, \ x \in \partial\mathcal{D}$$

where Δ is the Laplacian operator. Aris [1, pages-292-299] and Bebernes and Eberly [4], discuss this problem in the context of combustion problems.

A local minimizer of the variational formulation is a solution of the boundary value problem. The converse of this statement fails. Indeed, the boundary value problem has two solutions, but only one of these solutions is a local minimizer of the variational Bratu problem.

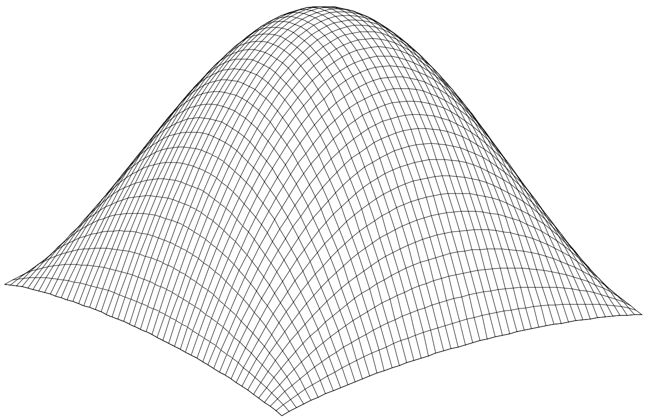

Figure 4.10: Solution to the variational Bratu problem with $\lambda = 5$

An interesting property of the variational Bratu problem is that f_λ is unbounded below for any $\lambda > 0$. This can be seen by noting that if v is any constant, positive function, then $f_\lambda(\alpha v) \to -\infty$ as $\alpha \to \infty$. Another interesting property of the variational Bratu problem is that if $\lambda_{FK} > 0$ is the Frank-Kamenetskii parameter then f_λ has a unique minimizer for $\lambda \in [0, \lambda_{FK}]$, but no minimizers for $\lambda > \lambda_{FK}$. If $\mathcal{D}$ is the unit square then $\lambda_{FK} \approx 6.81$. The solution for $\lambda = 5$ is shown in Figure 4.10.

A finite element approximation to this problem is obtained by minimizing f over the space of piecewise linear functions v with values $v_{i,j}$ at $z_{i,j}$, where $z_{i,j} \in \mathbb{R}^2$ are the vertices of a triangulation of $\mathcal{D}$ with grid spacings h_x and h_y. The values $v_{i,j}$ are obtained by solving the minimization problem

$$\min\{\sum \left(f^L_{i,j}(v) + f^U_{i,j}(v) \right) : v \in \mathbb{R}^n\},$$

where the function $f^L_{i,j}$ and $f^U_{i,j}$ are defined by

$$f^L_{i,j}(v) = \frac{h_x h_y}{4} \left\{ \left(\frac{v_{i+1,j} - v_{i,j}}{h_x} \right)^2 + \left(\frac{v_{i,j+1} - v_{i,j}}{h_y} \right)^2 - \lambda \mu^L_{i,j}, \right\},$$

$$\mu^L_{i,j} = \tfrac{2}{3} \left\{ \exp(v_{i,j}) + \exp(v_{i+1,j}) + \exp(v_{i,j+1}) \right\},$$

$$f^U_{i,j}(v) = \frac{h_x h_y}{4} \left\{ \left(\frac{v_{i-1,j} - v_{i,j}}{h_x} \right)^2 + \left(\frac{v_{i,j-1} - v_{i,j}}{h_y} \right)^2 - \lambda \mu^U_{i,j} \right\},$$

$$\mu^U_{i,j} = \tfrac{2}{3} \left\{ \exp(v_{i,j}) + \exp(v_{i-1,j}) + \exp(v_{i,j-1}) \right\}.$$

Note that in this formulation $f^L_{i,j}$ is defined only when $0 \le i \le n_x$ and $0 \le j \le n_y$, while $f^U_{i,j}$ is defined when $1 \le i \le n_x + 1$ and $1 \le j \le n_y + 1$.

The finite element discretization of the variational Bratu problem shares many of the properties of the infinite dimensional version of the problem. In particular, if $\mathcal{D}$ is the unit square then $\lambda_{FK} \approx 6.81$ for reasonable values of the mesh size. The solution of the variational Bratu problem for $\lambda < \lambda_{FK}$ is not difficult provided the starting point is chosen carefully; otherwise a minimization algorithm diverges.

4.8 Homogeneous Superconductors: 2-D Ginzburg–Landau

In the Ginzburg-Landau model for superconductivity the Gibbs free energy is

$$\int_{\mathcal{D}} \left\{ \alpha(x)|v(x)|^2 + \frac{\beta(x)}{2}|v(x)|^4 + \frac{1}{4m}\left\| \left(\frac{\hbar}{i}\nabla - \frac{2e}{c}A(x) \right) v(x) \right\|^2 + \frac{1}{8\pi}\left\| (\nabla \times A)(x) \right\|^2 \right\} dx,$$

where $\mathcal{D}$ is a three-dimensional region, $v : \mathbb{R}^3 \to \mathbb{C}$ is a complex-valued function (the order parameter) such that $|v|^2$ represents the local density of the superconducting electrons, and the vector-valued function $A : \mathbb{R}^3 \to \mathbb{R}^3$ is the vector potential. In general, the functions $\alpha : \mathcal{D} \to \mathbb{R}$ and $\beta : \mathcal{D} \to \mathbb{R}$ vary with temperature and position. The physical constants m, e, c, and $\hbar$ have the usual meaning.

The solution of the Ginzburg-Landau equations for superconductors in the presence of a magnetic field requires the determination of the functions $v : \mathbb{R}^3 \to \mathbb{C}$ and $A : \mathbb{R}^3 \to \mathbb{R}^3$ that minimize the Gibbs free energy. In the formulation of this problem, we follow Doria, Gubernatis and Rainer [12] by considering a homogeneous superconductor with a vector potential perpendicular to the superconductor. These assumptions imply that v and A depend only on the first two coordinates, and allow the Gibbs free energy functional to be expressed in the dimensionless form

$$\int_{\mathcal{D}} \left\{ -|v(x)|^2 + \tfrac{1}{2}|v(x)|^4 + \left\| [\nabla - iA(x)]\, v(x) \right\|^2 + \kappa^2 \left\| (\nabla \times A)(x) \right\|^2 \right\} dx,$$

where $v : \mathbb{R}^2 \to \mathbb{C}$ is the order parameter, $A : \mathbb{R}^2 \to \mathbb{R}^2$ is the vector potential, and κ is the Ginzburg-Landau constant. Boundary conditions are imposed on v and A. The Gibbs free energy functional is invariant under the transformation

$$v \to v \exp[iw], \quad A \to A + \nabla w,$$

for any function $w : \mathcal{D} \to \mathbb{R}$, and thus it is natural to require that $|v|$ and $\nabla \times A$ be periodic. In addition, the vector potential A must satisfy

$$\oint A(x)\, dx = 2\pi n_v,$$

where n_v is an integer. The periodicity conditions are satisfied if we require that

$$v(\xi, l_y) = v(\xi, 0), \quad A(\xi, l_y) = A(\xi, 0), \qquad 0 \le \xi \le l_x,$$

$$v(l_x,\xi) = v(0,\xi)\exp[iw(0,\xi)], \quad A(l_x,\xi) = A(0,\xi) + \nabla w(0,\xi), \qquad 0 \le \xi \le l_y.$$

Note that with this choice $\nabla \times A$ is periodic for any function $w : \mathcal{D} \to \mathbb{R}$. If we choose $w(\xi_1,\xi_2) = 2\pi n_v \xi_2$, then the vector potential also satisfies the line integral condition.

The finite dimensional approximations to the Ginzburg-Landau equations used in this problem follow the work of Doria, Gubernatis and Rainer [12], Wang and Hu [39], and Garner, Spanbauer, Benedek, Strandburg, Wright, and Plassmann [16]. Vertices $z_{i,j} \in \mathcal{D}$ for a 2-dimensional superconductor of the form $\mathcal{D} = (0,l_x)\times(0,l_y)$ are obtained by choosing grid spacings h_x and h_y and defining grid points

$$z_{i,j} = ((i-1)h_x, +(j-1)h_y), \qquad 1 \le i \le n_x+1, \quad 1 \le j \le n_y+1,$$

such that $z_{n_x+1,n_y+1} = (l_x,l_y)$. The solution of the Ginzburg-Landau equations then leads to the minimization problem

$$\min\left\{\sum\left(-|v_{i,j}|^2 + \tfrac{1}{2}|v_{i,j}|^4 + f_{i,j}(v,a)\right) : v \in \mathbb{C}^n,\ a \in \mathbb{R}^{2n}\right\},$$

where

$$\begin{aligned} f_{i,j}(v,a) \quad = \quad & \left|\frac{v_{i+1,j} - v_{i,j}\exp[ih_x a^{(1)}_{i,j}]}{h_x}\right|^2 + \left|\frac{v_{i,j+1} - v_{i,j}\exp[ih_y a^{(2)}_{i,j}]}{h_y}\right|^2 + \\ & \kappa^2\left(\frac{a^{(2)}_{i+1,j} - a^{(2)}_{i,j}}{h_x} - \frac{a^{(1)}_{i,j+1} - a^{(1)}_{i,j}}{h_y}\right)^2 . \end{aligned}$$

In this formulation $v_{i,j}$ is an approximation to V at $z_{i,j}$, and the vector $\left(a^{(1)}_{i,j}, a^{(2)}_{i,j}\right)$ is an approximation to A at $z_{i,j}$. The periodicity conditions are used to express the problem in terms of the variables $v_{i,j}, a^{(1)}_{i,j}, a^{(2)}_{i,j}$ for $1 \le i \le n_x$ and $1 \le j \le n_y$. Thus, this problem has $4n_x n_y$ variables.

The approximations used in this formulation are only first-order accurate, but on the other hand, this formulation preserves the invariance properties of the Gibbs free energy functional, that is, this formulation is invariant under the transformation

$$v_{i,j} \to v_{i,j}\exp[iw_{i,j}],$$

$$w^{(1)}_{i,j} \to w^{(1)}_{i,j} + \left(\frac{w_{i+1,j} - w_{i,j}}{h_x}\right), \quad w^{(2)}_{i,j} \to w^{(2)}_{i,j} + \left(\frac{w_{i,j+1} - w_{i,j}}{h_x}\right).$$

This invariance property is the discrete form of the invariance of the Gibbs free energy functional under the transformations $v \to v\exp[iw]$ and $A \to A + \nabla w$.

The solutions of interest in the Ginzburg-Landau model correspond to vortex configurations, where the order parameter is nonzero except at a set of isolated points (vortices). Surface and contour plots of the order parameter (see Figures 4.11 and 4.12) show the vortex structure for $\kappa = 5$ and $n_v = 8$. These plots are obtained by using the periodicity of

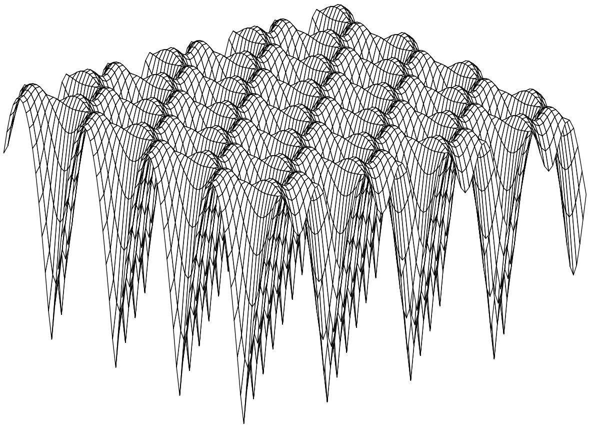

Figure 4.11: Surface plot of the order parameter for $\kappa = 5$ and $n_v = 8$

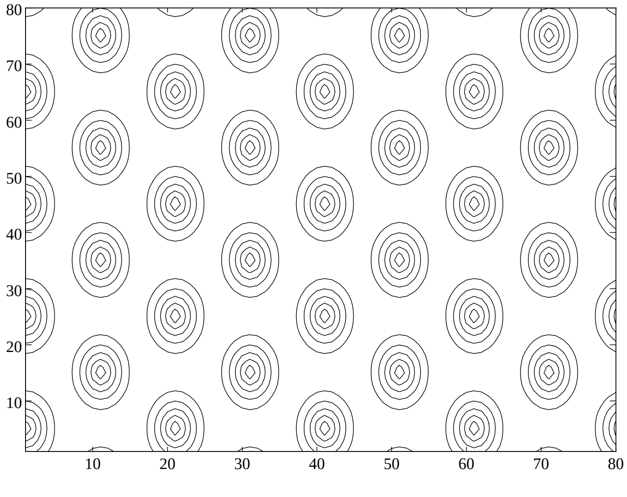

Figure 4.12: Contour plot of the order parameter for $\kappa = 5$ and $n_v = 8$

v to extend v to the domain $(0, 2l_x) \times (0, 2l_y)$, and plotting v in the larger domain. Thus, these plots show a solution v with 8 vortices in $\mathcal{D}$.

The integer n_v controls the number of vortices in $\mathcal{D}$ provided the average magnetic field

$$B_{av} = \frac{2\pi n_v \kappa}{l_x l_y}$$

is fixed. In Figures 4.11 and 4.12, $n_v = 8$ and $l_x l_y = 36\sqrt{3}$. Thus, if we want to obtain a solution with $n_v = 2$ vortices in $\mathcal{D}$ we need to decrease the area of $\mathcal{D}$ to $9\sqrt{3}$.

The regular structure shown in Figures 4.11 and 4.12 is only obtained if the sides l_x, l_y of $\mathcal{D}$ have the ratio

$$\frac{l_y}{l_x} = \sqrt{3}.$$

An interesting exercise is to explore the vortex structure for other ratios, while fixing $n_v = 8$ and $l_x l_y = 36\sqrt{3}$.

Numerical results for this 2-dimensional Ginzburg-Landau model have been obtained by Doria, Gubernatis and Rainer [12] with simulated annealing, Wang and Hu [39] with steepest descent, and Garner, Spanbauer, Benedek, Strandburg, Wright, and Plassmann [16] with limited memory quasi-Newton methods and Newton's method.

5 Software for Systems of Nonlinear Equations

The subroutines described in this section define systems of nonlinear equations of the form

$$f(x) = 0, \qquad x_l \le x \le x_u,$$

where $f : \mathbb{R}^n \to \mathbb{R}^n$ defines the residuals, and x_l, x_u are bounds on the solution. Each subroutine has a six letter name. The first letter denotes arithmetic precision: `d` for double precision and `s` for single precision (we assume double precision in the description below). The second through fourth letters describe the problem (for example, `fdc` corresponds to the `f`low in a `d`riven `c`avity problem). The fifth and sixth letters denote the purpose of the subroutine; for systems of nonlinear equations they can be `fj`, `sp`, or `js`.

Subroutines with names ending in `fj` compute the standard starting point and evaluate the function f and Jacobian matrix of f at $x \in \mathbb{R}^n$. Where appropriate, these subroutines also provide upper and lower bounds on the solution. The action of these subroutines depends on the character variable `task` as follows:

Evaluate the function if `task = 'F'`.

Evaluate the Jacobian matrix if `task = 'J'`.

Evaluate the function and the Jacobian matrix if `task = 'FJ'`.

Evaluate the standard starting point x_s if `task = 'XS'`.

Evaluate the lower bound x_l if `task = 'XL'`.

Evaluate the upper bound x_u if `task = 'XU'`.

The function value is returned in the array `fvec`, the Jacobian matrix is returned in the array `fjac` (with leading dimension `ldfjac`), and the starting point x_s and bounds x_l, x_u are returned in the array `x`.

Subroutines with names ending in `js` compute the products Js, where s is any vector and J is the Jacobian matrix at x. The result is returned in the array `y`.

Subroutines with names ending in `sp` compute the sparsity structure of the Jacobian matrix. These subroutines provide the number `nnz` of nonzeroes in the Jacobian matrix, and arrays `indrow`, and `indcol` that specify the row and column indices, respectively, of the nonzeroes in the Jacobian matrix.

Several of the problems arise from the discretization of systems of boundary value problems by a k-stage collocation method. If boundary conditions are given at $t = a$ and $t = b$, and

$$a = t_1 < t_2 < \cdots < t_{n_0} < t_{n_0+1} = b$$

is a partitioning of $[a, b]$ into n_0 subintervals, the collocation method approximates the solution to a system of p boundary value problems by a vector-valued piecewise polynomial function $u_\pi : [a, b] \to \mathbb{R}^p$. The use of the collocation method leads to a system of nonlinear equations. An advantage of the implementation of the collocation method used in these test problems is that, if m_s is the order of the s-th boundary value problem, the s-th component of

$$u_\pi^{(j-1)}(t_i), \qquad 1 \le i \le n_0, \quad 1 \le j \le m_s,$$

is directly available from the solution vector to the system of nonlinear equations. The s-th component of $u_\pi^{(j-1)}(t_i)$ is stored in the $\sigma_p(i-1) + \sigma_{s-1} + j$ location of the array `x`, where

$$\sigma_r = rk + \sum_{l=1}^{r} m_l .$$

This information is useful for obtaining plots of the j-th derivative of the s-th component of u_π. If the variable `nint` is the number n_0 of subintervals, and the variables `np` and `ns` contain σ_p and σ_{s-1}, respectively, then the pseudo-code

```
do i = 1, nint
   v(i) = x((i-1)np+ns+j)
end do
```

stores in the array `v` the value of the s-th component of $u_\pi^{(j-1)}(t_i)$. Given the array `v`, standard plotting subroutines can be used to obtain plots of $u_\pi^{(j-1)}$.

5.1 Flow in a Channel

The subroutine

```
dficfj(n,x,fvec,fjac,ldfjac,task,r,nint)
```

computes the function, Jacobian, and standard starting vector for the flow in a channel problem. The parameter `r` is the Reynolds number R, and must be positive. The parameter `nint` is the number n_0 of subintervals in the collocation method. The user must provide a positive value of `nint` and set `n = 8*nint`.

The subroutine

```
dficsp(n,nint,nnz,indrow,indcol)
```

computes the sparsity structure of J and the subroutine

```
dficjs(n,x,s,y,r,nint)
```

computes the product Js where J is the Jacobian matrix.

The tangential velocity u' is of interest in this problem. Plots of the tangential velocity can be obtained by noting that $u'(t_i)$ for $1 \le i \le n_0$, is stored in location $8(i-1)+2$ of the array `x`.

5.2 Swirling Flow between Disks

The subroutine

```
dsfdfj(n,x,fvec,fjac,ldfjac,task,eps,nint)
```

computes the function, Jacobian, and standard starting vector for the swirling flow between disks problem. The parameter `eps` is the viscosity parameter ϵ, and must be positive. The parameter `nint` is the number n_0 of subintervals in the collocation method. The user must provide a positive value of `nint` and set `n = 14*nint`.

The subroutine

```
dsfdsp(n,nint,nnz,indrow,indcol)
```

computes the sparsity structure of J and the subroutine

```
dsfdjs(n,x,s,y,eps,nint)
```

computes the product Js where J is the Jacobian matrix.

The radial velocity f', and the angular velocity g, are of interest in this problem. Plots of the radial velocity can be obtained by noting that $f'(t_i)$ for $1 \leq i \leq n_0$, is stored in location $14(i-1)+2$ of the array `x`. Plots of the angular velocity can be obtained by noting that $g(t_i)$ for $1 \leq i \leq n_0$, is stored in location $14(i-1)+9$ of the array `x`.

5.3 Incompressible Elastic Rods

The subroutine

```
dierfj(n,x,fvec,fjac,ldfjac,task,a,b,c,nint)
```

computes the function, Jacobian, and standard starting vector for the inverse elastic rod problem. The parameters `a`, `b`, and `c` are the x-coordinate a, the y-coordinate b, and the local angle of inclination c, respectively, of the free end of the elastic rod. The parameter `nint` is the number n_0 of subintervals in the collocation method. The user must provide a positive value of `nint` and set `n = 15*nint + 3`.

The subroutine

```
diersp(n,nint,nnz,indrow,indcol)
```

computes the sparsity structure of J and the subroutine

```
dierjs(n,x,s,y,a,b,c,nint)
```

computes the product Js where J is the Jacobian matrix.

The shape of the deformed elastic rod is of interest in this problem. Plots of the deformed elastic rod can be obtained by noting that the coordinates $(x(t_i),\ y(t_i))$ for $1 \leq i \leq n_0$, of points on the elastica are stored in locations $15(i-1)+1$ and $15(i-1)+6$, respectively, of the array `x`.

5.4 Solid Fuel Ignition

The subroutine

```
dsfifj(nx,ny,x,fvec,fjac,ldfjac,task,lambda)
```

computes the function, Jacobian, and standard starting vector for the solid fuel ignition problem. The parameters nx and ny are the number of interior gridpoints, n_x and n_y, in each of the respective coordinate directions. The parameter lambda is the Frank-Kamenetskii parameter λ. Values of λ in the interval [0,6.81] are of interest. A typical value is $\lambda = 5$.

The subroutine

```
dsfisp(nx,ny,nnz,indrow,indcol)
```

computes the sparsity structure of J and the subroutine

```
dsfijs(nx,ny,x,s,y,lambda)
```

computes the product Js where J is the Jacobian matrix.

Plots of the solution surface v can be obtained by noting that the value of the approximation v at the (i, j) grid point is stored in x((j-1)*nx+i).

5.5 Flow in a Driven Cavity

The subroutine

```
dfdcfj(nx,ny,x,fvec,fjac,ldfjac,task,r)
```

computes the function, Jacobian, and standard starting vector for the flow in a driven cavity problem. The parameters nx and ny are the number of interior gridpoints, n_x and n_y, in each of the respective coordinate directions. The parameter r is the Reynolds number R, and must be positive.

The subroutine

```
dfdcsp(nx,ny,nnz,indrow,indcol)
```

computes the sparsity structure of J and the subroutine

```
dfdcjs(nx,ny,x,s,y,r)
```

computes the product Js where J is the Jacobian matrix.

Streamlines and equivorticity lines of the flow are of interest in this problem. Streamlines are the contours of the stream function ψ with discrete finite difference approximation psi where

```
do j = 1, ny
   do i = 1, nx
      psi(i,j) = x((j-1)nx+i)
   end do
end do
```

Equivorticity lines are the contours of $-\Delta\psi$ with discrete finite difference approximation `v` where

```
do j = 1, ny
   do i = 1, nx
      v(i,j) = -(psi(i+1,j) - 2*psi(i,j) + psi(i-1,j))/(hx*hx)
               -(psi(i,j+1) - 2*psi(i,j) + psi(i,j-1))/(hy*hy)
   end do
end do

psi(0,j) = psi(nx+1,j) = psi(i,0) = psi(i,ny+1) = 0
```

where `hx` is the grid spacing $h_x = 1/(n_x+1)$ in the x-direction, and `hy` is the grid spacing $h_y = 1/(n_y+1)$ in the y-direction.

5.6 Human Heart Dipole

The subroutine

```
dhhdfj(n,x,fvec,fjac,ldfjac,task,prob)
```

computes the function, Jacobian, and standard starting vector for the human heart dipole problem. The user must set `n` $= 8$. The parameter `prob` specifies one of five versions of the problem. The user must set prob to the string `'HHDK'` where `K` must be replaced by an integer k with $1 \le k \le 5$. Lower and upper bounds are provided.

5.7 Combustion of Propane - Full Formulation

The subroutine

```
dcpffj(n,x,fvec,fjac,ldfjac,task)
```

computes the function, Jacobian, and standard starting vector for the full formulation of the combustion of propane problem. The user must set `n` $= 11$. Lower bounds are provided.

5.8 Combustion of Propane – Reduced Formulation

```
dcprfj(n,x,fvec,fjac,ldfjac,task)
```

computes the function, Jacobian, and standard starting vector for the reduced formulation of the combustion of propane problem. The user must set `n` $= 5$. Lower bounds are provided.

6 Software for Nonlinear Least Squares Problems

The subroutines described in this section define nonlinear least squares problems of the form

$$\min\{\|f(x)\|_2^2 : x_l \le x \le x_u\},$$

where $f : \mathbb{R}^n \to \mathbb{R}^m$ defines the residuals of the least squares problem, and x_l, x_u are bounds on the solution. Each subroutine has a six letter name. The first letter denotes arithmetic precision: `d` for double precision and `s` for single precision (we assume double precision in the description below). The second through fourth letters describe the problem (for example, `iac` corresponds to the isomerization of α-pinene – collocation formulation problem). The fifth and sixth letters denote the purpose of the subroutine; for the least squares problems they can can only be `fj`.

Subroutines with names ending in `fj` compute the standard starting point and evaluate the function f and Jacobian matrix of f at $x \in \mathbb{R}^n$. Where appropriate, these subroutines also provide upper and lower bounds on the solution. The action of these subroutines depends on the character variable `task` as follows:

Evaluate the function if `task = 'F'`.

Evaluate the Jacobian matrix if `task = 'J'`.

Evaluate the function and the Jacobian matrix if `task = 'FJ'`.

Evaluate the standard starting point x_s if `task = 'XS'`.

Evaluate the lower bound x_l if `task = 'XL'`.

Evaluate the upper bound x_u if `task = 'XU'`.

The function value is returned in the array `fvec`, the Jacobian matrix is returned in the array `fjac` (with leading dimension `ldfjac`), and the starting point x_s and bounds x_l, x_u are returned in the array `x`.

6.1 Isomerization of α-pinene – Direct Formulation

The subroutine

```
diadfj(m,n,x,fvec,fjac,ldfjac,task,nh)
```

defines the direct formulation of the α-pinene problem. The user must set `m` $= 40$ and `n` $= 5$. The parameter `nh` is the number n_0 of consecutive Runge-Kutta steps taken between observations. A typical value is `nh` $= 10$. Lower bounds are provided.

6.2 Isomerization of α-pinene – Collocation Formulation

The subroutine

```
diacfj(m,n,x,fvec,fjac,ldfjac,task,nint,sigma)
```

defines the collocation formulation of the α-pinene problem. The parameter `nint` is the number n_0 of subintervals in the collocation method. The user must provide a positive value of `nint` and set `m` $= 25 *$ `nint` $+ 40$ and `n` $= 25 *$ `nint` $+ 5$. The parameter `sigma` is the weight σ used for each constraint equation in the l_2 penalty approach. A typical value is $\sigma = 10^6$.

In this problem the approximate solution $u(\cdot, \theta)$ to the linear kinetic problem is of interest. Plots of the components of $u(\cdot, \theta)$ can be obtained by noting that the s-th component of $u(t_i, \theta)$ for $1 \leq i \leq n_0$, is stored in location $25(i-1)+5(s-1)+1$ of the array `x`.

6.3 Coating Thickness Standardization

The subroutine

```
dctsfj(m,n,x,fvec,fjac,ldfjac,task)
```

defines the coating thickness standardization problem. The user must set `m` $= 252$ and `n` $= 134$.

6.4 Exponential Data Fitting I

The subroutine

```
ddf1fj(m,n,x,fvec,fjac,ldfjac,task)
```

defines the exponential data fitting I problem. The user must set `m` $= 33$ and `n` $= 5$. Lower and upper bounds are provided.

6.5 Exponential Data Fitting II

The subroutine

```
ddf2fj(m,n,x,fvec,fjac,ldfjac,task)
```

defines the exponential data fitting II problem. The user must set `m` $= 65$ and `n` $= 11$. Lower and upper bounds are provided.

6.6 Analysis of Thermistor Resistance

The subroutine

```
datrfj(m,n,x,fvec,fjac,ldfjac,task)
```

defines the analysis of thermistor resistance problem. The user must set `m` $= 16$ and `n` $= 3$.

6.7 Analysis of an Enzyme Reaction

The subroutine

```
daerfj(m,n,x,fvec,fjac,ldfjac,task)
```

defines the analysis of an enzyme reaction problem. The user must set `m` = 11 and `n` = 4.

6.8 Chebyshev Quadrature

The subroutine

```
dchqfj(m,n,x,fvec,fjac,ldfjac,task)
```

defines the Chebyshev quadrature problem. Any positive values of `m` $\geq$ `n` are permissible. Lower and upper bounds are provided.

7 Software for Minimization Problems

The subroutines described in this section define minimization problems of the form

$$\min\{f(x) : x_l \le x \le x_u\},$$

where $f : \mathbb{R}^n \to \mathbb{R}$, and the vectors x_l, x_u specify bounds on the solution. Each subroutine has a six letter name. The first letter denotes arithmetic precision: `d` for double precision and `s` for single precision (we assume double precision in the description below). The second through fourth letters describe the problem (for example, `msa` corresponds to the `m`inimal `s`urface `a`rea problem). The fifth and sixth letters denote the purpose of the subroutine; for minimization problems they can be `fg`, `sp`, or `hs`.

Subroutines with names ending in `fg` compute the standard starting point and evaluate the function f and gradient ∇f at $x \in \mathbb{R}^n$. Where appropriate, these subroutines also provide upper and lower bounds on the solution. The action of these subroutines depends on the character variable `task` as follows:

Evaluate the function if `task = 'F'`.

Evaluate the gradient if `task = 'G'`.

Evaluate the function and the gradient if `task = 'FG'`.

Evaluate the standard starting point x_s if `task = 'XS'`.

Evaluate the lower bound x_l if `task = 'XL'`.

Evaluate the upper bound x_u if `task = 'XU'`.

The function value is returned in the variable `f`, the gradient is returned in the array `fgrad`, and the starting point x_s and bounds x_l, x_u are returned in the array `x`.

Subroutines with names ending in `hs` compute the products Hs, where s is any vector, and H is the Hessian matrix at x. The result is returned in the array `y`.

Subroutines with names ending in `sp` compute the sparsity structure of the Hessian matrix of the problem. These subroutines provide the number `nnz` of nonzeroes in the lower triangle of the Hessian matrix, and arrays `indrow`, and `indcol` that specify the row and column indices, respectively, of the nonzeroes in the lower triangle of the Hessian matrix.

In most of the problems, the vector $x \in \mathbb{R}^n$ defines a piecewise linear approximation v to the solution of the variational problem. The approximation v is defined on a triangulation of a rectangular domain $\mathcal{D}$ with n_x interior points in the x-direction and n_y interior points in the y-direction. The value of v at the (i,j) vertex of the triangulation is stored in the $(j-1)n_x + i$ location of the array `x`. Thus, the pseudo-code segment

```
do j = 1, ny
   do i = 1, nx
      v(i,j) = x((j-1)nx+i)
   end do
end do
```

stores in the array v the values of the approximation v. Given the array v, standard surface and contour plotting routines can be used to view the approximation v. In this formulation $n_y = 1$ for 1-dimensional domains.

7.1 Elastic-Plastic Torsion

The subroutine

```
deptfg(nx,ny,x,f,fgrad,task,c)
```

computes the function, gradient, and standard starting vector for the elastic-plastic torsion problem. The parameters nx and ny are the numbers of interior gridpoints, n_x and n_y, in each of the respective coordinate directions. The parameter c is the angle of twist, c, per unit length on the bar. A typical value is $c = 5$. Lower and upper bounds are provided.

The subroutine

```
deptsp(nx,ny,nnz,indrow,indcol)
```

computes the sparsity structure of the Hessian matrix and and the subroutine

```
depths(nx,ny,s,y)
```

computes the product $y = Hs$ where H is the Hessian matrix.

The resulting stress field v on the cylindrical bar is of interest in this problem. The value of the finite element approximation to the v at the (i, j) grid point is stored in x((j-1)*nx+i).

7.2 Pressure Distribution in a Journal Bearing

The subroutine

```
dpjbfg(nx,ny,x,f,fgrad,task,ecc,b)
```

computes the function, gradient, and standard starting vector for the pressure distribution in a journal bearing problem. The parameters nx and ny are the numbers of interior gridpoints, n_x and n_y, in each of the respective coordinate directions. The parameter ecc is the eccentricity, ϵ, of the journal bearing, and b specifies the domain $\mathcal{D} = (0, 2\pi) \times (0, b)$. Typical values for these parameters are $\epsilon = 0.1$ and $b = 10$. Lower bounds are provided.

The subroutine

```
dpjbsp(nx,ny,nnz,indrow,indcol)
```

computes the sparsity structure of the Hessian matrix and the subroutine

```
dpjbhs(nx,ny,s,y,ecc,b)
```

computes the product $y = Hs$ where H is the Hessian matrix. The subroutine

```
dpjbds(nx,ny,w,ecc,b)
```

determines a diagonal scaling matrix `w` as the square root of the diagonal elements of H.

The resulting pressure distribution v on the film of lubricant is of interest in this problem. The value of the finite element approximation to the v at the (i,j) grid point is stored in `x((j-1)*nx+i)`.

7.3 Minimal Surfaces

The subroutine

```
dmsafg(nx,ny,x,f,fgrad,task,bottom,top,left,right)
```

computes the function, gradient, and standard starting vector for the minimal surface area problem. The parameters `nx` and `ny` are the numbers of interior gridpoints, n_x and n_y, in each of the respective coordinate directions. The arrays `bottom,top,left,right` specify the boundary conditions for the surface.

The subroutine

```
dmsabc(nx,ny,hx,hy,bottom,top,left,right)
```

determines the boundary conditions for Enneper's minimal surface. Other boundary conditions can be obtained by modifying `dmsabc`. The subroutine

```
dmsasp(nx,ny,nnz,indrow,indcol)
```

computes the sparsity structure of the Hessian matrix and the subroutine

```
dmsahs(nx,ny,x,s,y,bottom,top,left,right)
```

computes the product $y = Hs$ where H is the Hessian matrix at x.

The resulting surface v is of interest in this problem. The value of the finite element approximation to v at the (i,j) grid point is stored in `x((j-1)*nx+i)`.

7.4 Optimal Design with Composite Materials

The subroutine

```
dodcfg(nx,ny,x,f,fgrad,task,lambda)
```

computes the function, gradient, and standard starting vector for the optimal design with composite materials problem. The parameters `nx` and `ny` are the numbers of interior gridpoints, n_x and n_y, in each of the respective coordinate directions. The parameter `lambda` is the Lagrange multiplier λ. Values of interest are $\lambda \in [0,1]$; a typical value is $\lambda = 0.008$.

The subroutine

```
dodcps(t,mu1,mu2,t1,t2,result,task,lambda)
```

computes $\psi(\sqrt{t})$, $\psi'(\sqrt{t})/t$, and $\psi''(\sqrt{t})/t$. Other functions can be obtained by modifying `dodcps`. The subroutine

```
dodcsp(nx,ny,nnz,indrow,indcol)
```

computes the sparsity structure of the Hessian matrix and the subroutine

```
dodchs(nx,ny,x,s,y,lambda)
```

computes the product $y = Hs$ where H is the Hessian matrix at x.

The norm of the gradient of the stress field $\|\nabla v\|$ is of interest in this problem. The value of the finite element approximation to v at the (i,j) grid point is stored in `x((j-1)*nx+i)`. The gradient of v can be extracted from v by a standard finite difference scheme with the pseudo-code

```
do j = 1, ny
   do i = 1, nx
      dv(i,j) = -(v(i+1,j) - 2*v(i,j) + v(i-1,j))/(hx*hx)
                -(v(i,j+1) - 2*v(i,j) + v(i,j-1))/(hy*hy)
   end do
end do

v(0,j) = v(nx+1,j) = v(i,0) = v(i,ny+1) = 0
```

where `hx` is the grid spacing $h_x = 1/(n_x+1)$ in the x-direction, and `hy` is the grid spacing $h_x = 1/(n_x+1)$ in the y-direction.

7.5 Inhomogeneous Superconductors: 1-D Ginzburg-Landau

The subroutine

```
dgl1fg(n,x,f,fgrad,t)
```

computes the function, gradient, and standard starting vector for the one-dimensional Ginzburg-Landau problem. The parameter n is the number of interior gridpoints n. The parameter t specifies the temperature t. Values of interest are $t \in [3.73, 7.32]$; a typical value is $t = 5$.

The subroutine

```
dgl1sp(n,nnz,indrow,indcol)
```

computes the sparsity structure of the Hessian matrix and the subroutine

```
dgl1hs(n,x,s,y,t)
```

computes the product $y = Hs$ where H is the Hessian matrix at x.

7.6 Molecular Conformation: Leonard-Jones Clusters

The subroutine

```
dlj2fg(n,x,f,fgrad,task,natoms)
```

computes the function, gradient, and standard starting vector for the two-dimensional Leonard-Jones problem. The parameter natoms is the number of atoms in the cluster. The user must provide a positive value of natoms and set n = 2*natoms. The structure, or molecular conformation, assumed by the atoms is of interest in this problem. The coordinates of the k-th atom are stored in (x(2k-1),x(2k)).

The subroutine

```
dlj3fg(n,x,f,fgrad,task,natoms)
```

computes the function, gradient, and standard starting vector for the three-dimensional Leonard-Jones problem. The parameter natoms is the number of atoms in the cluster. The user must provide a positive value of natoms and set n = 3*natoms. The structure, or molecular conformation, assumed by the atoms is of interest in this problem. The coordinates of the k-th atom are stored in (x(3k-2),x(3k-1),x(3k)).

7.7 Steady-State Combustion

The subroutine

```
dsscfg(nx,ny,x,f,fgrad,task,lambda)
```

computes the function, gradient, and standard starting vector for the steady-state combustion (Bratu) problem. The parameters nx and ny are the numbers of interior gridpoints, n_x and n_y, in each of the respective coordinate directions. The parameter lambda is the

Frank-Kamenetskii parameter λ of the steady-state combustion problem. Values of interest are $\lambda \in [0, 6.81]$; a typical value is $\lambda = 5.0$.

The subroutine

```
dsscsp(nx,ny,nnz,indrow,indcol)
```

computes the sparsity structure of the Hessian matrix and the subroutine

```
dsschs(nx,ny,x,s,y,lambda)
```

computes the product $y = Hs$ where H is the Hessian matrix at x.

The maximum value of v as a function of λ is of interest in this problem. This value can be obtained for a fixed `lambda` by setting $\mathrm{vmax} = \max(\mid \texttt{x(k)} \mid), 1 \leq \texttt{k} \leq n_x n_y$.

7.8 Homogeneous Superconductors: 2-D Ginzburg-Landau

The subroutine

```
dgl2fg(nx,ny,x,f,fgrad,task,w)
```

computes the function, gradient, and standard starting vector for the 2-D Ginzburg-Landau problem. The parameters `nx` and `ny` are the numbers of interior gridpoints, n_x and n_y, in each of the respective coordinate directions.

The subroutine

```
dgl2sp(nx,ny,nnz,indrow,indcol)
```

computes the sparsity structure of the Hessian matrix and the subroutine

```
dgl2hs(nx,ny,x,s,y)
```

computes the product $y = Hs$ where H is the Hessian matrix at x.

The magnitude of the complex valued order parameter v is of interest in this problem. The real and imaginary parts of the finite difference approximation to v at the (i, j) grid point are stored in `x((j-1)*nx+i)` and `x(nx*ny+(j-1)*nx+i)`, respectively.

Acknowledgments

Don Austin and Ben Rosen deserve special credit for their guidance and support during this project. We are grateful, in particular, to David Gay, Bob Maier, Paul Plassmann, Janet Rogers, and Steve Wright for their test problems. Many others contributed with comments and suggestions for improvements; in particular, we thank Uri Ascher, Karin Bennett, Larry Biegler, Burt Garbow, Jonathan Goodman, Alex Morgan, and Layne Watson.

References

[1] R. Aris, *The Mathematical Theory of Diffusion and Reaction in Permeable Catalysts*, Oxford, 1975.

[2] U. Ascher, *Solving boundary-value problems with a spline-collocation code*, J. Comput. Phy., 34 (1980), pp. 401–413.

[3] U. M. Ascher, R. M. M. Mattheij, and R. D. Russell, *Numerical solution of boundary value problems for ordinary differential equations*, Prentice Hall, 1988.

[4] J. Bebernes and D. Eberly, *Mathematical Problems from Combustion Theory*, Applied Mathematical Sciences 83, Springer-Verlag, 1989.

[5] P. Björstad, *Fast numerical solution of the biharmonic dirichlet problem on rectangles*, SIAM Journal of Numerical Analysis, 20 (1983), pp. 59–71.

[6] G. E. P. Box, W. G. Hunter, J. F. MacGregor, and J. Erjavec, *Some problems associated with the analysis of multiresponse data*, Technometrics, 15 (1973), pp. 33–51.

[7] P. Brown and Y. Saad, *Hybrid krylov methods for nonlinear systems of equations*, SIAM Journal on Scientific and Statistical Computing, 11 (1990), pp. 450–481.

[8] P. N. Brown and Y. Saad, *Hybrid Krylov subspace methods for nonlinear systems of equations*, SIAM J. Sci. Statist. Comput., 11 (1990), pp. 450–481.

[9] G. Cimatti, *On a problem of the theory of lubrication governed by a variational inequality*, Appl. Math. Optim., 3 (1977), pp. 227–242.

[10] G. Cimatti and O. Menchi, *On the numerical solution of a variational inequality connected with the hydrodynamic lubrication of a complete journal bearing*, Calcolo, 15 (1978), pp. 249–258.

[11] J. E. Dennis, D. M. Gay, and P. A. Vu, *A new nonlinear equations test problem*, Report 83-16, Rice University, Houston, Texas, 1983. Revised January 1986.

[12] M. M. Doria, J. E. Gubernatis, and D. Rainer, *On solving the Ginzburg-Landau equations by simulated annealing*, Phys. Rev. B, 41 (1990), p. 6335.

[13] C. M. Elliott and J. R. Ockendon, *Weak and Variational Methods for Moving Boundary Problems*, vol. 59 of Research Notes in Mathematics, Pittman, 1982.

[14] R. Fletcher, *Function minimization without derivatives – A review*, Comput. J., 8 (1965), pp. 33–41.

[15] J. Garner and R. Benedek, *Solution of Ginzburg-Landau equations for inhomogeneous superconductors by nonlinear optimization*, Phys. Rev. B, 42 (1990), pp. 376–385.

[16] J. Garner, M. Spanbauer, R. Benedek, K. Strandburg, S. Wright, and P. Plassmann, *Critical fields of Josephson-coupled superconducting multilayers*, Phys. Rev. B, 44 (1992), pp. 7973–7983.

[17] R. Glowinski, *Numerical Methods for Nonlinear Variational Problems*, Springer-Verlag, 1984.

[18] R. Glowinski, H. B. Keller, and L. Reinhart, *Continuation-conjugate gradient methods for the least squares solution of nonlinear boundary problems*, SIAM J. Sci. Statist. Comput., 6 (1985), pp. 793–832.

[19] J. Goodman, R. Kohn, and L. Reyna, *Numerical study of a relaxed variational problem from optimal design*, Comput. Methods Appl. Mech. Engrg., 57 (1986), pp. 107–127.

[20] M. R. Hoare, *Structure and dynamics of simple microclusters*, Advances in Chemical Physics, 40 (1979), pp. 49–135.

[21] C. L. Huang, *Applications of quasilinearization technique to the vertical channel and heat convection*, Int. J. Nonlinear Mech., 13 (1978), pp. 55–60.

[22] F. Kikuchi, *Finite element approximations to bifurcation problems of turning point type*, in Computing Methods in Applied Sciences and Engineering, R. Glowinski and J. L. Lions, eds., Lecture Notes in Mathematics, Springer-Verlag, 1977.

[23] J. Kowalik and J. F. Morrison, *Analysis of kinetic data for allosteric enzyme reactions as a nonlinear regression problem*, Math. Biosci., 2 (1968), pp. 57–66.

[24] Y. Lin and C. W. Cryer, *An alternating direction implicit algorithm for the solution of linear complementarity problems arising from free boundary problems*, Appl. Math. Optim., 13 (1985), pp. 1–17.

[25] J. N. Lyness, *When not to use an automatic quadrature routine*, SIAM Rev., 25 (1983), pp. 63–87.

[26] J. B. McLeod and S. V. Parter, *On the flow between two counter-rotating infinite plane disks*, Arch. Rat. Mech. Anal, 54 (1974), pp. 301–327.

[27] K. Meintjes and A. P. Morgan, *Chemical equilibrium systems as numerical test problems*, ACM Trans. Math. Software, 16 (1990), pp. 143–151.

[28] R. R. Meyer, *Theoretical and computational aspects of nonlinear regression*, in Nonlinear Programming, J. B. Rosen, O. L. Mangasarian, and K. Ritter, eds., Academic Press, 1970.

[29] J. J. Moré and G. Toraldo, *On the solution of large quadratic programming problems with bound constraints*, SIAM J. Optimization, 1 (1991), pp. 93–113.

[30] A. P. Morgan, A. Sommese, and L. T. Watson, *Mathematical reduction of a heart dipole model*, Journal of Computational and Applied Mathematics, 27 (1989), pp. 407–410.

[31] C. V. Nelson and B. C. Hodgkin, *Determination of magnitudes, directions and locations of two independent dipoles in a circular conducting region from boundary potential measurements*, IEEE Trans. Biomed. Eng., 28 (1981), pp. 817–823.

[32] J. C. C. Nitsche, *Lectures on Minimal Surfaces*, vol. 1, Cambridge University Press, 1989.

[33] J. A. Northby, *Structure and binding of Lennard-Jones clusters:* $13 \leq n \leq 147$, Journal of Chemical Physics, 87 (1987), pp. 6166–6177.

[34] D. P. O'Leary and W. H. Yang, *Elastoplastic torsion by quadratic programming*, Comput. Methods Appl. Mech. Engrg., 16 (1978), pp. 361–368.

[35] M. R. Osborne, *Some aspects of non-linear least squares calculations*, in Numerical Methods for Nonlinear Optimization, F. A. Lootsma, ed., Academic Press, 1972.

[36] S. V. Parter, *On the swirling flow between rotating coaxial disks: A survey*, in Theory and Applications of Singular Perturbations, W. Eckhaus and E. M. de Jager, eds., vol. 942 of Lecture Notes in Mathematics, Springer-Verlag, 1982, pp. 258–280.

[37] R. Schreiber and H. Keller, *Driven cavity flows by efficient numerical techniques*, Journal of Computational Physics, 49 (1983), pp. 310–333.

[38] I.-B. Tjoa and L. T. Biegler, *Simultaneous solution and optimization strategies for parameter estimation of differential-algebraic equations systems*, Ind. Eng. Chem. Res., 30 (1991), pp. 376–385.

[39] Z. D. Wang and C. R. Hu, *A numerical relaxation approach for solving the general Ginzburg-Landau equations for type-II superconductors*, Phys. Rev. B, 44 (1991), p. 11918.

[40] L. T. Watson and C. Y. Wang, *A homotopy method applied to elastica problems*, International Journal of Solids and Structures, 17 (1981), pp. 29–37.

[41] G. L. Xue, R. S. Maier, and J. B. Rosen, *Minimizing the Lennard-Jones potential function on a massively parallel computer*, Preprint 91-115, AHPCRC, University of Minnesota, Minneapolis, Minnesota, 1991.