

# NONLINEAR PROGRAMMING: ALGORITHMS, SOFTWARE, AND APPLICATIONS

*From Small to Very Large Scale Optimization*

Klaus Schittkowski

*University of Bayreuth*

*Department of Computer Science*

*D-95440 Bayreuth*

klaus.schittkowski@uni-bayreuth.de

Christian Zillober

*University of Bayreuth*

*Department of Computer Science*

*D-95440 Bayreuth*

christian.zillober@uni-bayreuth.de

**Abstract** We introduce some methods for constrained nonlinear programming that are widely used in practice and that are known under the names SQP for sequential quadratic programming and SCP for sequential convex programming. In both cases, convex subproblems are formulated, in the first case a quadratic programming problem, in the second case a separable nonlinear program in inverse variables. The methods are outlined in a uniform way and the results of some comparative performance tests are listed. We especially show the suitability of sequential convex programming methods to solve some classes of very large scale nonlinear programs, where implicitly defined systems of equations seem to support the usage of inverse approximations. The areas of interest are structural mechanical optimization, i.e., topology optimization, and optimal control of partial differential equations after a full discretization. In addition, a few industrial applications and case studies are shown to illustrate practical situations under which the codes implemented by the authors are in use.

**Keywords:** sequential quadratic programming, SQP, sequential convex programming, SCP, comparative tests, industrial applications, topology optimization, optimal control of semilinear elliptic equations

## Introduction

Over the last years, mathematical optimization became a powerful tool in many *real-life* application areas. Proceeding from a formal mathematical model simulating the behavior of a practical system, optimization algorithms are applied to minimize a so-called cost function subject to some constraints. A typical example is the minimization of the weight of a mechanical structure under given loads and under constraints for admissible stresses, displacements, or dynamic responses. Highly complex industrial and academic design problems can be solved today by means of nonlinear programming algorithms without any chance to get equally qualified results by traditional empirical approaches.

We consider the smooth, constrained optimization problem to minimize a scalar objective function  $f(x)$  under nonlinear inequality constraints,

$$x \in \mathbb{R}^n : \begin{array}{l} \min f(x) \\ g(x) \leq 0 \end{array} , \quad (1)$$

where  $x$  is an  $n$ -dimensional parameter vector. The vector-valued function  $g(x)$  defines  $m$  inequality constraints,  $g(x) = (g_1(x), \dots, g_m(x))^T$ . To simplify the notation, equality constraints and upper or lower bounds of variables are omitted. We assume that the feasible domain of (1) is non-empty and bounded, and that the functions  $f(x)$  and  $g(x)$  are smooth, i.e., twice continuously differentiable.

Since we suppose that problem (1) is nonconvex and nonlinear in general, the basic idea is to replace it by a sequence of *simpler* problems. *Simpler* means in this case that the structure of the subproblem is much easier to analyze and that the subproblem is uniquely solvable by an available *black box* technique. In particular, it is assumed that the applied numerical algorithm does not require any additional function or gradient evaluations of the original functions  $f(x)$  and  $g(x)$ . An essential requirement is that we get a first order approximation of (1), i.e., that function and gradient values of the original problem and the subproblem coincide at a current iterate.

Sequential quadratic programming (SQP) methods are very well known and are considered as the standard general purpose algorithm for solving smooth nonlinear optimization problems at least under the following assumptions:

- The problem is not too big.
- Function and especially gradient values can be evaluated within sufficient precision.

- The problem is smooth and well-scaled.

In this case, the subproblems consist of strictly convex quadratic programming problems with inequality constraints obtained by linearizing the constraints and by approximating the Lagrangian function of (1) quadratically. SQP methods have their roots in unconstrained optimization, and can be considered as extensions of quasi-Newton methods taking constraints into account. The basic idea is to establish a quadratic approximation based on second order information with the goal to achieve a fast local convergence speed. Second order information about the Hessian of the Lagrangian is updated by a positive definite quasi-Newton matrix. The linearly constrained, strictly convex quadratic program must be solved in each iteration step by an available *black box* solver.

Despite of the success of SQP methods, another class of efficient optimization algorithms was proposed by engineers, where the motivation is found in mechanical structural optimization. The method is based on the observation that in some special cases, typical structural constraints become linear in the inverse variables. Although this special situation is rarely observed in practice, a suitable substitution of structural variables by inverse ones depending on the sign of the corresponding partial derivatives and subsequent linearization is expected to linearize constraints somehow.

More general convex approximations are introduced known under the name *moving asymptotes* (MMA). The goal is to construct convex and separable subproblems, for which efficient solvers are available based on interior point techniques. Thus, we denote this class of methods by SCP, an abbreviation for *sequential convex programming*. In other words, SQP methods are based on local second order approximations, whereas SCP methods are applying global first order convex approximations.

SCP methods can be used to solve very large scale optimization or VLSO problems, respectively, without any further adoptions of the fundamental structure. Typically they are applied to solve topology optimization problems in mechanical engineering. It seems that these methods are particularly efficient in situations where objective function and constraints depend also on *state variables*, i.e., variables defined implicitly by an internal system of equations.

The subsequent section contains a brief outline of SQP and SCP methods to illustrate their basic structure. More details are found in the references cited. The results of some comparative numerical tests of both approaches are shown for a set of 79 structural optimization problems and for a set of 306 standard academic test problems. Section 3 shows how SCP methods can be applied to solve VLSO problems. The

two classes of test problems under consideration are from topology optimization and optimal control of semilinear elliptic partial differential equations. Although the presented examples proceed from relatively simple *academic* formulations, they show at least the capability of SCP methods for solving large optimization problems without assuming any special sparsity patterns.

To show that SQP and SCP methods are of practical interest and are routinely used in industry for design and production, we outline a few case studies in Section 4. Especially, we briefly introduce applications how to find the

- optimal design of horn radiators for satellite communication (As-trium),
- optimal design of surface acoustic wave filters (Epcos),
- on-line control of a tubular reactor (BASF),
- weight-reduction of a cruise ship (Meyer Werft).

We want to give an impression of the numerical complexity of the optimization models and the importance and efficiency of optimization codes in these cases.

## 1. Sequential Quadratic Versus Sequential Convex Programming Methods

### 1.1 A General Framework

Since we assume that our optimization problem (1) is nonconvex and nonlinear in general, the basic idea is to replace (1) by a sequence of *simpler* problems. Starting from an initial design vector  $x_0 \in \mathbb{R}^n$  and an initial multiplier estimate  $u_0 \in \mathbb{R}^m$ , iterates  $x_k \in \mathbb{R}^n$  and  $u_k \in \mathbb{R}^m$  are computed successively by solving subproblems of the form

$$y \in \mathbb{R}^n : \begin{array}{l} \min f^k(y) \\ g^k(y) \leq 0 \end{array} . \quad (2)$$

Let  $y_k$  be the optimal solution and  $v_k$  the corresponding Lagrangian multiplier of (2). A new iterate is computed by

$$\begin{aligned} x_{k+1} &= x_k + \alpha_k(y_k - x_k) , \\ u_{k+1} &= u_k + \alpha_k(v_k - u_k) , \end{aligned} \quad (3)$$

where  $\alpha_k$  is a steplength parameter discussed subsequently.

*Simpler* means that the subproblem has a specific mathematical structure which is easier to analyze, and that it is efficiently solvable by an available *black box* technique, more or less independently of the underlying model structure. In particular, it is assumed that the numerical algorithm for solving (2) does not require any additional function or gradient evaluations of the original functions  $f(x)$  and  $g_j(x)$ ,  $j = 1, \dots, m$ . Note that we are looking for a simultaneous approximation of an optimal solution  $x^*$  and of the corresponding multiplier vector  $u^*$ .

Now we summarize the requirements to describe SQP and SCP algorithms in a uniform way:

- 1 (2) is strictly convex and smooth, i.e., functions  $f^k(x)$  and  $g_j^k(x)$  are twice continuously differentiable,  $j = 1, \dots, m$ .
- 2 (2) is a first order approximation of (1) at  $x_k$ , i.e.,  $f(x_k) = f^k(x_k)$ ,  $\nabla f(x_k) = \nabla f^k(x_k)$ ,  $g(x_k) = g^k(x_k)$ , and  $\nabla g(x_k) = \nabla g^k(x_k)$ .
- 3 The search direction  $(y_k - x_k, v_k - u_k)$  is a descent direction for an augmented Lagrangian merit function introduced below.
- 4 The feasible domain of (2) is non-empty and bounded.

Strict convexity of (2) means that the objective function  $f^k(x)$  is strictly convex and that the constraints  $g_j^k(x)$  are convex functions for all iterates  $x_k$  and  $j = 1, \dots, m$ . Since the feasible domain is supposed to be non-empty, (2) has a unique solution  $y_k \in \mathbb{R}^n$  with Lagrangian multiplier  $v_k \in \mathbb{R}^m$ . A further important consequence is that if  $y_k = x_k$ , then  $x_k$  and  $v_k$  solve the general nonlinear programming problem (1) in the sense of a stationary solution.

A line search is introduced to stabilize the solution process, particularly helpful when starting from a bad initial guess. We are looking for an  $\alpha_k$ , see (3),  $0 < \alpha_k \leq 1$ , so that a step along a merit function  $\Psi_k(\alpha)$  from the current iterate to the new one becomes *acceptable*. The idea is to penalize the Lagrange function in the  $L_2$  norm as soon as constraints are violated, by defining

$$\Phi_r(x, u) = f(x) + \sum_{j \in J} \left( u_j g_j(x) + \frac{1}{2} r_j g_j(x)^2 \right) - \frac{1}{2} \sum_{j \in K} u_j^2 / r_j \quad (4)$$

Then we set

$$\Psi_k(\alpha) = \Phi_{r_k} \left( \begin{pmatrix} x_k \\ u_k \end{pmatrix} + \alpha \begin{pmatrix} y_k - x_k \\ v_k - u_k \end{pmatrix} \right), \quad (5)$$

where  $J = \{j : g_j(x) \geq -u_j/r_j\}$  and  $K = \{1, \dots, m\} \setminus J$  define the constraints considered as active or inactive, respectively.

The steplength parameter  $\alpha_k$  is required in (3) to enforce global convergence of the optimization method, i.e., the approximation of a point satisfying the necessary Karush-Kuhn-Tucker optimality conditions when starting from arbitrary initial values, e.g., a user-provided  $x_0 \in \mathbb{R}^n$  and  $u_0 = 0$ . The merit function defined by (4) is also called augmented Lagrange function, see for example Rockafellar [31]. The corresponding penalty parameter  $r_k$  at the  $k$ -th iterate that controls the degree of constraint violation, must be chosen carefully to guarantee a descent direction of the merit function, so that the line search is well-defined,

$$\Psi'_k(0) = \nabla \Phi_{r_k}(x_k, u_k)^T \begin{pmatrix} y_k - x_k \\ v_k - u_k \end{pmatrix} < 0 . \quad (6)$$

The line search consists of a successive reduction of  $\alpha$  starting at 1, usually combined with a quadratic interpolation, until a sufficient decrease condition is obtained.

Standard techniques to approximate constraints, for example successive linearization, can lead to inconsistent constraints in (2). In these cases, it is possible to introduce an additional variable and to modify objective function and constraints, for example by

$$\begin{aligned} \min f^k(y) + \rho_k y_{n+1}^2 \\ y \in \mathbb{R}^{n+1} : \quad g^k(y) - y_{n+1} \leq 0 \quad , \\ -y_{n+1} \leq 0 \end{aligned} \quad (7)$$

in the simplest form. The penalty term  $\rho_k$  is added to the objective function to reduce the influence of the additional variable  $y_{n+1}$  as much as possible. The index  $k$  implies that this parameter also needs to be updated during the algorithm. It is obvious that (7) always possesses a feasible solution.

## 1.2 Sequential Quadratic Programming

Sequential quadratic programming or SQP methods belong to the most powerful nonlinear programming algorithms we know today for solving differentiable nonlinear programming problems of the form (1). The theoretical background is described in Stoer [44] in form of a review, and in Spellucci [43] in form of an extensive text book. From the more practical point of view, SQP methods are also introduced in the books of Papalambros and Wilde [28] or Edgar and Himmelblau [9]. Their excellent numerical performance is tested and compared with other methods in Schittkowski [33–34] and Hock, Schittkowski [16]. Since many years they belong to the most frequently used algorithms to solve practical optimization problems.

The basic idea is to formulate and solve a quadratic programming subproblem in each iteration, which is obtained by linearizing the constraints and approximating the Lagrange function of (1) quadratically. To formulate the subproblem, we proceed from given iterates  $x_k \in \mathbb{R}^n$ , an approximation of the solution,  $u_k \in \mathbb{R}^m$ , an approximation of the vector of multipliers, and  $B_k \in \mathbb{R}^{n \times n}$ , an approximation of the Hessian of the Lagrange function. Then we obtain subproblem (2) by defining

$$\begin{aligned} f^k(y) &= \frac{1}{2} (y - x_k)^T B_k (y - x_k) + \nabla f(x_k)^T (y - x_k) + f(x_k) , \\ g_j^k(y) &= \nabla g_j(x_k)^T (y - x_k) + g_j(x_k) , \quad j = 1, \dots, m . \end{aligned} \quad (8)$$

It is immediately seen that the requirements of the previous section for (2) are satisfied. The key idea is to approximate also second order information to get a fast final convergence speed. The update of the matrix  $B_k$  can be performed by standard quasi-Newton techniques known from unconstrained optimization subject to the Lagrangian function of the nonlinear program. In most cases, the BFGS-method is applied, see Powell [30, 29], or Stoer [44], starting from the identity matrix or any other positive definite matrix  $B_0$ . A simple modification as for example proposed by Powell [30] guarantees positive definite matrices.

Among the most attractive features of sequential quadratic programming methods is the superlinear convergence speed in the neighborhood of a solution, i.e.,

$$\|x_{k+1} - x^*\| < \gamma_k \|x_k - x^*\| \quad (9)$$

with  $\gamma_k \rightarrow 0$ .

The motivation for the fast convergence speed of SQP methods is based on the following observation: an SQP method is identical to Newton's method to solve the necessary optimality conditions of (1), if  $B_k$  is the Hessian of the Lagrange function at  $x_k$  and  $u_k$  and if we start sufficiently close to a solution. The statement is easily derived in case of equality constraints only, but holds also for inequality restrictions.

There remain a few comments to summarize some interesting features of SQP methods:

- Linear constraints and bounds of variables remain satisfied.
- In case of  $n$  active constraints, the SQP method behaves like Newton's method for solving the corresponding system of equations, i.e., the local convergence speed is even quadratically.
- The algorithm is globally convergent and the local convergence speed is superlinear.

- A simple reformulation allows the efficient solution of constrained nonlinear least squares problems, see Schittkowski [37, 40].
- A large number of constraints can be treated by an active set strategy, see Schittkowski [38]. In particular, the computation of gradients for inactive restrictions can be omitted.
- There exists a large variety of different extensions to solve also large scale problems, see Gould and Toint [13] for a review.

### 1.3 Sequential Convex Programming

Sequential convex programming methods are developed mainly for mechanical structural optimization. The first approach of Fleury and Braibant [11] and Fleury [10] is known under the name convex linearization (CONLIN) and exploits the observation that in some special cases, typical structural constraints become linear in the inverse variables. Although this special situation is always found in case of statically determinate structures, it is rarely observed in practice. However, a suitable substitution by inverse variables depending on the sign of the corresponding partial derivatives and subsequent linearization is expected to linearize constraints somehow.

For the CONLIN method, Nguyen et al. [26] gave a convergence proof but only for the case that (1) consists of a concave objective function and constraints which is of minor practical interest. They showed also that a generalization to non-concave constraints is not possible. More general convex approximations are introduced by Svanberg [45] known under the name *method of moving asymptotes* (MMA). The goal is always to construct nonlinear convex and separable subproblems, for which efficient solvers are available. Using the flexibility of the asymptotes which influence the curvature of the approximations, it is possible to avoid the concavity assumption.

Given an iterate  $x_k$ , the basic idea is to linearize  $f$  and  $g_j$  at  $x_k$  subject to transformed variables  $(U_i^k - x_i)^{-1}$  and  $(x_i - L_i^k)^{-1}$  depending on the sign of the corresponding first partial derivative.  $U_i^k$  and  $L_i^k$  are reasonable bounds and are adapted by the algorithm after each successful step. Also several other transformations have been developed in the past.

By defining suitable index sets

$$I_k^+ = \{i : \frac{\partial}{\partial x_i} f(x_k) \geq 0\} \quad , \quad I_k^- = \{i : \frac{\partial}{\partial x_i} f(x_k) < 0\}$$

for objective function and, in a similar way,  $I_{jk}^+$  and  $I_{jk}^-$  for constraints, we get the corresponding approximating functions of subproblem (2) by

$$\begin{aligned} f^k(y) &= \alpha_0^k + \sum_{i \in I_k^+} \frac{\beta_{i,0}^k}{U_i^k - y_i} - \sum_{i \in I_k^-} \frac{\beta_{i,0}^k}{y_i - L_i^k}, \\ g_j^k(y) &= \alpha_j^k + \sum_{i \in I_{jk}^+} \frac{\beta_{i,j}^k}{U_i^k - y_i} - \sum_{i \in I_{jk}^-} \frac{\beta_{i,j}^k}{y_i - L_i^k}, \end{aligned} \tag{10}$$

$j = 1, \dots, m$ , where  $y = (y_1, \dots, y_n)^T$ . The coefficients  $\alpha_j^k$  and  $\beta_{i,j}^k$ ,  $j = 0, \dots, m$  are chosen to satisfy the requirements of Section 2.1, i.e., that (2) is convex and a first order approximation of (1) at  $x_k$ . By an appropriate regularization of the objective function, strict convexity of  $f^k(y)$  is guaranteed, see Zillober [50]. As shown there, the search direction  $(y_k - x_k, v_k - u_k)$  is a descent direction for the augmented Lagrangian merit function (4,5), see also (6). The approximation scheme (10) can be applied only to inequality constraints. Additional equality constraints can be added, but are linearized as for SQP methods.

The choice of the asymptotes  $L_i^k$  and  $U_i^k$ , is crucial for the computational behavior of the method, in particular since additional lower and upper bounds are usually available. Additional safeguards ensure the compatibility of this procedure with the overall scheme and guarantee global convergence. A small positive constant is introduced to avoid that the difference between the asymptotes and the current iteration point becomes too small. However, these safeguards are rarely used in practice, see Zillober [50] for more details.

For the first SCP codes developed, the convex and separable subproblems are solved by a dual approach, where dense linear systems of equations with  $m$  rows and columns are solved, cf. Svanberg [45] or Fleury [10]. Recently, a predictor-corrector interior point method for the solution of the subproblems was proposed by Zillober [49]. The advantage is to formulate either  $n \times n$  or  $m \times m$  linear systems of equations leading to a more flexible treatment of large problems. The resulting algorithm is very efficient especially for large scale mechanical engineering problems, and given sparsity patterns of the original problem data can be exploited.

To summarize, the most important features of SCP methods are:

- Linear equality constraints and bounds of variables remain satisfied.
- The algorithm is globally convergent.

- As for SQP methods, a large number of constraints can be treated by an active set strategy, see Zillober [52, 51]. In particular, the computation of gradients for inactive restrictions can be omitted.
- Large scale problems can be handled by different variants of the solution procedure for the subproblem, see Zillober et al. [53], where sparsity of problem data can be exploited.

## 1.4 Comparative Results

Our numerical tests use all 306 academic and real-life test problems published in Hock and Schittkowski [16] and in Schittkowski [36]. Part of them are also available in the CUTE library, see Bongartz et al. [5]. The test problems possess also nonlinear equality constraints and additional lower and upper bounds for the variables. The two codes under consideration, NLPQLP of Schittkowski [39] and SCIP of Zillober [51], are able to solve more general problems

$$\begin{aligned}
 & \min f(x) \\
 x \in \mathbb{R}^n : & \quad \begin{aligned} & h(x) = 0 \quad , \\ & g(x) \leq 0 \quad , \\ & x_l \leq x \leq x_u \quad , \end{aligned}
 \end{aligned} \tag{11}$$

with additional smooth functions  $h(x) = (h_1, \dots, h_{m_e})^T$  for equality constraints and bounds  $x_l < x_u$ .

Since analytical derivatives are not available for all problems, we approximate them numerically by a five-point difference formula. The test examples are provided with exact solutions, either known from analytical evaluation or from the best numerical data found so far. Since the calculation times are very short, we count only function and gradient evaluations. This is a realistic assumption, since for the practical applications in mind calculation times for evaluating model functions dominate and the numerical efforts within an optimization code are negligible.

The result of a test run is considered as a successful return, if the relative error in the objective function is less than a given tolerance  $\epsilon$  and if the maximum constraint violation is less than  $\epsilon^2$ . We take into account that a code returns a solution with a better function value than the known one subject to the error tolerance of the allowed constraint violation. However, there is still the possibility that an algorithm terminates at a local solution different from the one known in advance. Thus, we call a test run a successful one, if the internal termination conditions are satisfied subject to a reasonably small tolerance, if the obtained solution is feasible, and if the objective function value is significantly larger

<i>code</i>	<i>p<sub>succ</sub></i>	<i>n<sub>f</sub></i>	<i>n<sub>it</sub></i>
<i>NLPQLP</i>	100 %	33	21
<i>SCPIP</i>	93 %	74	42

Table 1. Performance Results for Standard Test Problems

than the known one. For our numerical tests, we use  $\epsilon = 0.01$ , i.e., we require a final accuracy of one per cent, see Zillober et al. [53] for more details.

The code NLPQLP of Schittkowski [39] represents the most recent version of NLPQL which is frequently used in academic and commercial institutions, see Schittkowski [35]. Functions and gradients must be provided by reverse communication and the quadratic programming subproblems are solved by the primal-dual method of Goldfarb and Idnani [12] based on numerically stable orthogonal decompositions. The SQP algorithm is executed with termination accuracy  $10^{-8}$  and the maximum number of iterations is 500. In the SCP implementation SCPIP of Zillober [52, 51], the convex subproblems are solved by the predictor-corrector interior point method described in Zillober [49]. Input variables and tolerances are chosen in a way such that the termination conditions for SCPIP and NLPQLP are comparable.

Table 1 shows the percentage of successful test runs,  $p_{succ}$ , the average number of function calls,  $n_f$ , and the average number of iterations,  $n_{it}$ . Function evaluations needed for gradient approximations, are not counted for  $n_f$ . Their average number is  $4 \times n_f$ . Many test problems are unconstrained or possess a highly nonlinear objective function preventing SCP from converging as fast as SQP methods. Moreover, bounds are often set far away from the optimal solution, leading to initial asymptotes too far away from the region of interest. Since SCP methods do not possess fast local convergence properties, SCPIP needs about twice as many iterations and function evaluations.

The situation is different in mechanical structural optimization, where the SCP methods have been invented. In the numerical study of Schittkowski et al. [41], 79 finite element formulations of academic and practical problems are collected based on the simulation package MBB-LAGRANGE, see Knepe et al. [19]. The maximum number of variables is 144 and a maximum number of constraints 1020 without box-constraints. NLPQL, see Schittkowski [35], and MMA, former versions of NLPQLP and SCPIP, respectively, are among the 11 optimization algorithms under consideration. To give an impression on the behavior of SQP versus MMA, we repeat some results of Schittkowski et al. [41], see Table 2.

One of the main difficulties of a comparative performance study is that the optimization program solve only a certain subset of test problems successfully, which differs from code to code. Thus, mean values of a performance criterion are evaluated pairwise over the set of successfully solved test problems of two algorithms, and then compared in form of a matrix, see the priority theory of Saaty [32] and also Lootsma [21]. The decision whether the result of a test run is considered as a successful one or not, depends on a tolerance  $\epsilon$  which is set to  $\epsilon = 0.01$  and  $\epsilon = 0.00001$ , respectively.

<i>code</i>	$\epsilon = 0.01$			$\epsilon = 0.00001$		
	<i>p<sub>succ</sub></i>	<i>n<sub>f</sub></i>	<i>n<sub>it</sub></i>	<i>p<sub>succ</sub></i>	<i>n<sub>f</sub></i>	<i>n<sub>it</sub></i>
<i>NLPQL</i>	84 %	2.0	1.6	77 %	1.3	1.3
<i>MMA</i>	73 %	1.0	1.0	73 %	1.0	1.0

Table 2. Performance Results for Structural Optimization Test Problems

The figures of Table 2 represent the scaled relative performance data when comparing the codes among each other. We conclude for example that for  $\epsilon = 0.01$ , NLPQL requires about twice as many gradient evaluations or iterations, respectively, as MMA. When requiring a higher termination accuracy, however, NLPQL needs only 30 % more gradient calls. On the other hand, NLPQL is a bit more reliable than MMA.

## 2. Very Large Scale Optimization by Sequential Convex Programming

### 2.1 Topology Optimization

To give an impression about the capabilities of an SCP implementation for solving very large scale nonlinear programming problems, we consider now structural mechanical optimization, more precisely topology optimization. Given a predefined domain in the 2D/3D space with boundary conditions and external load, the intention is to distribute a percentage of the initial mass on the given domain such that a global measure takes a minimum, see Bendsøe [2] or Bendsøe and Sigmund [3] for a broader introduction. Assuming isotropic material, the so-called power law approach, see also Bendsøe [1] or Mlejnek [25], leads to a

nonlinear program of the form

$$\begin{aligned}
 & \min u^T p \\
 x \in \mathbb{R}^n, u \in \mathbb{R}^d : & \quad V(x) \leq aV_0 \ , \\
 & \quad K(x)u = p \ , \\
 & \quad 0 < x_l \leq x \leq 1 \ ,
 \end{aligned} \tag{12}$$

where  $x = (x_1, \dots, x_n)^T$  denotes the relative material densities, artificially introduced variables. In the final solution, we consider a small value of  $x_i$  as zero or no mass, a larger value as one or full mass. Theoretically, one is only interested in 0-1 solutions, which are not guaranteed by the continuous approach applied.  $u = (u_1, \dots, u_d)^T$  is the displacement vector computed from the linear system of equations  $K(x)u = p$  with a positive definite stiffness matrix  $K(x)$  and an external load vector  $p$ .  $d$  denotes the number of degrees of freedom of the structure. We assume without loss of generality that there is only one load case. The goal is to minimize the so-called compliance or, in other words, to make the structure as stiff as possible.

It is essential to understand that the system of linear equations  $K(x)u = p$  can be considered as the state equations of our optimization problem. In practical situations, finite element simulation software is available to set up the stiffness matrix and to solve the system  $K(x)u = p$  internally. To indicate that  $u$  depends on the relative densities  $x$ , we use the notation  $u(x)$ .

The relative densities and the elementary stiffness matrices  $K_i$  define  $K(x)$  by

$$K(x) = \sum_{i=1}^n x_i^q K_i \ .$$

$V(x)$  is the volume of the structure, usually a linear function of the design variables,

$$V(x) = \sum_{i=1}^n x_i V_i \ ,$$

where  $V_i$  is the volume of the  $i$ -th finite element.  $V_0$  is the available volume,  $V_0 = \sum_{i=1}^n V_i$ , and  $a$  with  $0 < a < 1$  the given fraction of the full volume to distribute the available mass.  $x_l$  is a vector of small positive numbers for avoiding singularities. The nonlinearity  $x_i^q$  in the state equation is found heuristically and usually applied in practice with  $q = 3$ . Its role is to penalize intermediate values between the lower bound and 1.

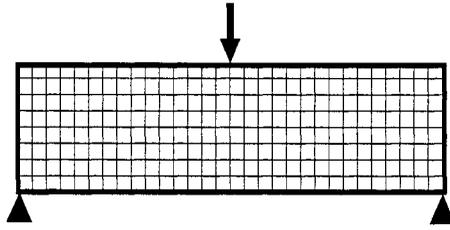


Figure 1. Design Region of Beam

The partial derivatives of the objective function of problem (12) are computed from

$$\frac{\partial}{\partial x_j} \left( u(x)^T p \right) = -q x_j^{q-1} u(x)^T K_j u(x) \quad (13)$$

for  $j = 1, \dots, n$ , see Zillober et al. [53]. Since the elementary stiffness matrices  $K_j$  are very sparse, for example containing only non-zero entries on an  $8 \times 8$ -submatrix in case of the rectangular elements used in this section, the  $j$ -th partial derivative is computed very efficiently as soon as the displacement vector  $u(x)$  is available.

The solution of topology optimization problems easily leads to very large scale, highly nonlinear programs. The probably most simple example is a beam, which is loaded in the middle and supported at the two lower vertices. The design region is a rectangular plate, see Figure 1, discretized by rectangular finite elements. For symmetry reasons, we consider only one half of the beam for our calculations. The number of horizontal grid lines is denoted by  $n_x$ , the number of vertical grid lines by  $n_y$ . The solution of topology optimization problems as outlined so far, produces a strange phenomenon, a checkerboard-type material distribution in certain regions. Thus, an additional filter is applied by which partial derivatives are modified by certain weights depending on a given radius  $r_f$  around the considered element, see for example Bendsøe and Sigmund [3]. In addition, Table 3 shows the total number of optimization variables,  $n$ , the number of iterations,  $n_{it}$ , the final objective function value,  $f(x)$ , and the gradient norm of Lagrangian function,  $\|\nabla_x L(x, u)\|$ . The results show that that SCP methods are able to solve dense nonlinear programs with more than  $10^6$  variables. More detailed computational results are presented in Zillober et al. [53].

$n_x$	$n_y$	$n$	$n_{it}$	$f(x)$	$\ \nabla_x L(x, u)\ $	$r_f$
600	400	240,000	22	52.63	1.3E-3	8
600	400	240,000	26	54.25	6.5E-4	0
1,050	700	735,000	38	54.39	4.6E-4	10
1,260	840	1,058,400	43	56.55	1.3E-5	0

Table 3. Numerical Results of SCPIP for the Half Beam

## 2.2 Optimal Control of Semilinear Elliptic Partial Differential Equations

The intention behind the numerical tests of this section is to show that SCP methods can be applied also to optimization problems which are completely different from the original mechanical engineering applications. We consider a series of test problems investigated by Maurer and Mittelmann [22–23] when studying necessary optimality conditions for optimal control of elliptic partial differential equations with state and control constraints. They differ by the type of control, boundary and interior control, the cost functional, the non-homogeneous part of the elliptic equation, and the boundary conditions. Results for a typical test problem are shown below. More detailed numerical results are presented in Zillober et al. [53], where the performance of the code SCPIP is compared with the best codes of the Maurer and Mittelmann study.

Proceeding from the two-dimensional unit square  $\Omega$  and the boundary  $\Gamma$ , the optimal control problem is defined by

$$\begin{aligned}
 & \min \frac{1}{2} \int_{\Omega} (y(x) - \sin(2\pi x_1) \sin(2\pi x_2))^2 + u(x)^2 \, dx \\
 & \begin{matrix} u \in L^\infty(\Omega), \\ y \in C^2(\Omega) \end{matrix} : \begin{matrix} \Delta y + e^y = 0, & x \in \Omega, \\ \partial_\nu y + y = 0, & x \in \Gamma, \\ y \leq 0.371, & -8 \leq u \leq 9. \end{matrix}
 \end{aligned} \tag{14}$$

$u$  is the control function we want to compute subject to constant lower and upper bounds, and  $y$  denotes the state variable, i.e., the solution of the semilinear elliptic partial differential equation  $\Delta y + e^y = 0$  subject to a Neumann boundary condition of the form  $\partial_\nu y + y = 0$ .  $\partial_\nu y$  denotes the outward unit normal along the boundary  $\Gamma$ . The solution of the state equations depends on the control function  $u$ , and a state constraint for  $y$  is given in form of an upper constant bound. The cost function is of tracking type, see Ito and Kunisch [17] depending on the spatial variable  $x$ .

$N + 1$	$n$	$m$	$n_{it}$	$f(x)$
100	19,998	10,197	20	0.0528
200	79,998	40,397	22	0.0530
300	179,998	90,597	16	0.0535
400	319,998	160,797	18	0.0534
500	499,998	250,997	16	0.0536
600	719,998	361,197	16	0.0537

Table 4. Numerical Results for a Semilinear Elliptic Control Problem

The elliptic control problem (14) is pointwise discretized subject to the control and state variables as proposed by Maurer and Mittelmann [22–23] based on a uniform grid of size  $N$  for discretizing  $\Omega$  and a five-star-formula for the Laplace operator. Thus, we get a set of  $N^2$  equality constraints

$$4y_{ij} - y_{i,j-1} - y_{i-1,j} - y_{i,j+1} - y_{i+1,j} + h^2 e^{y_{ij}} = 0 . \quad (15)$$

First derivatives in Neumann boundary conditions are approximated by forward or backward differences, respectively.

It is important to understand that SCP methods are not invented to solve equality constrained problems. Convex approximation cannot be applied to equality constraints, which are linearized internally, see Zillober [49]. Problem (14) is solved by the SCP code SCPIP with termination accuracy  $\epsilon = 10^{-7}$  for the optimality condition and  $\epsilon = 10^{-10}$  for maximum constraint violation. Starting values are  $u_0 = 0$  and  $y_0 = 0$  for all test runs. The total number of variables,  $n$ , and the number of equality constraints,  $m$ , are shown in Table 4 together with the number of SCPIP iterations,  $n_{it}$ , and the final objective function value,  $f(u, y)$ . The grid size  $N$  varies between 100 and 600.

### 3. Case Study: Horn Radiators for Satellite Communication

Corrugated horns are frequently used as reflector feed sources for large space antennae, for example for INTELSAT satellites. The goal is to achieve a given spatial energy distribution of the radio frequency (RF) waves, called the radiation or directional characteristic. The transmission quality of the information carried by the RF signals is strongly determined by the directional characteristics of the feeding horn as determined by its geometric structure.

The electromagnetic field theory is based on Maxwell's equations relating the electrical field  $E$ , the magnetic field  $H$ , the electrical displace-

ment, and the magnetic induction to electrical charge density and current density, see Collin [8] or Silver [42]. Under some basic assumptions, particularly homogeneous and isotropic media, Maxwell's equations can be transformed into an equivalent system of two coupled equations. They have the form of a wave equation,

$$\nabla^2 \Psi - c^2 \frac{\partial^2}{\partial t^2} \Psi + f = 0$$

with displacement  $f$  enforcing the wave, and wave velocity  $c$ .  $\Psi$  is to be replaced either by  $E$  or  $H$ , respectively.

For circular horns with rotational symmetry, the usage of cylindrical coordinates  $(\rho, \phi, z)$  is advantageous, especially since only waves propagating in  $z$  direction occur. Thus, a scalar wave equation in cylindrical coordinates can be derived from which general solution is obtained, see for example Collin [8] for more details.

By assuming that the surface of the wave guide has ideal conductivity, and that homogeneous Dirichlet boundary conditions  $\Psi = 0$  for  $\Psi = E$  and Neumann boundary conditions  $\partial\Psi/\partial n = 0$  for  $\Psi = H$  at the surface are applied, we get the eigenmodes or eigenwaves for the circular wave guide. Since they form a complete orthogonal system, electromagnetic field distribution in a circular wave guide can be expanded into an infinite series of eigenfunctions, and is completely described by the amplitudes of the modes. For the discussed problem, only the transversal eigenfunctions of the wave guides need to be considered and the eigenfunctions of the circular wave guide can be expressed analytically by trigonometric and Bessel functions.

In principle, the radiated far field pattern of a horn is determined by the field distribution of the waves emitted from the aperture. On the other hand, the aperture field distribution itself is uniquely determined by the excitation in the feeding wave guide and by the interior geometry of the horn. Therefore, assuming a given excitation, the far field is mainly influenced by the design of the interior geometry of the horn. Usually, the horn is excited by the  $TE_{11}$  mode, which is the fundamental, i.e., the first solution of the wave equation in cylindrical coordinates. In order to obtain a rotational symmetric distribution of the energy density of the field in the horn aperture, a quasi-periodical corrugated wall structure according to Figure 2 is assumed, see Johnson and Jasik [18].

To reduce the number of optimization parameters, the horn geometry is described by two envelope functions from which the actual geometric data for ridges and slots can be derived. Typically, a horn is subdivided into three sections, see Figure 3, consisting of an input section, a conical section, and an aperture section. For the input and the aperture

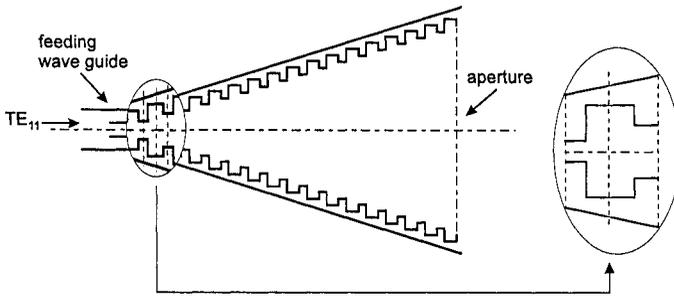


Figure 2. Cross Sectional View of a Circular Corrugated Horn

section, the interior and outer shape of slots and ridges is approximated by a second-order polynomial, while a linear function is used to describe the conical section. It is assumed that the envelope functions of ridges and slots are parallel in conical and aperture section. By this simple analytical approach, it is possible to approximate any reasonable geometry with sufficient accuracy by the design parameters.

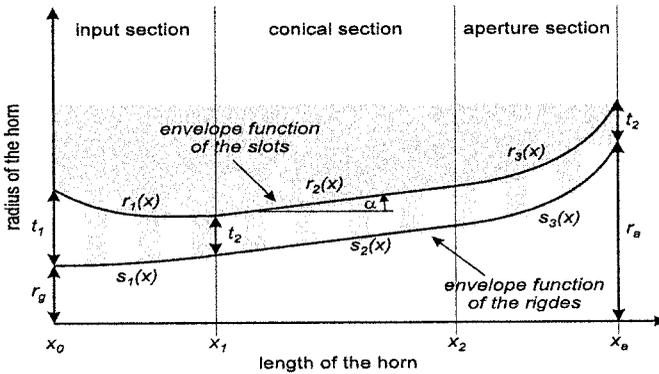


Figure 3. Envelope Functions of a Circular Corrugated Horn

A circular corrugated horn has a modular structure, where each module consists of a step transition between two circular wave guides with different diameters, see Figure 4. The amplitudes of waves, travelling towards and away from the break point, are coupled by a so-called scattering matrix. By combining all modules of the horn step by step, the corresponding scattering matrix describing the total transition of amplitudes from the entry point to the aperture can be computed by successive matrix operations, see Hartwanger et al. [15] or Mittra [24].

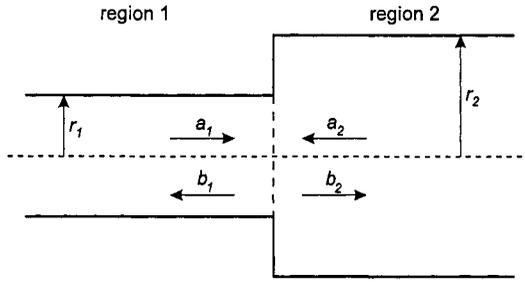


Figure 4. Cross Sectional View of One Module

From Maxwell's equations, it follows that the tangential electrical and magnetic field components must be continuous at the interface between two wave guides. This continuity condition is exploited to compute a relation between the mode amplitudes of the excident  $b_{E,j}^k, b_{H,j}^k$  and incident  $a_{E,j}^k, a_{H,j}^k$  waves in each wave guide of a module, see Figure 4,  $k = 1, 2$ . Then voltage and current coefficients  $U_{H,j}^k, U_{E,j}^k, I_{H,j}^k$ , and  $I_{E,j}^k$  are defined by the amplitudes.

As mentioned before, the tangential fields must be continuous at the transition between two wave guides. Moreover, boundary conditions must be satisfied,  $E_2 = 0$  for  $r_1 \leq r \leq r_2$ . Now only  $n_1$  eigenwaves in region 1 and  $n_2$  eigenwaves in region 2 are considered. The electric field in area 1 is expanded subject to the eigenfunctions in area 2 and the magnetic field in area 2 subject to the eigenfunctions in area 1. After some manipulations, in particular interchanging integrals and finite sums, the following relationship between voltage coefficients in region 1 and 2 can be formulated in matrix notation:

$$\begin{pmatrix} U_E^2 \\ U_H^2 \end{pmatrix} = \begin{pmatrix} X_{EE} & X_{HE} \\ X_{EH} & X_{HH} \end{pmatrix} \begin{pmatrix} U_E^1 \\ U_H^1 \end{pmatrix}. \quad (16)$$

Here  $U_E^k$  and  $U_H^k$  are vectors consisting of the coefficients  $U_{E,j}^k$  and  $U_{H,j}^k$  for  $j = 1, \dots, n_k$ , respectively,  $k = 1, 2$ . The elements of the matrix  $X_{EE}$  are given by

$$X_{EE}^{ij} = \int_0^{r_2} \int_0^{2\pi} e_{E,i}^2(\rho, z, \phi)^T e_{E,j}^1(\rho, z, \phi) \rho d\phi d\rho \quad (17)$$

with tangential field vectors  $e_{E,i}^k(\rho, z, \phi)$  for both regions  $k = 1$  and  $k = 2$ . In the same way  $X_{HE}, X_{EH}$ , and  $X_{HH}$  are defined. Moreover, matrix equations for the current coefficients are available.

Next, the relationship between the mode amplitude vectors  $b_E^k$  and  $b_H^k$  of the excident waves  $b_{E,j}^k$ ,  $b_{H,j}^k$ , and  $a_E^k$  and  $a_H^k$  of the incident waves  $a_{E,j}^k$ ,  $a_{H,j}^k$ ,  $j = 1, \dots, n_k$ ,  $k = 1, 2$ , are evaluated through a so-called scattering matrix. By combining all scattering matrices of successive modules, we compute the total scattering matrix relating the amplitudes at the feed input with those at the aperture,

$$\begin{pmatrix} b_1(p) \\ b_2(p) \end{pmatrix} = \begin{pmatrix} S_{11}^*(p) & S_{12}^*(p) \\ S_{21}^*(p) & S_{22}^*(p) \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \end{pmatrix}. \quad (18)$$

The vector  $a_1$  describes the amplitudes of the modes exciting the horn, the  $TE_{11}$  mode in our case. Thus,  $a_1$  is the  $2n_1$ -dimensional unity vector. The vector  $a_2$  contains the amplitudes of the reflected modes at the horn aperture, known from the evaluation of the far field. Only a simple matrix  $\times$  vector computation is performed to get the modes of reflected waves  $b_1(p)$  and  $b_2(p)$ , once the scattering matrix is known.

The main goal of the optimization procedure is to find an interior geometry  $p$  of the horn so that the distances of  $b_2(p)^j$  from given amplitudes  $\bar{b}_2^j$  for  $j = 1, \dots, 2n_2$  become as small as possible. The first component of the vector  $b_1(p)$  is a physically significant parameter, the so-called return loss, representing the power reflected at the throat of the horn. Obviously, this return loss should be minimized as well. The phase of the return loss and further components of  $b_1(p)$  are not of interest.

From these considerations, the least squares optimization problem

$$p \in \mathbb{R}^n : \min \sum_{j=1}^{2n_2} (b_2^j(p) - \bar{b}_2^j)^2 + \mu b_1^1(p)^2 \quad (19)$$

$$p_l \leq p \leq p_u$$

is obtained. The upper index  $j$  denotes the  $j$ -th coefficient of the corresponding vector,  $\mu$  a suitable weight, and  $p_l$ ,  $p_u$  lower and upper bounds for the parameters to be optimized. Note also that complex numbers are evaluated throughout this section, leading to a separate evaluation of the regression function of (19) for the real and imaginary parts of  $b_2^j(p)$ .

The least squares problem is solved by a special variant of NLPQL called DFNLP, see Schittkowski [37], which retains typical features of a Gauss-Newton method after a certain transformation. For a typical test run under realistic assumptions, the radius of the feeding wave guide, and the radius of the aperture are kept constant,  $r_g = 11.28 \text{ mm}$  and  $r_a = 90.73 \text{ mm}$ , where 37 ridges and slots are assumed. Parameter names, initial values  $p_0$ , and optimal solution values  $p_{opt}$  are listed in Table 5. The number of modes, needed to calculate the scattering matrix, is 70.

<i>name</i>	$p_0^i$	$p_{opt}^i$	<i>comment</i>
$x_1$	50.0	111.85	length of input section
$x_{con}$	50.0	0.00	length of conical section
$x_o$	50.0	47.00	length of output section
$\alpha$	28.0	29.00	semi flare angle of conical section
$q$	0.25	0.20	quotient of slot and ridge width
$t_1$	12.5	11.97	depth of first slot in input section
$t_2$	7.2	7.82	depth of slots in conical section

Table 5. Initial and Optimal Parameter Values

Forward differences are used to evaluate numerical derivatives subject to a tolerance of  $10^{-7}$ , and  $\mu = 1$  was set for weighting the return loss. NLPQL needed 51 iterations to satisfy the stopping tolerance  $10^{-7}$ .

#### 4. Case Study: Design of Surface Acoustic Wave Filters

Computer-aided design optimization of electronic components is a powerful tool to reduce development costs on one hand and to improve the performance of the components on the other. A bandpass filter selects a band of frequencies out of the electro-magnetic spectrum. In this section, we consider surface-acoustic-wave (SAW) filters consisting of a piezo-electric substrate, where the surface is covered by metal structures. The incoming electrical signal is converted to a mechanical signal by this setup. The SAW filter acts as a transducer of electrical energy to mechanical energy and vice versa. The efficiency of the conversion depends strongly on the frequencies of the incoming signals and the geometry parameters of the metal structures, for example length, height, etc. On this basis, the characteristic properties of a filter are achieved.

Due to small physical sizes and unique electrical properties, SAW-bandpass filters raised tremendous interest in mobile phone applications. The large demand of the mobile phone industry is covered by large-scale, industrial mass-production of SAW-filters. For industrial applications, bandpass filters are designed in order to satisfy pre-defined electrical specifications. The *art of filter design* consists of defining the internal structure, or the geometry parameters, respectively, of a filter such that the specifications are satisfied. The electrical properties of the filters are simulated based on physical models. The simulation of a bandpass filter consists of the acoustic tracks, i.e., the areas on the piezo-electrical substrate on which the electrical energy is converted to mechanical vibrations and vice versa, and the electrical combinations of the different

acoustic tracks. Typically, only the properties of the acoustic tracks are varied during the design process, and are defined by several physical parameters. Some of them are given in form of real numbers, some others in form of integer numbers. As soon as the filter properties fit to the demands, the mass production of the filter is started.

When observing the surface of a single-crystal, we see that any deviation of an ion from its equilibrium position provokes a restoring force and an electrical field due to the piezo-electric effect. Describing the deviations of ions at the surface in terms of a scalar potential, we conclude that the SAW is described by a scalar wave equation

$$\phi_{tt} = c^2 \Delta \phi . \tag{20}$$

The boundary conditions are given by the physical conditions at the surface and are non-trivial, since the surface is partly covered by a metal layer. In addition, the piezo-electric crystal is non-isotropic, and the velocity of the wave depends on its direction. For the numerical simulation, additional effects such as polarization charges in the metal layers have to be taken into account. Consequently, the fundamental wave equation is not solvable in a closed form.

For this reason, Tobolka [46] introduced the P-matrix model as an equivalent mathematical description of the SAW. One element is a simple base cell, which consists of two acoustic ports, and an additional electric port. The acoustic ports describe the incoming and outgoing acoustic signals, the electrical ports the electric voltage at this cell, see Figure 5. The quantities  $a_1, a_2, b_1$  and  $b_2$  denote the intensities of the acoustic waves. In terms of a description based on the wave equation, we have  $a_1 \propto \phi$ ,  $u$  is the electrical voltage at the base cell, and  $i$  is the electrical current.

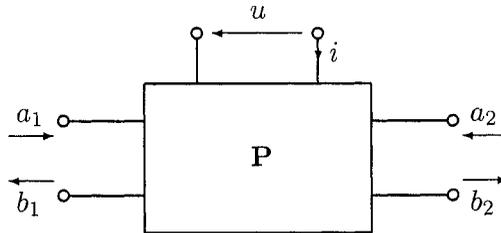


Figure 5. Base Cell of the P-Matrix Model with Two Acoustic and One Electric Port

The P-matrix model describes the interaction of the acoustic waves at the acoustic ports, with the electric port in linear form. Typically, a

transformation is given in the form

$$\begin{pmatrix} b_1 \\ b_2 \\ i \end{pmatrix} = \mathbf{P} \begin{pmatrix} a_1 \\ a_2 \\ u \end{pmatrix}, \tag{21}$$

where for example

$$\mathbf{P} = \begin{pmatrix} 0 & 1 & E \\ 1 & 0 & -E^* \\ -2E & 2E^* & 2|E|^2 + i(-\mathcal{H}\{2|E|^2\} + \omega C) \end{pmatrix}. \tag{22}$$

$\mathcal{H}$  denotes the Hilbert transformation,  $C$  the static capacity between two fingers,  $E$  the excitation given by

$$E = -i \, 0.5 \sqrt{\omega W K} \cdot \int_{tr} \sigma_e(x) \exp^{-ik|x|} dx ,$$

$\omega$  the frequency,  $W$  the aperture of the IDT,  $K$  a material constant, and  $\sigma_e$  the electric load distribution.

In general, the elements of  $P$  are the dimensionless acoustic reflection and transmission coefficients in the case of a short-circuited electrical port. The  $2 \times 2$  upper diagonal submatrix is therefore the scattering matrix of the acoustic waves and describes the interaction of the incoming and outgoing waves. Other elements characterize the relation of the acoustic waves with the electric voltage, i.e., the piezo-electric effect of the substrate, or the admittance of the base cell, i.e., the the quotient of current to voltage and the reciprocal value of the impedance.

Proceeding from the P-matrix model, we calculate the scattering matrix  $S$ . This matrix is the basic physical unit that describes the electro-acoustic properties of the acoustic tracks, and finally the filter itself. The transmission coefficient  $T$  is one element of the scattering matrix,  $T = S_{21}$ .

Mobile phone manufacturers provide strict specifications towards the design of a bandpass filter. Typically, the transmission has to be above certain bounds in the pass band and below certain bounds in the stop band depending on the actual frequency. These specifications have to be achieved by designing the filter in a proper way. Depending on the exact requirements upon the filter to be designed, different optimization problems can be derived.

To formulate the optimization problem, let us assume that  $x \in \mathbb{R}^n$  denotes the vector of continuous real design variables and  $y \in \mathbb{Z}^m$  the vector of the integer design variables.  $Z$  is the set of all integer values. By  $T(f, x, y)$  we denote the transmission subject to frequency  $f$  and

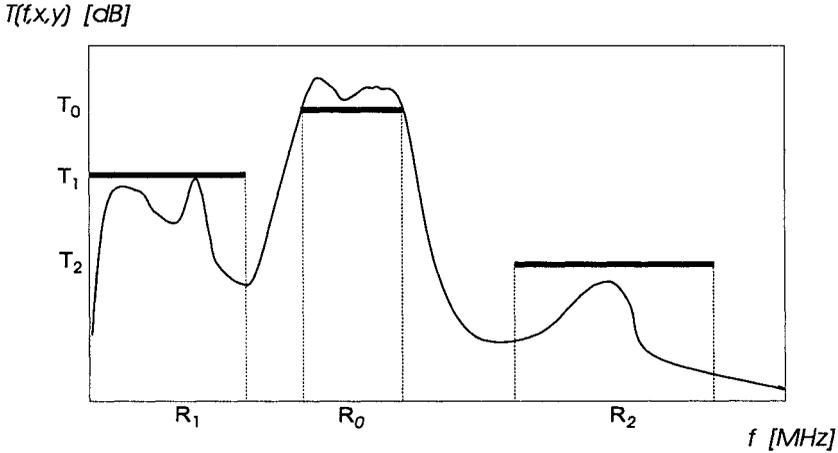


Figure 6. Design Goals of an SAW Filter

the optimization variables  $x$  and  $y$ . Some disjoint intervals  $R_0, \dots, R_s$  define the design space within the frequency interval  $f_l \leq f \leq f_u$ . Our goal is to maximize the minimal distance of transmission  $T(f, x, y)$  over the interval  $R_0$ , under lower bounds  $T_1, \dots, T_s$  for the transmission in the remaining intervals  $R_1, \dots, R_s$ . Moreover, it is required that the transmission is always above a certain bound in  $R_0$ , i.e., that  $T(f, x, y) \geq T_0$  for all  $f \in R_0$ . The optimization problem is formulated as

$$\begin{aligned} & \max \min \{T(f, x, y) : f \in R_0\} \\ & x \in \mathbb{R}^n, y \in Z^m : T(f, x, y) \leq T_i \text{ for } f \in R_i, i = 1, \dots, s, \quad (23) \\ & \underline{x} \leq x \leq \bar{x}, \quad \underline{y} \leq y \leq \bar{y}. \end{aligned}$$

Here  $\underline{x}, \bar{x} \in \mathbb{R}^n$  and  $\underline{y}, \bar{y} \in Z^m$  are lower and upper bounds for the design variables.

To transform the infinite dimensional optimization problem into a finite dimensional one, we proceed from a given discretization of the frequency variable  $f$  by an equidistant grid in each interval. The corresponding index sets are called  $J_0, J_1, \dots, J_s$ . Let  $l$  be the total number of all grid points. First we introduce the notation  $T_j(x, y) = T(f_j, x, y)$ ,  $f_j$  suitable grid point,  $j = 1, \dots, l$ . All indices are ordered sequentially so that  $\{1, \dots, l\} = J_0 \cup J_1 \cup \dots \cup J_s$ , i.e.,  $J_0 = \{1, \dots, l_0\}$ ,  $J_1 = \{l_0 + 1, \dots, l_1\}$ ,  $\dots$ ,  $J_s = \{l_{s-1} + 1, \dots, l\}$ . Then the discretized

optimization problem is

$$\begin{aligned} & \max \min \{T_j(x, y) : j \in J_0\} \\ x \in \mathbb{R}^n, y \in Z^m : & T_j(x, y) \leq T_i \text{ for } j \in J_i, i = 1, \dots, s, \\ & \underline{x} \leq x \leq \bar{x}, \underline{y} \leq y \leq \bar{y}. \end{aligned} \quad (24)$$

The existence of a feasible design is easily checked by performing the test  $T_j(x, y) \geq T_0$  for all  $j \in J_0$ . Problem (24) is equivalent to a smooth nonlinear program after a simple standard transformation.

Integer variables are handled in a specific way, see van de Braak et al. [47]. Since continuous relaxation is not possible, a certain combination of a quadratic approximation and direct search algorithm in the integer space is developed. A function evaluation for a set of integer variables requires the complete solution of the continuous optimization problem (24) by the SQP code NLPQL.

Lower and upper bounds for the ten design variables under consideration are shown in Table 6 together with initial values and final ones obtained by the code NLPQL. Simulation is performed with respect to 154 frequency points leading to 174 constraints in the continuous model. Altogether 36 calls of NLPQL are made within four quadratic approximation cycles and one additional direct search iteration. The total number of simulations, i.e., the number of evaluations of the transmission energy  $T_j(x, y)$  for all  $j$ , is 434 without the function calls needed for the gradient approximations. The purpose of the example is to show that the design goal is only achieved by taking also integer variables into account.

## 5. Case Study: Optimal Control of an Acetylene Reactor

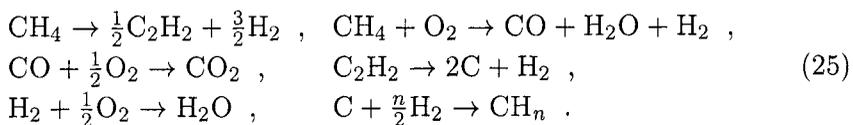
The computation of optimal feed controls for chemical reactors, especially for tubular reactors, is a well-known technique, see Edgar and Himmelblau [9], Nishida et al. [27], and Buzzi-Ferraris et al. [6–7]. The mathematical model is given as a distributed parameter system consisting of a set of first-order partial differential equations in one space dimension. The chemical reactions and the temperature depend on the spatial variable, whereas the dynamical decrease of the cross-sectional area caused by coke deposition is time-dependent. In both cases, we know initial values either in the form of time-dependent feed control functions or a constant tube diameter. Alternative approaches to compute optimal reactor feed rates are discussed in Birk et al. [4], and Liepelt and Schittkowski [20].

<i>variable</i>	<i>lower bound</i>	<i>initial value</i>	<i>optimal value</i>	<i>upper bound</i>
$y_1$	7	19	12	19
$y_2$	12	24	14	25.0
$y_3$	100	130	124	150
$x_1$	5.0	11.58	9.589	15.0
$x_2$	50.0	50.0	92.39	150.0
$x_3$	10.5	11.39	11.25	11.5
$x_4$	10.0	10.61	10.62	11.0
$x_5$	0.3	0.3	0.3	0.5
$x_6$	0.95	1.033	1.031	1.05
$x_7$	0.95	1.031	1.023	1.05
$x_8$	0.95	1.012	1.015	1.02
$x_9$	0.985	1.001	0.998	1.03
$x_{10}$	1.0	1.0	1.000	1.03

Table 6. Bounds, Initial, and Optimal Values for Design Variables

We consider a chemical reactor producing acetylene ( $C_2H_2$ ), reacting methane ( $CH_4$ ) in natural gas with oxygen. This reaction requires less oxygen compared with complete combustion. The products are quickly quenched to keep the acetylene from being converted entirely to coke, see Wansbrough [48]. During the reaction process, a small part of the carbon is deposited in the reactor as coke. The quantity and its distribution in the reactor depend on the reaction equations. Since it is impossible to measure the cross-sectional area directly, we need a mathematical model that describes the functional dependence of the cross-sectional area upon other system parameters. If the deposition of coke reaches a certain limit, the reactor must be stopped and the tube cleaned.

There are six reactions to be taken into account. Reactions 1 through 5 are the main ones that produce acetylene, but also undesirable byproducts such as coke. Reaction 6 is included only to balance the hydrogen stoichiometry,



The reactions can be described by the following system of ordinary differential equations, where  $C_i$  denotes the molar concentration of the  $i$ -th

component,

$$\begin{aligned}
 v(x, t) \frac{\partial}{\partial x} C_1(x, t) &= -r_1(x, t) - r_2(x, t) , \\
 v(x, t) \frac{\partial}{\partial x} C_2(x, t) &= -r_2(x, t) - \frac{1}{2}r_3(x, t) - \frac{1}{2}r_5(x, t) , \\
 v(x, t) \frac{\partial}{\partial x} C_3(x, t) &= \frac{1}{2}r_1(x, t) - r_4(x, t) , \\
 v(x, t) \frac{\partial}{\partial x} C_4(x, t) &= r_3(x, t) , \\
 v(x, t) \frac{\partial}{\partial x} C_5(x, t) &= \frac{3}{2}r_1(x, t) + r_2(x, t) + r_4(x, t) - r_5(x, t) \\
 &\quad - n(1 - \varepsilon)r_4(x, t) , \\
 v(x, t) \frac{\partial}{\partial x} C_6(x, t) &= r_2(x, t) - r_3(x, t) , \\
 v(x, t) \frac{\partial}{\partial x} C_7(x, t) &= r_2(x, t) + r_5(x, t) , \\
 v(x, t) \frac{\partial}{\partial x} C_8(x, t) &= 2(1 - \varepsilon)r_4(x, t)
 \end{aligned} \tag{26}$$

with a reaction parameter  $\varepsilon$ . Since the acetylene reactor is controlled by the feeds of natural gas and oxygen, these are the only components with non-vanishing initial values. Initial molar concentrations are given by

$$C_1^0(t) = \frac{\dot{m}_1(t)\rho_n}{M_1\dot{m}(t)} , \quad C_2^0(t) = \frac{\dot{m}_2(t)\rho_n}{M_2\dot{m}(t)} \tag{27}$$

with  $\rho_n RT_0 = p_0$  assuming ideal gas law.

The velocity of the mixture in the reactor depends on the cross-sectional area  $A(x, t)$ , the total mass flow  $\dot{m}(t)$  in the reactor, and the density  $\rho(x, t)$ , and is given by

$$v(x, t) = \frac{\dot{m}(t)}{\rho(x, t)A(x, t)} , \tag{28}$$

where the total mass flow  $\dot{m}(t) = \dot{m}_1(t) + \dot{m}_2(t)$  is the sum of the two input flows. The density of the mixture is given by  $\rho(x, t) = \sum_{j=1}^8 C_j(x, t)M_j$ , where  $M_j$  denotes the molar weight of the  $j$ -th component, and the temperature in the reactor can be described by the differential equation

$$\frac{\partial}{\partial x} T(x, t) = \frac{1}{\rho(x, t)v(x, t)c_p(x, t)} \sum_{i=1}^5 r_i(x, t)\Delta H_i \tag{29}$$

with the initial condition  $T(0, t) = T_0$ . The incremental change of the temperature is determined by the rate of heat release for all reactions, which depends on the total heat capacity  $c_p(x, t)$ .

The eight material balance equations depend on the rates of the various reactions and on the velocity of the mixture in the reactor, because this speed determines the time that the components spent in the reactor. The reaction rates are expressed by

$$\begin{aligned}
 r_1(x, t) &= k_1 \exp\left(-\frac{E_1}{R}(1/T(x, t) - 1/T_r)\right) C_1^{a_1}(x, t) , \\
 r_2(x, t) &= k_2 \exp\left(-\frac{E_2}{R}(1/T(x, t) - 1/T_r)\right) C_1(x, t) C_2^{a_2}(x, t) , \\
 r_3(x, t) &= k_3 \exp\left(-\frac{E_3}{R}(1/T(x, t) - 1/T_r)\right) C_6(x, t) C_2^{0.5}(x, t) , \\
 r_4(x, t) &= k_4 \exp\left(-\frac{E_4}{R}(1/T(x, t) - 1/T_r)\right) C_3^{a_4}(x, t) , \\
 r_5(x, t) &= k_5 \exp\left(-\frac{E_5}{R}(1/T(x, t) - 1/T_r)\right) C_5(x, t) C_2^{0.5}(x, t) ,
 \end{aligned} \tag{30}$$

with five reaction constants  $k_1, \dots, k_5$ , five activation energies  $E_1, \dots, E_5$ , and three reaction orders  $a_1, a_2$ , and  $a_4$ . For the smaller and less important reactions, the stoichiometric order can be used as an estimate for the reaction order. For the other reactions, these parameters have to be derived from the real reactor that is going to be examined. The average temperature  $T_r$  is used to scale the exponential functions and  $R$  denotes the gas constant.

If we neglect the deposition of coke, the underlying differential equation is stationary and does not depend on the time. But a decrease of the cross-sectional area  $A(x, t)$  increases the velocity  $v(x, t)$  of the mixture in the reactor, which influences the incremental change of the concentrations  $C(x, t)$  and the temperature  $T(x, t)$ , see (26), (28), and (29). The coke deposition is modelled by the time-dependent differential equation

$$\frac{\partial}{\partial t} A(x, t) = -\beta r_4(x, t) \tag{31}$$

with initial condition  $A(x, 0) = A_0$  and reaction parameter  $\beta$ . A typical contour plot of the cross sectional area over time and spatial variable is shown in Figure 7.

The optimal control problem consists of maximizing

$$J(s) = \int_0^{T_{max}} \left( \sum_{i=1}^8 P_i(t) \dot{m}_j(t) - \sum_{i=1}^2 P_i(t) s_i(t) \right) dt \tag{32}$$

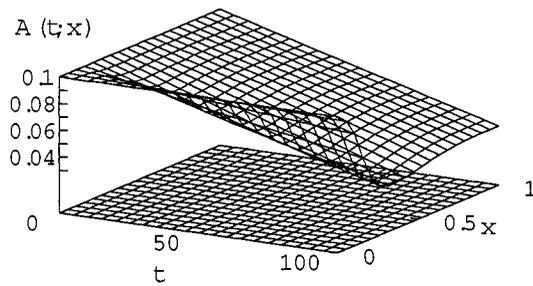


Figure 7. Cross Sectional Area

over all piecewise linear functions within a given range subject to the constraints

$$\begin{aligned}
 A_{min} &\leq A(x, t) && , \\
 0 &\leq T(x, t) \leq T_{max} && , \\
 \dot{m}_{k,min}(t) &\leq \dot{m}_k(L, t) \leq \dot{m}_{k,max}(t) && , \quad k = 1, \dots, 8 \quad ,
 \end{aligned}
 \tag{33}$$

where  $P_i(t)$  is the price of the  $i$ -th component. Maximum temperature is  $T_{max} = 1.200$  and the tube length is 1, i.e.,  $x_L = 0$  and  $x_R = 1$ . More details and also the numerical data are presented in Birk et al. [4], where in addition also the optimal positions of maintenance intervals are to be computed.

The partial differential equation is discretizing subject to the time variable  $t$ , since the cross sectional area  $A(x, t)$  is monotone decreasing without any steep fronts. Thus, it is possible to perform a few Euler steps for integrating (31) leading to a system of ordinary differential equations in  $x$ . We use 5 equidistant time values for approximating the two control variables by piecewise linear functions. The number of lines is 15 and a 5-point difference formula is applied to discretize spatial derivatives. The resulting ODE is integrated by RADAU5 of Hairer and Wanner [14] with a relative and absolute error tolerance of  $10^{-6}$ . Thus we get 10 optimization variables and 510 nonlinear constraints by discretizing (26) at equidistant time and spatial values.

The SQP method NLPQL needs a relatively large number of 227 iterations to reach a solution. Obviously, the starting values are very poor as indicated by the values in the last column of the subsequent screen display. Moreover, objective function and constraints are badly scaled, see first and second column, and the stopping criterion KT for optimality, see last column, is decreased from  $10^7$  to  $10^{-6}$ .

IT	F	SCV	NA	I	ALPHA	KT
1	-.13741137D+05	.16D+04	540	0	.00D+00	.17D+07
2	-.23624315D+05	.54D+01	12	1	.10D+01	.88D+06
3	-.23938874D+05	.95D-02	12	1	.10D+01	.71D+05
4	-.24234123D+05	.35D-02	9	1	.10D+01	.90D+04
5	-.24339670D+05	.93D-04	8	1	.10D+01	.17D+02
.	.	.	.	.	.	.
.	.	.	.	.	.	.
224	-.28064309D+05	.00D+00	0	1	.10D+01	.38D+00
225	-.28064499D+05	.00D+00	0	1	.10D+01	.14D-02
226	-.28064500D+05	.00D+00	0	1	.10D+01	.42D-04
227	-.28064500D+05	.00D+00	0	1	.10D+01	.49D-06

But the final convergence speed is very fast indicating that at least a local solution is approximated. The optimal control function for  $O_2$  is shown in Figure 8, whereas the control variable for  $CH_4$  attains its upper bound and is not displayed.

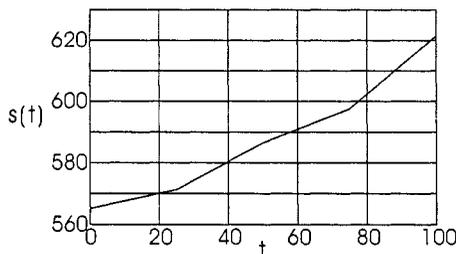


Figure 8. Optimal Input Control of  $O_2$

## 6. Case Study: Weight Reduction of a Cruise Ship

In this section, we describe an example arising in ship building industry with the goal to minimize the total weight of the mechanical structure. The optimization problem is part of the overall design process of a new cruise ship and the results are essential to analyze further substructures by engineers. More details about the project can be found in Zillober and Vogel [54]. The problem is to find the optimal thickness distribution of beams and shell elements of one of the worlds largest cruise ships called *Radiance of the Seas* with a total length of about 300 meters with respect to minimal weight subject to thousands of stress constraints.

The discretized finite element model of the ship consists of 32946 shell and 33637 beam elements with 46986 nodes in total subject to two different load cases called *sagging* and *hogging*, see Figure 9. To formulate

a practically relevant optimization problem, certain elements are linked to get finally  $n = 415$  design variables  $x_i$  with different shell thicknesses,  $i = 1, \dots, n$ . For each region, one element is selected for which the component stresses  $(\sigma^x)^s, (\sigma^y)^s, (\sigma^{xy})^s$  for load case *sagging* and  $(\sigma^x)^h, (\sigma^y)^h, (\sigma^{xy})^h$  for load case *hogging* are constrained. These quantities are calculated at the centroid of the middle layer of the element by the finite element analysis system ANSYS.

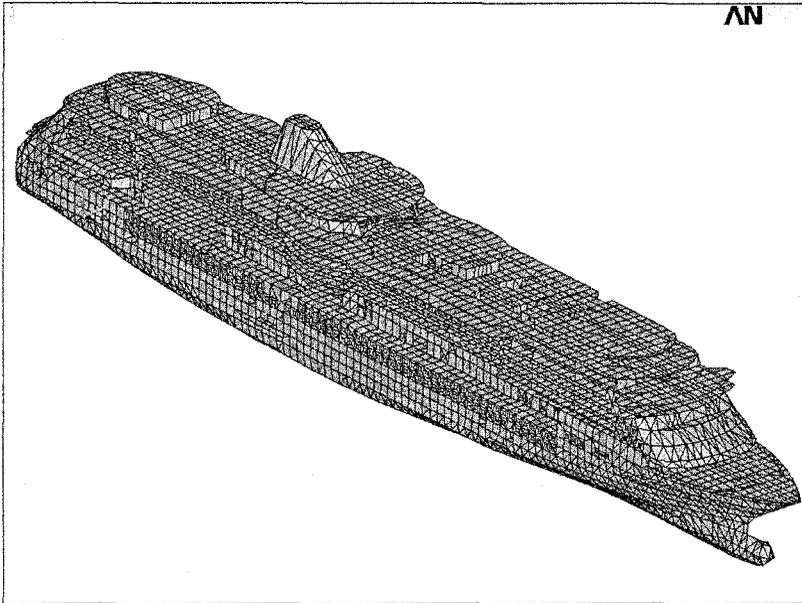


Figure 9. Finite Element Structure of the Ship to be Optimized

The optimization problem can be formulated as

$$\begin{aligned}
 & \min \quad \text{Volume}(x), \\
 & \sigma_{\min}^x \leq (\sigma_j^x(x))^s \leq \sigma_{\max}^x, \quad j = 1, \dots, n, \\
 & \sigma_{\min}^y \leq (\sigma_j^y(x))^s \leq \sigma_{\max}^y, \quad j = 1, \dots, n, \\
 & \sigma_{\min}^{xy} \leq (\sigma_j^{xy}(x))^s \leq \sigma_{\max}^{xy}, \quad j = 1, \dots, n, \\
 x \in \mathbb{R}^{415} : & \sigma_{\min}^x \leq (\sigma_j^x(x))^h \leq \sigma_{\max}^x, \quad j = 1, \dots, n, \\
 & \sigma_{\min}^y \leq (\sigma_j^y(x))^h \leq \sigma_{\max}^y, \quad j = 1, \dots, n, \\
 & \sigma_{\min}^{xy} \leq (\sigma_j^{xy}(x))^h \leq \sigma_{\max}^{xy}, \quad j = 1, \dots, n, \\
 & 5 \leq x_i \leq 30 \quad i = 1, \dots, n.
 \end{aligned} \tag{34}$$

Thus, the optimization problem consists of 415 optimization variables and 4980 constraints. It should be noted that the mechanical model does not take into account manufacturing constraints, such as fatigue life, buckling, etc. In other words, the weight reduction must be considered only as one part of the overall design process. Due to the high computational cost of the finite element simulations to be performed, the maximum number of iterations is set to 20. The total calculation time is about 30 hours for an SGI-ORIGIN workstation with 2 GB main memory allocated. SCIP of Zillober [51] terminated at a thickness distribution which corresponds to an acceptable weight reduction compared to the initial weight of the ship. The objective function history is shown in Figure 10.

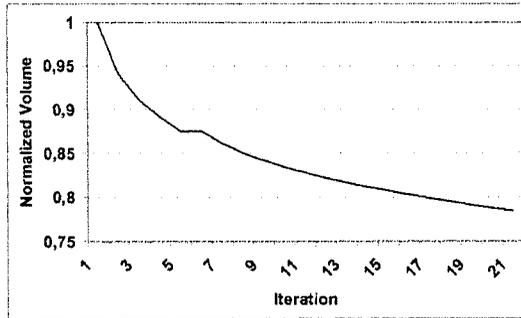


Figure 10. Iteration History of Structural Weight

## Acknowledgments

The authors wish to thank Meyer Werft Papenburg for granting permission to publish Figure 9.

## References

- [1] M.P. Bendsøe (1989). Optimal shape design as a material distribution problem. *Structural Optimization*, 1:193–202, 1989.
- [2] M.P. Bendsøe. *Optimization of Structural Topology, Shape and Material*. Springer, Heidelberg, 1995.
- [3] M.P. Bendsøe and O. Sigmund O. *Topology Optimization — Theory, Methods and Applications*. Springer, Heidelberg, 2003.

- [4] J. Birk, M. Liepelt, K. Schittkowski, and F. Vogel. Computation of optimal feed rates and operation intervals for tubular reactors. *Journal of Process Control*, 9:325–336, 1999.
- [5] I. Bongartz, A.R. Conn, N.I.M. Gould, and P.L. Toint. CUTE: Constrained and unconstrained testing environment. *ACM Trans. Math. Software*, 21:123–160, 1995.
- [6] G. Buzzi-Ferraris, G. Facchi, P. Forzetti, and E. Tronconi. Control optimization of tubular catalytic decay. *Industrial Engineering in Chemistry*, 23:126–131, 1984.
- [7] G. Buzzi-Ferraris, M. Morbidelli, P. Forzetti, and S. Carra. Deactivation of catalyst - mathematical models for the control and optimization of reactors. *International Chemical Engineering*, 24:441–451, 1984.
- [8] R.E. Collin. *Field Theory of Guided Waves*. IEEE Press, New York, 1991.
- [9] T.F. Edgar and D.M. Himmelblau. *Optimization of Chemical Processes*. McGraw-Hill, New York, 1988.
- [10] C. Fleury. An efficient dual optimizer based on convex approximation concepts. *Structural Optimization*, 1:81–89, 1989.
- [11] C. Fleury and V. Braibant. Structural optimization – a new dual method using mixed variables. *International Journal for Numerical Methods in Engineering*, 23:409–428, 1986.
- [12] D. Goldfarb and A. Idnani. A numerically stable method for solving strictly convex quadratic programs. *Mathematical Programming*, 27:1–33, 1983.
- [13] N.I.M. Gould and P.L. Toint. SQP methods for large-scale nonlinear programming. In *System Modelling and Optimization: Methods, Theory and Applications*. Kluwer, 2000.
- [14] E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems*. Springer, Berlin, Heidelberg, New York, 1991.
- [15] C. Hartwanger, K. Schittkowski, and H. Wolf. Computer aided optimal design of horn radiators for satellite communication. *Engineering Optimization*, 33:221–244, 2000.
- [16] W. Hock and K. Schittkowski. A comparative performance evaluation of 27 nonlinear programming codes. *Computing*, 30:335–358, 1983.
- [17] K. Ito and K. Kunisch. Augmented Lagrangian-SQP methods for nonlinear optimal control problems of tracking type. *SIAM Journal on Optimization*, 6:96–125, 1996.
- [18] R.C. Johnson and H. Jasik. *Antenna Engineering*. McGraw Hill, New York, 1984.
- [19] G. Knepe, J. Krammer, and E. Winkler. Structural optimization of large scale problems using MBB-LAGRANGE. Report MBB-S-PUB-305, Messerschmitt-Bölkow-Blohm, D-81663 Munich, 1987.
- [20] M. Liepelt and K. Schittkowski. Optimal control of distributed systems with break points. In M. Grötschel, S.O. Krumke, and J. Rambau, editors, *Online Optimization of Large Scale Systems*, pages 271–294. Springer, Berlin, 2000.
- [21] F.A. Lootsma. Fuzzy performance evaluation of nonlinear optimization methods. *Journal of Information and Optimization Sciences*, 10:15–44, 1981.

- [22] H. Maurer and H.D. Mittelmann. Optimization techniques for solving elliptic control problems with control and state constraints: Part I. Boundary control. *Computational Optimization and Applications*, 16:29–55, 2000.
- [23] H. Maurer and H.D. Mittelmann. Optimization techniques for solving elliptic control problems with control and state constraints: Part ii. distributed control. *Computational Optimization and Applications*, 18:141–160, 2001.
- [24] R. Mittra. *Computer Techniques for Electromagnetics*. Pergamon Press, Oxford, 1973.
- [25] H.P. Mlejnek. Some aspects of the genesis of structures. *Structural Optimization*, 5:64–69, 1992.
- [26] V.H. Nguyen, J.J. Strodiot, and C. Fleury. A mathematical convergence analysis for the convex linearization method for engineering design optimization. *Engineering Optimization*, 11:195–216, 1987.
- [27] N. Nishida, A. Ichikawa, and E. Tazaki. Optimal design and control in a class of distributed parameter systems under uncertainty. *AIChE Journal*, 18:561–568, 1972.
- [28] P.Y. Papalambros and D.J. Wilde. *Principles of Optimal Design*. Cambridge University Press, Cambridge, 2000.
- [29] M.J.D. Powell. The convergence of variable metric methods for nonlinearly constrained optimization calculations. In *Nonlinear Programming 3*. Academic Press, 1978.
- [30] M.J.D. Powell. A fast algorithm for nonlinearly constraint optimization calculations. In *Numerical Analysis, G.A. Watson ed., Lecture Notes in Mathematics*, volume 630. Springer, 1978.
- [31] R.T. Rockafellar. Augmented Lagrange multiplier functions and duality in non-convex programming. *Journal on Control*, 12:268–285, 1974.
- [32] T.L. Saaty. *The Analytic Hierarchy Process, Planning, Priority Setting, Resource Allocation*. McGraw Hill, New York, 1980.
- [33] K. Schittkowski. *Nonlinear Programming Codes*, volume 183 of *Lecture Notes in Economics and Mathematical Systems*. Springer, Berlin, Heidelberg, New York, 1980.
- [34] K. Schittkowski. Theory, implementation and test of a nonlinear programming algorithm. In *Optimization Methods in Structural Design*. N. Olhoff eds., Wissenschaftsverlag, 1983.
- [35] K. Schittkowski. NLPQL: A Fortran subroutine solving constrained programming problems. *Annals of Operations Research*, 5:485–500, 1985.
- [36] K. Schittkowski. *More Test Examples for Nonlinear Programming*, volume 187 of *Lecture Notes in Economics and Mathematical Systems*. Springer, Berlin, Heidelberg, New York, 1987.
- [37] K. Schittkowski. Solving nonlinear least squares problems by a general purpose sqp-method. In *Trends in Mathematical Optimization, K.-H. Hoffmann, J.-B. Hiriart-Urruty, C. Lemarechal, J. Zowe eds., International Series of Numerical Mathematics*, volume 184, pages 295–309. Birkhäuser, 1988.
- [38] K. Schittkowski. Solving nonlinear programming problems with very many constraints. *Optimization*, 25:179–196, 1992.

- [39] K. Schittkowski. NLPQLP: A new fortran implementation of a sequential quadratic programming algorithm for parallel computing. Research report, Department of Mathematics, University of Bayreuth, D-95440 Bayreuth, 2001.
- [40] K. Schittkowski. *Numerical Data Fitting in Dynamical Systems*. Kluwer, Dordrecht, 2002.
- [41] K. Schittkowski, C. Zillober, and R. Zotemantel. Numerical comparison of nonlinear programming algorithms for structural optimization. *Structural Optimization*, 7:1–28, 1994.
- [42] S. Silver. *Microwave Antenna Theory and Design*. McGraw Hill, 1949.
- [43] P. Spellucci. *Numerische Verfahren der nichtlinearen Optimierung*. Birkhäuser, 1993.
- [44] J. Stoer. Foundations of recursive quadratic programming methods for solving nonlinear programs. In *Computational Mathematical Programming*, K. Schittkowski, ed., NATO ASI Series, Series F: Computer and Systems Sciences, volume 15. Springer, 1985.
- [45] K. Svanberg. The method of moving asymptotes – a new method for structural optimization. *International Journal for Numerical Methods in Engineering*, 24:359–373, 1987.
- [46] G. Tobolka. Mixed matrix representation of SAW transducers. *Proceedings of the IEEE Ultrasonics Symposium*, 26:426–428, 1979.
- [47] G. van de Braak, M. J. Bünner, and K. Schittkowski. Optimal design of electronic components by mixed-integer nonlinear programming. *To appear: Engineering Optimization*, 2003.
- [48] R.W. Wansbrough. Modeling chemical reactors. *Chemical Engineering*, 5:95–102, 1985.
- [49] C. Zillober. A combined convex approximation – interior point approach for large scale nonlinear programming. *Optimization and Engineering*, 2:51–73, 2001.
- [50] C. Zillober. Global convergence of a nonlinear programming method using convex approximations. *Numerical Algorithms*, 27:256–289, 2001.
- [51] C. Zillober. SCIP – an efficient software tool for the solution of structural optimization problems. *Structural and Multidisciplinary Optimization*, 24:362–371, 2002.
- [52] C. Zillober. Software manual for SCIP 2.3. Research report, Department of Mathematics, University of Bayreuth, D-95440 Bayreuth, 2002.
- [53] C. Zillober, K. Schittkowski, and K. Moritzen. Very large scale optimization by sequential convex programming. *Optimization Methods and Software*, 18:103–121, 2004.
- [54] C. Zillober and F. Vogel. Solving large scale structural optimization problems. In *Proceedings of the 2nd ASMO UK/ISSMO Conference on Engineering Design Optimization*, pages 273–280. J. Siem ed., University of Swansea, Wales, 2000.