Control and Cybernetics

vol. **25** (1996) No. 5

Concepts of Newton and quasi-Newton methods for optimal shape design problems

by

Manfred Laumen and Ekkehard W. Sachs
Universität Trier, FB IV – Mathematik,
D–54286 Trier, Germany
e-mail: laumen@uni-trier.de, sachs@uni-trier.de

Abstract: This paper reviews some algorithms from unconstrained optimization and their application to optimal shape design problems. The methods presented include classical algorithms like the gradient method and Newton's method. They also incorporate modern versions of quasi–Newton algorithms and inexact Newton's method which are especially suited to solve these kinds of problems. We discuss a special class of optimization problems and show the applicability of these methods. The paper concludes with a discussion of numerical algorithms for optimal shape design problems solved by the mapping method. Numerical results are presented in the last section.

1. Introduction

Optimal shape design problems constitute an important class of applications of mathematical methods. On the theoretical side there are many challenging questions in connection with the theory of partial differential equations and also with optimization theory. In the numerical solution of these types of problems there are numerous interesting issues with the numerical solution of partial differential equations, for instance finite element codes.

On the other hand efficient optimization routines are important to keep the overall computation time within reasonable limits. It is the primary goal of this paper to give some insight into optimization methods in the light of optimal shape design problems. Since various codes in this area use gradient type algorithms we present these methods together with other more modern developments in optimization. We keep the technical details of this paper at an introductory level. It is hoped that the reader not so familiar with optimization will obtain some information on modern technology of numerical optimization algorithms.

Among the numerous applications of optimal shape design problems we mention only a few examples such as the design of a nozzle by Pironneau (1991), a thermal diffuser problem by Delfour, Payre and Zolesio (1983;1986), an airfoil boundary problem by Pironneau and Vossinis (1991), various beams and plates by Haslinger and Neittaanmäki (1988), Sokolowski and Zolesio (1987;1992) and thermal tomography problems by Banks and Kojima (1988;1989).

All of these problems involve a costly evaluation of an objective function. This is due to the fact that a solution of a partial differential equation is included in this procedure. The resulting optimization problem is very large and the special structure should be made use of in the code.

An optimal shape design problem can often be cast into the form to minimize an objective function, see e. g. (16)

$$J(u,y) \quad \text{where} \quad a(u;y,\eta) = l(u;\eta) \quad \forall \eta.$$

The equality constraint is given by a variational equality described by a bilinear form a and a linear functional l depending nonlinearly on the design parameter. It is reasonable to expect that for each design variable u there exists a solution of the variational equality which is denoted by $y(u)$. Then the optimization problem can be rewritten as an unconstrained problem

$$\text{Minimize} \quad f(u) = J(u, y(u)).$$

The advantage of this formulation is that the structure of the problem is taken care of automatically and that the number of variables for the objective function f is equal to the number of design variables u which is often much smaller than the number of the discretized points of the state variable y.

We want to point out that one could also pursue the approach to minimize $J(u,y)$ with regard to both variables y and u under the additional equality constraint $a(u;y,\eta) = l(u;\eta)$. This formulation would lead to SQP (Sequential Quadratic Programming) methods for a numerical solution. However, in order to use the special structure of optimal shape design problems, one would have to take special care in selecting special SQP methods. This leads to reduced SQP methods which exploit sparsity and the fact that a relatively small number of design variables might be present. These methods are not the issue in this paper and the reader is referred to Kupfer and Sachs (1991), Kunisch and Sachs (1992), Kupfer (1995), and Sachs (1994).

In the first sections we review the gradient method in connection with a step size rule which has an important impact on the convergence behavior. Following this we discuss Newton's method and its advantages and drawbacks.

Quasi–Newton methods are presented in some more detail including an outline of the convergence theory due to Dennis and Moré. We include a short description of quasi–Newton methods with structure. Another way of reducing the large computing time per iteration is based on the concept of inexact Newton methods. We give the basic framework of these methods and some remarks on the implementation of these powerful variants of Newton's method.

In the last three sections we apply these methods to optimal shape design problems. Given a general structure of optimization problems which often occurs in optimal shape design, we show how to compute the gradient and the second derivative. We also address the fact that a Hessian-vector multiplication can be achieved at the cost of a gradient evaluation. This issue makes inexact Newton methods with iterative solvers for the linear subproblems very competitive.

2. Unconstrained optimization

Let us consider the following general unconstrained optimization problem. Assume that

$$f : I\!R^n \to I\!R$$

is a smooth function in the sense given below. Then we arrive at the following

Optimization problem

$$\min \; f(x)$$

It is well known that a suitable substitute problem is to look for stationary points.

Stationary points for optimization problem

$$\text{Find } x_* \text{ with } \nabla f(x_*) = 0.$$

There are various applications in the literature and in practice.

Examples

- Optimal Shape Design
- Optimal Control
- Design of Feedback Control Laws

In the sequel we use the following notation

$$\begin{aligned} \text{current iterate} &\quad x_c, \\ \text{new iterate} &\quad x_+, \\ \text{previous iterate} &\quad x_-. \end{aligned}$$

3. Gradient method

The most popular gradient method is the method of steepest descent. It is easy to implement and consists of the following steps.

We assume that a starting point or a current iterate $x_c \in I\!R^n$ is given. The new iterate x_+ is then computed as follows

Algorithm 1

```
Given      $x_c$
Compute    $\nabla f(x_c)$
Set        $x_+ = x_c - \alpha_c \nabla f(x_c)$
```

Here the step size $\alpha_c \in I\!R$ could be constant or is determined by a line search technique such as Armijo's rule as presented in the next section.

We make the following assumption for convergence statements

Assumption 1 *Let the following hold for the function f*

$$\begin{array}{cl} f & \textit{contiously differentiable} \\ \nabla f & \textit{Lipschitz-continuous} \\ f & \textit{strictly convex} \end{array}$$

The last assumption on the global strict convexity can be relaxed to a local type at the expense of a local convergence statement instead of the following

Theorem 1 *Let Assumption 1 hold. Consider a sequence of iterates produced by the Algorithm 1.1, the steepest descent method. Then there exists $c \in (0, 1)$ such that*

$$\|x_+ - x_*\| \le c\|x_c - x_*\|,$$

i.e. a q-linear rate of convergence holds.

If one cannot show a local strict convexity around the optimal point then the rate of convergence can deteriorate to an estimate like

$$\|x_k - x_*\| \le c_1 \frac{1}{k}$$

for some $c_1 > 0$ and all $k \in I\!N$. As one can check easily this is an extremely slow convergence rate for the convergence of x_k to x_*. References and proofs of these theorems can be found in many textbooks on optimization, see e. g. Gruver and Sachs (1980).

4. Gradient method with step size rule

Gradient methods without step size rules can be found in several applications of optimal shape design. Unless one tries to avoid with this strategy local minimum early in the iteration process it is more advisable to incorporate a step size rule like Armijo's (1966) rule. There are many other step size strategies but we will concentrate on this one. Armijo's rule can be used for any algorithm using a descent direction, i. e. a direction $d \in I\!R^n$ with $\nabla f(x_c)^T d < 0$ at any point x_c. An example for such a direction is the negative gradient $d = -\nabla f(x_c)$. The Armijo rule can be implemented in the following way.

Algorithm 2 *Choose* $\rho \in (0,1)$, $\gamma > 1$, $\beta \in (0,1)$

```
Given                      $x_c$
Comp. a descent direction  $\nabla f(x_c)^T d < 0$
Increase $j \in I\!N$ until    $f(x_c + \gamma^j d) - f(x_c) > \rho \gamma^j \nabla f(x_c)^T d$
Set                        $\alpha_0 = \gamma^j$
Increase $j \in I\!N$ until    $f(x_c + \beta^j \alpha_0 d) - f(x_c) \leq \rho \beta^j \alpha_0 \nabla f(x_c)^T d$
Set                        $\alpha_c = \beta^j \alpha_0$
Set                        $x_+ = x_c - \alpha_c d$
```

This step size rule ensures that the new point will have a sufficient decrease and also that the step size rule cannot become arbitrarily small which could destroy the convergence of the method. Both properties are used in the following convergence theorem.

Theorem 2 *Let Assumption 1 hold except for the convexity requirement. If f is bounded from below then we have*

$$\lim_{k \to \infty} \nabla f(x_k)^T d_k = 0.$$

In the case of steepest descent, $d_k = -\nabla f(x_k)$, we obtain $\nabla f(x_k) \to 0$, i. e. each accumulation point of the sequence x_k is a stationary point.

5. Newton method

A method which improves dramatically the convergence behavior locally is Newton method.

Algorithm 3

```
Given  $x_c$
Solve  $\nabla^2 f(x_c) s = -\nabla f(x_c)$
Set    $x_+ = x_c + s$
```

The smoothness requirements for f are as follows.

Assumption 2 *Let the following hold for the function f in a neighborhood of the optimal point x_*.*

f	*twice continuously differentiable*
$\nabla^2 f$	*Lipschitz-continuous*
$\nabla^2 f(x_*)$	*invertible*

The rate of convergence which can be shown is quadratic.

Theorem 3 *Let Assumption 2 hold. If $\|x_0 - x_*\|$ is sufficiently small, then the Algorithm 3 is well defined and the distance from the solution can be estimated by*

$$\|x_+ - x_*\| \leq c \|x_c - x_*\|^2,$$

i.e. a quadratic rate of convergence.

References for Newton method are the books by Ortega and Rheinboldt (1970), Dennis and Schnabel (1983) and Kelley (1995). Due to this theorem the convergence of Newton method within the region of convergence is extremely fast and often offsets the additional cost of the computation of the second derivative. However, improvement can be made on certain aspects of Newton method.

- Computation of $\nabla^2 f(x_c)$
- Solving a linear system of equations at each iteration
- Local convergence

All these disadvantages have been addressed in the literature and we list some of the possibilities to alleviate these drawbacks.

- Quasi-Newton updates, Shamanski's method (evaluate new Hessian after r steps)
$$x_{k+j+1} = x_{k+j} - \nabla^2 f(x_k)^{-1} \nabla f(x_{k+j}), \quad j = 1, ..., r$$
- Nested iteration (infinite dimensional problems)
- Globalisation (Step size rule, Trust–Region–method)

In the sequel we will address some of the points mentioned previously which increase the efficiency in the use of optimization algorithms.

6. Quasi–Newton method

Quasi–Newton methods which are sometimes also called variable metric methods or secant methods have been successfully applied to various unconstrained optimization problems in the past decades. We do not give reference to the original papers but refer the interested reader to Dennis and Schnabel (1983) where an overview of the literature is given.

The main idea is to replace $\nabla^2 f(x_c)$ by some approximation $B_c \in I\!R^{n\times n}$ The basic algorithm can formulated in the following way.

Algorithm 4

```
*Given*     $x_c, B_c$
*Solve*     $B_c s = -\nabla f(x_c)$
*Set*       $x_+ = x_c + s$
*Compute*   $B_+$
```

In the next paragraph we give some motivation on how to choose the new update for the approximation of the Hessian. In the one-dimensional case the secant method determines uniquely a number B_+ by

$$B_+ = \frac{f'(x_+) - f'(x_c)}{x_+ - x_c}$$

This equation can be generalized to the more dimensional case as a requirement which a new update should fulfil. It is called the secant equation

$$B_+(x_+ - x_c) = \nabla f(x_+) - \nabla f(x_c) \tag{1}$$

or

$$B_+ s = y \quad \text{with} \quad y = \nabla f(x_+) - \nabla f(x_c). \tag{2}$$

If $n > 1$ then there are infinitely many $B_+ \in I\!R^{n\times n}$ which satisfy (1). On the other hand B_c contains already information on the Hessian accumulated from the previous steps. Therefore one wants to retain this information also for B_c. Hence one chooses B_+ as the matrix being closest to the previous update B_c which satisfies the scant condition (1).

$$B_+ \ \text{solves} \ \min \|B - B_c\|_F \quad \text{with} \ Bs = y, \ \ B \in I\!R^{n\times n}. \tag{3}$$

Here $\|\cdot\|_F$ denotes the Frobenius norm which yields a unique solution to (3). It can be shown that the solution of (3) can be computed explicitly and is given by the

Broyden update

$$B_+ = B_c + \frac{(y - B_c s)s^T}{s^T s}.$$

It is obvious that this update does not preserve the symmetry. Another update which has this property is the Symmetric-Rank-1 update.

SR1 update

$$B_+ = B_c + \frac{(y - B_c s)(y - B_c s)^T}{(y - B_c s)^T s}.$$

In the sufficiency condition for minimal points it is required that the Hessian be positive definite at the optimum. Hence it is reasonable to require this property also for the approximation of the Hessian. We give two examples of updates of this type. The first one was introduced by Davidon, Fletcher and Powell.

DFP update

$$B_+ = B_c + \frac{(y - B_c s)y^T + y(y - B_c s)^T}{y^T s} - (y - B_c s)^T s \frac{yy^T}{(y^T s)^2}.$$

The next update is attributed to Broyden, Fletcher, Goldfarb and Shanno. It is the most widely used update in the context of unconstrained optimization.

BFGS update

$$B_+ = B_c + \frac{yy^T}{y^T s} - \frac{B_c s s^T B_c}{s^T B_c s}.$$

7. Convergence analysis

A detailed convergence analysis has been given by Dennis and Moré which can be read in the books by Dennis and Schnabel (1983) and Kelley (1995). The convergence of quasi–Newton methods can be split into two parts. The first condition guarantees the convergence and the q-linear rate. This condition describes in which way the distance of the new approximation from the true Hessian can be estimated by the corresponding distance of the old one.

Assumption 3 (Bounded deterioration property) *Let x_* denote a stationary point. An update formula for B satisfies the bounded deterioration property if there is a constant c_1 such that for all x_c, B_c the new x_+, B_+ satisfy*

$$\|B_+ - \nabla^2 f(x_*)\| \leq \|B_c - \nabla^2 f(x_*)\|(1+\sigma_c) + c_1\sigma_c$$

with $\sigma_c = \max(\|x_+ - x_*\|, \|x_c - x_*\|)$.

If an arbitrary update formula satisfies this property then the following theorem ensures linear convergence.

Theorem 4 *Let an update formula satisfy the Bounded Deterioration Property (Assumption 3). Let Assumption 2 hold for x_*. If $\|B_0 - \nabla^2 f(x_*)\|$ and $\|x_0 - x_*\|$ are sufficiently small, then the quasi–Newton method is well defined and there is $c \in (0,1)$ such that*

$$\|x_+ - x_*\| \leq c\|x_c - x_*\|,$$

q-linear convergence, holds.

If a faster rate of convergence should be obtained then a more stringent requirement on the closeness of the approximations of the Hessian to the true Hessian is required. The following property ensures the superlinear rate of convergece. Note that this does not require the convergence of the approximations to the true Hessian.

Assumption 4 (Dennis–Moré condition) *Let x_* be a stationary point. An update formula for B satisfies the Dennis–Moré condition, if the steps $s_k = x_{k+1} - x_k$ produced by the quasi–Newton method satisfy*

$$\lim_{k\to\infty} \|(B_k - \nabla^2 f(x_*))\frac{s_k}{\|s_k\|}\| = 0$$

with the updates B_k.

This assumption is necessary and sufficient for the superlinear rate of convergence.

Theorem 5 *Let an update formula for B be given and consider a sequence x_k, B_k produced by the quasi–Newton method. Assume that the iterates converge to a stationary point x_*, i. e. $x_k \to x_*$, which satisfies Assumption 2. Then the q-superlinear rate of convergence*

$$\lim_{k\to\infty} \frac{\|x_{k+1} - x_*\|}{\|x_k - x_*\|} = 0$$

holds if and only if the Dennis–Moré condition holds.

This framework can be applied to Broyden, BFGS, DFP and other updates to obtain a convergence statement of the following type:

Theorem 6 *Suppose that Assumption 2 holds. If (x_0, B_0) are sufficiently close to $(x_*, \nabla^2 f(x_*))$ then the quasi–Newton method with the Broyden, DFP or BFGS update is well defined and the rate of convergence of x_k to x_* is superlinear.*

8. Quasi-Newton methods with structure

In various applications it is possible to compute a certain part of the Hessian at reasonable cost whereas other parts are prohibitively expensive to calculate. The standard examples are minimization problems of nonlinear least squares type which can also be found among optimal shape design problems.

Let $R : I\!R^n \to I\!R^m$ be a nonlinear map such that

$$f(x) = \frac{1}{2}\|R(x)\|^2 = \frac{1}{2}\sum_{l=1}^{m} r_l(x)^2.$$

Then the first derivative is

$$\nabla f(x) = J(x)^T R(x), \qquad \text{Jacobian} \quad J(x) \quad \text{of} \quad R(x)$$

and the second derivative is given by

$$\nabla^2 f(x) = J(x)^T J(x) + \sum_{l=1}^{m} r_l(x) \nabla^2 r_l(x).$$

In a standard quasi–Newton method one would approximate all of the Hessian by an update. Since one has to compute the Jacobian $J(x)$ in any case to evaluate the right hand side of the linear system to be solved, the first term in the sum of the Hessian is known exactly and would not need to be approximated.

In structured quasi–Newton updates one distinguishes between a computed part $C(x)$ and an approximated part $A(x)$ of the Hessian:

$$\nabla^2 f(x) = C(x) + A(x).$$

Clearly, in the case of nonlinear least squares we choose

$$C(x) = J(x)^T J(x), \quad A(x) = \sum_{l=1}^{m} r_l(x) \nabla^2 r_l(x).$$

The step for a structured quasi–Newton method requires the solution of

$$(C(x_c) + A_c)s = -\nabla f(x_c).$$

The update for the matrix A should satisfy the secant equation (2) and from that we obtain a modification $y^{\#}$ of y for the update formula. Altogether we obtain the following algorithm in the case of a structured Broyden update.

Algorithm 5

```
Given   $x_c, A_c$
Solve   $(C(x_c) + A_c)s = -\nabla f(x_c)$
Set     $x_+ = x_c + s$
Set     $y^{\#} = \nabla f(x_+) - \nabla f(x_c) - C(x_+)s$
Set     $A_+ = A_c + \frac{(y^{\#} - A_c s)s^T}{s^T s}$
```

Note that indeed the secant equation holds

$$(C(x_+) + A_+)s = C(x_+)s + y^{\#} = \nabla f(x_+) - \nabla f(x_c).$$

In the last section have outlined a principal concept for the convergence proof which can be adapted to various special cases. With regard to this one has to check the Bounded Deterioration Property and the Dennis–Moré condition for the approximations $C(x_k) + A_k$ of the Hessian. The following theorem holds for the structured Broyden update.

Theorem 7 *Let the Assumption 2 be satisfied. If (x_0, A_0) are sufficiently close to $(x_*, A(x_*))$ then the iterates x_k produced by the structured Broyden–update are well defined and converge to x_* at a superlinear rate.*

Similar theorems hold for the DFP and BFGS update and under additional assumption for the SR–1 update. For proofs of these methods we refer to Dennis and Schnabel (1983).

In a more recent publication, Huschens (1994) exploits not only additive structure as in this section but also multiplicative structure. This yields an efficient modification of the Gauß–Newton method to solve nonlinear least squares problems.

9. Inexact Newton method

At each step of Newton's method a linear system of equations has to be solved. There is an abundance of methods available to solve these subproblems.

If one has to solve large scale linear systems which easily occur in the presence of discretized differential equations, one has to use iterative methods like Conjugate Gradient or GMRES. The main advantage of these methods is that they only need a subroutine which outputs the result of a matrix-vector product but not the matrix itself.

The reason for the need of iterative methods is the size of the problem. In many cases it is no longer possible to store a matrix due to the size or due to the way it enters the problem. The linearized state equation which is a matrix-vector product might be solved with a special finite element software. Hence no explicit matrix formulation describing the linear transformation for the linearized state equation is available.

Another reason for using iterative methods is the condition number which occurs for some discretized problems being prohibitively high for a direct method leading to an incorrect solution. Here preconditioned GMRES or CG methods are a useful approach to this difficulty.

If a linear system is solved iteratively the question of a proper termination criterion arises. Although one could run the iteration up to the accuracy of the roundoff error this might not be advantageous. When far away from the solution it could be sufficient to obtain only an approximate solution of the linear system. This way one could save computing time in the initial process. This concept is called inexact Newton method.

Algorithm 6 *Choose a sequence $\eta_k > 0$.*

```
Given        $x_c$
Find s with  $\|\nabla^2 f(x_c)s + \nabla f(x_c)\| \leq \eta_c \|\nabla f(x_c)\|$
Set          $x_+ = x_c + s$
```

The next theorem deals with the question of how small the tolerance should be to keep the fast rate of the overall algorithm.

Theorem 8 *Let Assumption 2 hold. If x_0 is sufficiently close to x_* then there is $\eta_* > 0$ such that in dependence of the choice for η_k the following rates of convergence are obtained for the sequence x_k which is produced by the inexact Newton method.*

$$\begin{aligned} 0 < \eta_k \leq \eta_* \quad &\Rightarrow \quad \textit{linear convergence,} \\ \eta_k \to 0 \quad &\Rightarrow \quad \textit{superlinear convergence,} \\ \eta_k \leq c\|\nabla f(x_k)\| \quad &\Rightarrow \quad \textit{quadratic convergence.} \end{aligned}$$

A proof has been given by Dembo, Eisenstat and Steihaug (1982). The theorem clearly states that the residual of the solution of the linear system is the measure for the termination criterion of the iterative linear solver. The closer the iterates x_k approach a stationary point the more accurate the solution has to be. Extensions of these methods have been investigated in various recent papers, e. g. by Brown and Saad (1994), Eisenstat and Walker (1994).

10. Special optimization problems

In this section we take a look at optimization problems with a particular structure. This structure shows up in many optimal shape design problems. We use this simplified structure in order to clarify some of the implementational details.

$$\text{Minimize } \phi(y,x) \text{ such that } e(y,x)=0. \tag{4}$$

If $e(y,x)=0$ denotes a differential equation with y being the solution and x the design parameter, then this equation can often be solved uniquely in y for each x. This is true under the following assumption.

Assumption 5 *Let $e : \mathbb{R}^{n+m} \longrightarrow \mathbb{R}^m$ and $\phi : \mathbb{R}^{n+m} \longrightarrow \mathbb{R}$ continuously differentiable on $\mathbb{R}^{n+m}$. Let $e_y(y,x) \in \mathbb{R}^{m\times m}$ be invertible for all $(y,x) \in \mathbb{R}^{n+m}$.*

The implicit function theorem allows to define the following map.

Lemma 1 *Let Assumption 5 hold and let x_* and y_* satisfy $e(y_*,x_*)=0$. Then there exists a unique mapping in a neighborhood U of x_**

$$y : U \longrightarrow \mathbb{R}^m,$$

with

$$e(y(x),x)=0 \quad \forall x \in U.$$

Furthermore, $y(\cdot)$ is continuously differentiable and its derivative $y'(x) \in \mathbb{R}^{m\times n}$ is given by

$$y'(x) = -e_y(y(x),x)^{-1} e_x(y(x),x) \quad \forall x \in \mathbb{R}^m. \tag{5}$$

Then one can define the following optimization problem which is equivalent to (4)

$$\text{Minimize } f(x)=\phi(y(x),x) \tag{6}$$

This is an unconstrained optimization problem. The gradient of this function can be computed if $y'(x)$ is known. Note, however, that this requires the computation of the $m\times n$ matrix $y'(x)$ of sensitivities, i. e. the solution of n linear systems of the size $m\times m$.

An alternative approach via the idea of an adjoint variable is shown in the next lemma. In this case only one linear system (7) of m equations in m unknowns has to be solved.

Lemma 2 *Let Assumption 5 hold. Define for $x \in I\!R^n$ a vector $p(x) \in I\!R^m$ as the unique solution of*

$$e_y(y(x),x)^T p(x) = -\nabla_y \phi(y(x),x). \tag{7}$$

Then

$$\nabla f(x) = e_x(y(x),x)^T p(x) + \nabla_x \phi(y(x),x). \tag{8}$$

Proof
The proof follows from the chain rule and (5).

$$\begin{aligned} \nabla f(x) &= y'(x)^T \nabla_y \phi(y(x),x) + \nabla_x \phi(y(x),x) \\ &= -e_x(y(x),x)^T e_y(y(x),x)^{-T} \nabla_y \phi(y(x),x) + \nabla_x \phi(y(x),x) \\ &= e_x(y(x),x)^T p(x) + \nabla_x \phi(y(x),x). \end{aligned}$$

□

The next step is to compute the second derivative.

Assumption 6 *Let $e : I\!R^{n+m} \to I\!R^m$ and $\phi : I\!R^{n+m} \to I\!R$ twice continuously differentiable on $I\!R^{n+m}$. Let $e_y(y,x)$ be invertible for all $(y,x) \in I\!R^{n+m}$.*

In order to simplify the notation we define the Lagrangian $L : I\!R^{2m+n} \to I\!R$ as

$$L(y,x,p) = e(y,x)^T p + \phi(y,x).$$

Then equation (7) can be rewritten as $L_y(y(x),x,p(x)) = 0$. The gradient of f is given by

$$\nabla f(x) = L_x(y(x),x,p(x)).$$

Differentiating f a second time requires the differentiation of p.

Lemma 3 *Let Assumption 6 hold. Then $p(x)$ as defined in (7) is differentiable and*

$$\begin{aligned} p'(x) &= -e_y(y(x),x)^{-T} h_x(p(x),x) \\ &= e_y(y(x),x)^{-T}(L_{yx}(y(x),x,p(x)) + L_{yy}(y(x),x,p(x))y'(x)). \end{aligned} \tag{9}$$

Proof
Define a map $h : I\!R^{m+n} \to I\!R^m$ by

$$h(p,x) = L_y(y(x),x,p(x)) = e_y(y(x),x)^T p + \nabla_y \phi(y(x),x)$$

where by (7) $p(x)$ solves $h(p(x),x) = 0$. Since $h_p(p,x) = e_y(y(x),x)^T$ is invertible by Assumption 5 the implicit function theorem, Lemma 1, can be applied again and (9) follows.

□

Similarly to the sensitivity equations for $y'(x)$ the computation of $p'(x)$ requires the solution of n systems of the size $m \times m$.

The Hessian of f can be represented by using the Lagrangian.

Theorem 9 *Let Assumption 6 hold. Then f is twice differentiable.*

$$\begin{aligned} \nabla^2 f(x) \;=\; & L_{xx}(y(x),x,p(x)) + y'(x)^T L_{yx}(y(x),x,p(x)) \\ & + L_{xy}(y(x),x,p(x))y'(x) + y'(x)^T L_{yy}(y(x),x,p(x))y'(x). \end{aligned} \tag{10}$$

Proof

We use the representation $\nabla f(x) = L_x(y(x),x,p(x))$ to derive

$$\begin{aligned} \nabla^2 f(x) &= \tfrac{d}{dx} L_x(y(x),x,p(x)) \\ &= e_x(y(x),x)^T p'(x) \\ &\quad + L_{xx}(y(x),x,p(x)) + L_{xy}(y(x),x,p(x))y'(x). \end{aligned} \tag{11}$$

With (9) and (5)

$$\begin{aligned} \nabla^2 f(x) &= e_x(y(x),x)^T e_y(y(x),x)^{-T} h_x(p(x),x) \\ &\quad + L_{xx}(y(x),x,p(x)) + L_{xy}(y(x),x,p(x))y'(x) \\ &= y'(x)^T h_x(p(x),x) \\ &\quad + L_{xx}(y(x),x,p(x)) + L_{xy}(y(x),x,p(x))y'(x) \end{aligned}$$

which proves (10) using

$$h_x(p(x),x) = L_{yx}(y(x),x,p(x)) + L_{yy}(y(x),x,p(x))y'(x).$$

□

It is important to note that the computation of the Hessian with this Theorem 9 requires the solution of n linear systems of the size $m \times m$ for $y'(x)$. The computation of $p'(x)$ is not needed in this case.

If iterative methods are used to solve Newton equation $\nabla^2 f(x)s = -\nabla f(x)$ like in inexact Newton methods the following theorem describing the matrix-vector multiplication is an alternative.

Theorem 10 *Let Assumption 6 hold. Then for $s \in \mathbb{R}^n$ the multiplication $\nabla^2 f(x)s$ can be carried out as follows. Solve*

$$e_y(y(x),x)v = -e_x(y(x),x)s \tag{12}$$

for $v \in \mathbb{R}^m$ and

$$e_y(y(x),x)^T w = L_{yx}(y(x),x,p(x)) + L_{yy}(y(x),x,p(x))v \tag{13}$$

for $w \in \mathbb{R}^m$. Then we have

$$\nabla^2 f(x)s = e_x(y(x),x)w + L_{xx}(y(x),x,p(x))s + L_{xy}(y(x),x,p(x))v \tag{14}$$

Proof

From (5) we obtain that $v = y'(x)s$. Equation (9) shows that $w = p'(x)s$. Then (14) can be rewritten as

$$\begin{aligned}\nabla^2 f(x)s &= e_x(y(x), x)p'(x)s \\ &\quad + L_{xx}(y(x), x, p(x))s + L_{xy}(y(x), x, p(x))y'(x)s,\end{aligned}$$

which proves (14) using (11).

□

This shows that at each iteration of an iterative solver for Newton equation the solution of two linear systems of the size $m \times m$ is required in contrast to n for the computation of the complete Hessian.

It is very much problem dependent which of the two approaches outlined above should be preferred.

11. Optimal shape design problem

In this section we show how to apply some of the previous results to a problem in optimal shape design. For simplicity we state this problem in the infinite dimensional setting as presented in the doctoral thesis of Laumen (1996d). This is no drawback because all of the previous results can be extended to the infinite dimensional case. Moreover the discretized finite dimensional optimal shape design problem will not only be of the structure shown in the previous section, but the infinite dimensional formulation enables us also to compare the infinite dimensional iterates with the discretized ones leading to a mesh independence behavior Laumen (1996b).

Let the bounded domain $\tilde{\Omega} \subset I\!R^2$ be parametrized by a function $u \in U_{ad}$ and U_{ad} a subset of a function space U defined on an interval $I \subset I\!R$. Let $\tilde{J} : U \times \tilde{V}$ be twice Fréchet-differentiable and the Sobolev space $\tilde{V}$ be given by

$$\tilde{V}(\tilde{\Omega}) = \{v \in H^1(\tilde{\Omega}) \ : \ \gamma_0 \phi|_{\tilde{\Gamma}_0} = 0\}$$

with the partition $\tilde{\Omega} = \tilde{\Gamma}_0 \cup \tilde{\Gamma}_1$ and the trace operator $\gamma_0 \in L(H^1(\tilde{\Omega}), H^{\frac{1}{2}}(\tilde{\Gamma}_0))$.

Then the optimal shape design problem is

$$\min_{u \in U_{ad}} \tilde{J}(u, \tilde{y}), \quad \text{where } \tilde{a}(\tilde{y}, \tilde{\eta}) = \tilde{l}(\tilde{\eta}) \quad \forall\, \tilde{\eta} \in \tilde{V} := \tilde{V}(\tilde{\Omega}), \tag{15}$$

We use the mapping method to transform the problem onto a fixed domain $\Omega = (0, 1) \times I$. For the fixed domain problem we drop the notation with the tilde. This yields

$$\min_{u \in U_{ad}} J(u, y), \quad \text{where } a(u; y, \eta) = l(u; \eta) \quad \forall\, \eta \in V, \tag{16}$$

with a general bilinear form

$$a(u; y, \eta) = \sum_{|i|,|j| \leq 1} \langle a_{ij}(u) D^i y, D^j \eta \rangle_{\mathcal{L}^2(\Omega)} + \langle b(u) y, \eta \rangle_{\mathcal{L}^2(\Gamma_1)},$$

and an arbitrary linear functional

$$l(u;\eta) = \sum_{|i|\leq 1} \langle f_i(u), \eta\rangle_{\mathcal{L}^2(\Omega)} + \langle f(u), \eta\rangle_{\mathcal{L}^2(\Gamma_1)}.$$

In this problem the domain is fixed at the expense of the fact that the coefficients in the variational equality depend nonlinearly on the parameter function $u \in U_{ad}$.

Note that this formulation is of similar type as the optimization problem (4), where the equality constraint is defined by the variational equality (16). We can prove that for each u there exists a unique $y(u)$ under the following assumption:

Assumption 7 *Let $\Omega \subset I\!\!R^2$ have a Lipschitz continuous boundary, $a(u;\cdot,\cdot)$ be a V-elliptic bilinear form for all $u \in \mathcal{U}_{ad}$, and $f(u) \in V'$, $a_{ij}(u) \in \mathcal{L}^\infty(\Omega)$, $b(u) \in \mathcal{L}^\infty(\Gamma_1)$. Let f, a_{ij}, and b be once continuously Fréchet-differentiable in u.*

The following statements can be found in Laumen (1996c) for this problem.

Theorem 11 *Let Assumption 7 be true. For each $l(u;\cdot) \in V'$, there exists a unique solution $y \in V$ and a unique solution $p \in V$ of the two variational equations*

$$\begin{array}{rcll} a(u;y,\eta) & = & l(u;\eta) & \forall\, \eta \in V, \\ a(u;\eta,p) & = & l(u;\eta) & \forall\, \eta \in V, \end{array} \tag{17}$$

which define solution operators $y : U_{ad} \to V$ with $y = y(u)$ and $p : U_{ad} \to V$ with $p = p(u)$.

Thus we are able to rewrite the optimization problem (16) as an unconstrained one

$$\min_{u\in U_{ad}} f(u), \quad \text{where} \quad f(u) = J(u, y(u)) \tag{18}$$

Under additional smoothness assumptions on the functions involved one can prove how to compute the first derivative.

Theorem 12 *The solution operators $y : U_{ad} \to V$ and $p : U_{ad} \to V$ are Fréchet-differentiable with $y'(u)$, $p'(u) \in L(U,V)$. Moreover, $\hat{y}^v := y'(u)(v)$ and $\hat{p}^v := p'(u)(v)$ are the unique solutions of the variational problems*

$$a(u;\hat{y}^v,\eta) = l_u(u;\eta)(v) - a_u(u;y(u),\eta)(v) \qquad \forall \eta \in V, \tag{19}$$

$$a(u;\eta,\hat{p}^v) = l_u(u;\eta)(v) - a_u(u;\eta,p(u))(v) \qquad \forall \eta \in V, \tag{20}$$

with

$$a_u(u;y,\eta)(v) = \sum_{|i|,|j|\leq 1} \langle a'_{ij}(u)(v) D^i y, D^j \eta\rangle_{\mathcal{L}^2(\Omega)} + \langle b'(u)(v) y, \eta\rangle_{\mathcal{L}^2(\Gamma_1)}$$

and

$$l_u(u;\eta)(v) = \langle f'(u)(v), \eta\rangle_{\mathcal{V}'\times\mathcal{V}}.$$

This theorem enables us to compute the first derivative using the concept of adjoint equations by setting $l(u; , \eta) = J_y(u, y(u))(\eta)$ in (17) of Theorem 11.

Theorem 13 *There exists a unique solution $p \in V$ of the adjoint equation*

$$a(u; \eta, p) = J_y(u, y(u))(\eta) \qquad \forall \eta \in V \tag{21}$$

and a unique solution operator $p : U_{ad} \to V$ with $p = p(u)$. Moreover, the cost function f is also Fréchet-differentiable and the action of $f'(u) \in L(U, \mathbb{R})$ on $v \in U$ is defined by

$$f'(u)(v) = J_u(u, y(u))(v) + l_u(u; p(u))(v) - a_u(u; y(u), p(u))(v).$$

For the second derivative we give the result for a Hessian applied to a vector.

Theorem 14 *The derivative $\hat{p} := \hat{p}^w := p'(u)(w)$ of the adjoint equation is defined by the solution of the variational problem*

$$\begin{aligned} a(u; \eta, \hat{p}) \quad = \quad & J_{yu}(u, y(u))(\eta)(w) + J_{yy}(u, y(u))(\eta)(y'(u)(w)) \\ & -a_u(u; \eta, p(u))(w) \quad \forall \eta \in V. \end{aligned}$$

The cost function f is also twice Fréchet-differentiable and $f''(u) \in L(U, L(U, \mathbb{R}))$ is defined by

$$\begin{aligned} & f''(u)(v)(w) \\ & \quad = J_{uu}(u, y(u), z(u))(v)(w) + J_{uy}(u, y(u), z(u))(v)\,(y'(u)(w)) \\ & \quad -a_u(u; y'(u)(w), p(u))(v) - a_u(u; y(u), p'(u)(w))(v) \\ & \quad -a_{uu}(u; y(u), p(u))(v)(w) \\ & \quad +l_u(u; p'(u)(w))(v) + l_{uu}(u; p(u))(v)(w) \end{aligned}$$

with

$$\begin{aligned} & a_{uu}(u; y, p)(v)(w) \\ & \quad = \textstyle\sum_{|i|,|j|\le 1} \langle a''_{ij}(u)(v)(w) D^i y, D^j p\rangle_{\mathcal{L}^2(\Omega)} + \langle b''(u)(v)(w) y, p\rangle_{\mathcal{L}^2(\Gamma_1)} \end{aligned}$$

and

$$l_{uu}(u; p)(v)(w) = \langle f''(u)(v)(w), p\rangle_{\mathcal{V}' \times \mathcal{V}}.$$

12. Numerical results

The numerical results were obtained for an example of an optimal shape design problem presented in Laumen (1996a). The minimization problem (16) is based on a discretization of the state space V and the control space U. Both spaces are discretized by linear spline functions on $\mathbb{R}^2$ and $\mathbb{R}^1$, respectively. The problem specific structure has been exploited comprehensively as outlined in Laumen (1996c).

it	time	$\|f'_N(u_i^M)\|$
0	8	$0.167E-01$
100	719	$0.120E-03$
500	3327	$0.187E-04$
1000	6350	$0.336E-05$
1500	9234	$0.626E-06$
2000	11990	$0.117E-06$
2500	14623	$0.220E-07$
2737	15826	$0.997E-08$

Table 1. Gradient method without line-search technique

it	time	$\|f'_N(u_i^M)\|$
0	8	$0.167E-01$
50	1147	$0.386E-04$
100	2205	$0.496E-05$
150	3216	$0.664E-06$
200	4168	$0.932E-07$
250	5072	$0.297E-07$
265	5333	$0.844E-08$

Table 2. Gradient method

it	time	$\|f'_N(u_i^M)\|$
0	8	$0.167E-01$
1	395	$0.245E-03$
2	654	$0.620E-05$
3	840	$0.865E-06$
4	996	$0.202E-06$
5	1131	$0.599E-07$
6	1251	$0.230E-07$
7	1347	$0.102E-07$
8	1442	$0.513E-08$

Table 3. Newton's method

In Table 12.1 the iteration for a gradient method without line search is presented. It shows an enormous number of iterations until the convergence criterion is satisfied.

The numerical behavior of the gradient method can be improved vastly if a line search is incorporated. This is illustrated in Table 12.2.

Newton method clearly outperforms the gradient method in terms of iterations. However, note that also with regard to timing Newton method is faster by a factor of 3 than the gradient method with line search.

Among the quasi–Newton methods tested the SR–1 update showed the best performance. In Table 11.4 it shows clearly that although the number of iterates is higher than for Newton method it has a much lower computing time. Comparing Table 12.4 with 12.1 it shows a dramatic improvement although in this algorithm only first order information, i. e. the computation of gradients, is required.

it	time	$\|f'_N(u_i^M)\|$
0	8	$0.167E-01$
5	46	$0.407E-03$
10	85	$0.105E-03$
15	121	$0.531E-04$
20	156	$0.113E-04$
25	191	$0.742E-05$
30	224	$0.975E-06$
35	256	$0.116E-06$
40	288	$0.769E-07$
45	319	$0.155E-07$
46	325	$0.923E-08$

Table 4. Quasi-Newton method with SR1-update

References

ARMIJO, L. (1966) Minimization of functions having Lipschitz-continuous first partial derivatives. *Pacific J. Math.*, **16**, 1–3.

BANKS, H.T. and KOJIMA, F. (1988) *Boundary shape identification problems in two-dimensional domains related to thermal testing of materials.* Technical Report 181654, NASA Contractor Report, NASA, Langley Research Center.

BANKS, H.T. and KOJIMA, F. (1989) Boundary shape identification problems in two-dimensional domains related to thermal testing of materials. *Quarterly of Applied Math.*, **XLVII**, 273–293.

BROWN, P.N. and SAAD, Y. (1994) Convergence theory of nonlinear Newton–Krylov algorithms. *SIAM J. Optimization*, **4**, 2, 297–330.

DELFOUR, M., PAYRE, G. and ZOLÉSIO, J.-P. (1983) Optimal design of a minimum weight thermal diffuser with constraint on the output thermal power flux. *Appl. Math. Optim.*, **9**, 225–262.

DELFOUR, M., PAYRE, G. and ZOLÉSIO, J.-P. (1986) Optimal parametrized design of thermal diffusers for communication satellites. *Optim. Control Appl. & Methods.*, **7**, 163–184.

DEMBO, R.S., EISENSTAT, S.C. and STEIHAUG, T. (1982) Inexact Newton methods. *SIAM J. Numer. Anal.*, **19**, 400–408.

DENNIS, J.E. and SCHNABEL, R.B. (1983) *Numerical Methods for Unconstrained Optimization and Nonlinear Equations.* Prentice-Hall, Englewood Cliffs, N.J.

EISENSTAT, S.C. and WALKER, H.F. (1994) Globally convergent inexact Newton methods. *SIAM J. Optimization*, **4**, 2, 393–422.

GRUVER, W.A. and SACHS, E. (1980) *Algorithmic Methods In Optimal Control.* Pitman, London.

HASLINGER, J. and NEITTAANMÄKI, P. (1988) *Finite Element Approximation for Optimal Shape Design: Theory and Applications.* John Wiley & Sons, New York.

HUSCHENS, J. (1994) On the use of product structure in secant methods for nonlinear least squares problems. *SIAM J. of Optimization*, 108–129.

KELLEY, C.T. (1995) *Iterative Methods for Linear and Nonlinear Equations.* SIAM, Philadelphia.

KUNISCH, K. and SACHS, E.W. (1992) Reduced SQP methods for parameter identification problems. *SIAM J. Numer. Anal.*, **29**, 1793–1820.

KUPFER, F.-S. (1995) An infinite dimensional convergence theory for reduced SQP methods in Hilbert space. *SIAM J. Optimization*, **6**, 126–163.

KUPFER, F.-S. and SACHS, E.W. (1990) A prospective look at SQP methods for semilinear parabolic control problems. In: K.-H. Hoffmann and W. Krabs, eds., *Optimal Control of Partial Differential Equations, Irsee 1990*, Lect. Notes in Control and Inform. Sciences, **149**, 143–157. Springer.

LAUMEN, M. (1996A) *A comparison of numerical methods for optimal shape* *design problems.* Technical report, Universität Trier, Forschungsbericht.

LAUMEN, M. (1996B) *A modified mesh independence principle of Newton's method.* Technical report, Universität Trier, Forschungsbericht.

LAUMEN, M. (1996C) *Newton's method for a class of optimal shape design problems.* Technical report, Universität Trier, Forschungsbericht.

LAUMEN, M. (1996D) *Numerical methods for optimal shape design problems.* PhD thesis, Fb IV - Mathematik, Universität Trier, 54286 Trier, Germany.

ORTEGA, J.M. and RHEINBOLDT,W.C. (1970) *Iterative Solution of Nonlinear Equations in Several Variables.* Academic Press, New York.

PIRONNEAU, O. (1984) *Optimal Shape Design for Elliptic Systems.* Springer-Verlag, New York, Berlin, Heidelberg, Tokyo.

PIRONNEAU, O. and VOSSINIS, A. (1991) *Comparison of some optimization algorithms for optimum shape design in aerodynamics.* Programme 6: Calcul scientifique, Modélisation et Logiciels numériques 1392, INRIA.

SACHS, E.W. (1994) Control applications of reduced SQP methods. In R. Bulirsch and D. Kraft, eds., *Computational Optimal Control, Int. Series Num. Math.*, **115**, 89–104. Birkhäuser.

SOKOLOWSKI, J. and ZOLESIO, J.P. (1987) Shape design sensitivity analysis of plates and plane elastic solids under unilateral constraints. *J. of Optim. Theory and Appl.*, **54**, 361–381.

SOKOLOWŠKI, J. and ZOLESIO, J.P. (1992) *Introduction to Shape Optimization: Shape Sensitivity Analysis.* Springer-Verlag, Berlin-Heidelberg.