Numerical treatment of a shape optimization problem in thermoelasticity *

J. Sprekels[†]
Institute of Applied Analysis
and Stochastics (IAAS)
Berlin

H. Goldberg and F. Tröltzsch[‡]
Technical University of Chemnitz–Zwickau
Faculty of Mathematics

Abstract

A shape optimization problem connected with the thermal deformation of elastic bodies is considered. Based on a finite element discretization of the thermoelastic system, a gradient method for solving the associated nonlinear programming problem is discussed. Several approaches to calculate the needed gradients are investigated. Finally, some numerical examples are presented.

1 Introduction

In [15], Sprekels and Tröltzsch investigated an optimal shape design problem for a system of thermoelastic equations. It is connected with the following question: Suppose that an isotropic and homogeneous solid body is subjected to a prescribed (known) thermal treatment. Due to the temperature change, the body undergoes a thermoelastic deformation, that is, the induced thermal stresses force the body to change its shape in time. The following question arises: Which *initial* shape must the body be given in order that its *final* shape after the thermal treatment resembles a desired prescribed form as closely as possible?

A very simplified problem of this type was treated in [15] by transforming the system of equations onto a fixed domain, thus converting the optimal shape design to the optimal control of the coefficients in a transformed system of partial differential equations on a fixed domain.

*Supported by Deutsche Forschungsgemeinschaft (DFG), SPP "Anwendungsbezogene Optimierung und Steuerung"

[†]Mohrenstraße 39, D–10117 Berlin, Germany

[‡]PSF 964, D–09009 Chemnitz, Germany

The existence of optimal shapes and associated first order necessary conditions were proved. In our opinion, this method is not sufficiently effective for solving the problem numerically. First, the transformation ends up with a convection term in the heat equation which may be a source of difficulties. Second, as the transformed system is not standard, the available numerical methods for thermoelastic systems do not apply.

Therefore, we decided to discretize the underlying domain, to handle the system of thermoelasticity by an effective finite element multigrid technique and to establish a descent method to determine the optimal nodal points of the discretized shape. Methods of this type were frequently and succesfully applied to different problems. Basic work to this field was done by PIRONNEAU [12], HASLINGER/ NEITTAANMÄKI [5], and recently, by SOKOLOWSKI/ ZOLESIO [14]. We refer also to the references of these textbooks. Moreover, we mention BUTT/ RUBIO [1], CHEU [2], DELFOUR/ PAYRE/ ZOLESIO [3], DEMS/ MRÓZ [4], HASLINGER/ NEITTAANMÄKI/ SALMENJOKI [6], KAMIYA/ KITA [7], KARAFIAT [8], MÄKINEN [9], MERIC [10], MYŚLINSKI [11], SALMENJOKI [13], and YAO/ CHOI [16], where finite element techniques are used to discuss optimal shape design problems analytically and / or numerically.

The remainder of the paper is organized as follows: In section 2 we formulate the optimal shape design problem. Its discretized form is introduced in section 3. Different numerical approaches to solve the discretized problem are considered in section 4. We focus our investigations on the computation of derivatives with respect to nodal points in order to obtain a descent direction for gradient methods. The final part of this section is devoted to the presentation of some numerical examples.

2 The optimal shape design problem

Let $s \in C^{1,r}[0,d]$ be given, and suppose that initially, at $t = 0$, the body occupies the domain (see Fig. 1)

$$\Omega(s) = \{(x_1, x_2) \in I\!R^2 : 0 < x_1 < d,\ 0 < x_2 < s(x_1)\}. \tag{2.1}$$

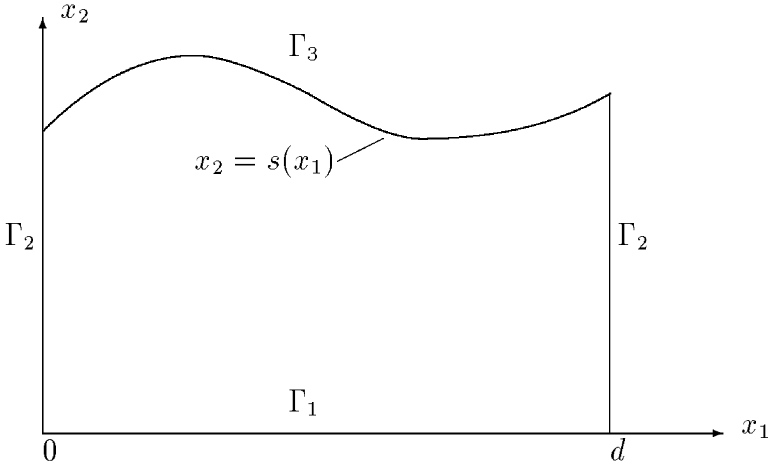

Figure 1: Shape of the domain $\Omega(s)$.

The set $\Gamma_3(s) = \{x_2 = s(x_1)\}$ denotes that part of the boundary $\Gamma(s)$ which is to be shaped.

The function s plays the role of the control variable. Let $\underline{s}$, $\overline{s}$, $\overline{s}'$, c denote fixed positive constants, and let $r \in (0,1)$, s_a, $s_b \in I\!R$ be given. We assume that s satisfies the conditions

$$0 < \underline{s} \le s(\tau) \le \bar{s}\,, \quad |s'(\tau)| \le \bar{s}'\,, \quad \forall\, \tau \in [0,d]\,, \tag{2.2}$$

$$|s'(\tau_1) - s'(\tau_1)| \le c\,|\tau_1 - \tau_2|^r\,, \quad \forall\, \tau \in [0,d]\,, \tag{2.3}$$

$$s(0) = s_a\,, \quad s(d) = s_b\,, \tag{2.4}$$

$$\underline{s} \le s_a \le \bar{s}\,, \quad \underline{s} \le s_b \le \bar{s}\,. \tag{2.5}$$

Thus, $s \in U^{ad}$, where $U^{ad} = \{s \in C^{1,r}[0,d] : s \text{ satisfies (2.2)–(2.5) }\}$, defines the set of *admissible functions.* Note that U^{ad} forms a nonempty, convex and compact subset of $C^1[0,d]$.

Next, let $\vartheta = \vartheta(t,x)$, $t \in [0,T]$, $x = (x_1,x_2) \in \overline{\Omega}(s)$, denote the *temperature.* We assume that ϑ satisfies the parabolic problem

$$\begin{array}{rcllll} \vartheta_t(t,x) & = & a\Delta_x\vartheta(t,x) & \text{in} & (0,T] \times \Omega(s) \\ \vartheta(0,x) & = & \vartheta_0(x) & \text{in} & \Omega(s) \\ \dfrac{\partial\vartheta}{\partial n}(t,x) & = & 0 & \text{on} & (0,T] \times (\Gamma_1 \cup \Gamma_2) \\ \dfrac{\partial\vartheta}{\partial n}(t,x) & = & \alpha(\vartheta_s(x) - \vartheta(t,x)) & \text{on} & (0,T] \times \Gamma_3, \end{array} \tag{2.6}$$

where ϑ_0 and ϑ_s are the (given) initial temperature and the temperature of the surrounding medium at $\Gamma_3(s)$, respectively, and $a = \frac{\kappa}{c\rho}$, $\alpha = \frac{\alpha_s}{\kappa}$. The positive constants ρ, c, κ, α_s stand for mass density, specific heat, heat conductivity and heat exchange coefficient, in that order.

Let $\tau_{ij} = \tau_{ij}(u(t,x))$ and $\varepsilon_{ij} = \varepsilon_{ij}(u(t,x))$ denote the components of the *stress* and (linearized) *strain tensors*, respectively, where $u = (u^1(t,x), u^2(t,x))$ is the *displacement vector*. Assuming a linear thermoelastic behaviour, we obtain for the quasistatic regime

$$-\tfrac{E}{2(1+\nu)}\left(\Delta u + \left(\tfrac{1}{1-2\nu}\right)\mathbf{grad}\ (\mathbf{div}\ u)\right) \;=\; \rho F - \tfrac{E}{\nu}\alpha_{el}\,\mathbf{grad}\,\vartheta \qquad \text{in}\ \ \Omega(s), \tag{2.7}$$

where E, ν and α_{el} are the Young modulus, the Poisson ratio and the linear heat stretch coefficient, respectively. F is the body force.

We impose the boundary conditions

$$\begin{aligned} u &= 0 \text{ on } \Gamma_1 \\ u^1 &= 0 \text{ on } \Gamma_2 \\ \tau_{12} &= 0 \text{ on } \Gamma_2 \\ \tau_{11}n^1 + \tau_{12}n^2 &= 0 \text{ on } \Gamma_3(s) \\ \tau_{21}n^1 + \tau_{22}n^2 &= 0 \text{ on } \Gamma_3(s), \end{aligned} \tag{2.8}$$

where $n = (n^1, n^2)$ is the outward unit normal vector on $\Gamma_3(s)$. It is convenient to introduce the Lamé constants $\mu = \frac{E}{2(1+\nu)}$, $\lambda = \frac{\nu E}{(1+\nu)(1-2\nu)}$ and the coefficient $\beta = \frac{E}{\nu}\alpha_{el}$.

Let $\hat{s} = \hat{s}(x_1)$ denote the upper boundary of the thermally deformed domain, which should be approximated as closely as possible at the final time T. Assuming that the vertical displacement u^2 dominates the horizontal one, the following optimal shape design problem is considered:

$$\textbf{(P)}\ \begin{cases} \text{Minimize} \\ \qquad \displaystyle\int_0^d \left(s(x_1) + u^2(T, x_1, s(x_1)) - \hat{s}(x_1)\right)^2 dx_1 \qquad (2.9) \\ \text{subject to } s \in U^{ad} \text{ and to the equations (2.6)–(2.8).} \end{cases}$$

As only the final displacement $u(T,x)$ plays a role in (2.9), we write in the sequel $u(x) := u(T,x)$. This displacement u is obtained from (2.7), where $\vartheta = \vartheta(T,x)$ is inserted in the right–hand side.

We also introduce the abbreviating notations

$$\begin{aligned} (u,v)_\Omega &= \int_\Omega u^T(x)v(x)\,dx, \\ (u,v)_\Gamma &= \int_\Gamma u^T(x)v(x)\,dS_x. \end{aligned}$$

Then the weak formulation of the state equations (2.6)–(2.8) reads

$$(\vartheta_t(t), w)_{\Omega(s)} + a\ (\nabla\vartheta(t), \nabla w)_{\Omega(s)} + \alpha\ (\vartheta(t), w)_{\Gamma_3(s)} = \alpha\ (\vartheta_s(t), w)_{\Gamma_3(s)}, \qquad \text{for all } w \in H^1(\Omega(s)) \text{ and a.e. } t \in (0,T). \tag{2.10}$$

$$\vartheta(0) = \vartheta_0, \tag{2.11}$$

$$(\tau_{ij}(u(x)), \varepsilon_{ij}(v(x)))_{\Omega(s)} = (\rho F, v)_{\Omega(s)}, \tag{2.12}$$

for all $v \in V(s)$, where $V(s) = \{v \in (H^1(\Omega(s))^2 : v = 0 \text{ on } \Gamma_1,\ v^1 = 0 \text{ on } \Gamma_2\}$, and where the summation convention is used.

The quantities ϑ, u, τ, and ε are connected through the relations

$$\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial u^i}{\partial x_j} + \frac{\partial u^j}{\partial x_i}\right), \tag{2.13}$$

$$\begin{pmatrix} \tau_{11} \\ \tau_{22} \\ \tau_{12} \end{pmatrix} = \begin{pmatrix} 2\mu+\lambda & \lambda & 0 \\ \lambda & 2\mu+\lambda & 0 \\ 0 & 0 & 2\mu \end{pmatrix} \begin{pmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{12} \end{pmatrix} - \beta\vartheta \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}. \tag{2.14}$$

From several papers it is known, that under the assumptions made here a unique solution (ϑ, u) to the weak thermoelasticity system (2.10)–(2.12) exists. In [15] the existence of at least one optimal shape function was shown under the conditions $\vartheta_0 \in H^2(\Omega(s))$, $\vartheta_s = \vartheta_s(t, x_1)$, $\vartheta_s \in C^1([0,T], L_2(0,d))$, $F = F(x_1)$, $F \in L_2(0,d)$, and a compatibility condition between ϑ_0 and $\vartheta_s(0)$.

3 Discretization of the problem

We consider an approximation of the boundary $\Gamma_3(s)$ by linear B–splines. Let the interval $[0,d]$ be divided into $m-1$ subintervals $[x_i, x_{i+1}]$, $i = 1,\dots,m-1$ and let $z_i = s(x_i)$, $i = 1,\dots,m$ denote the values of the piecewise linear shape function in the nodal point x_i. Beginning from a given vector $z \in I\!R^m$, $z = (z_k)_{k=1,\dots,m}$ the shape function is

$$s(x) = \sum_{i=1}^{m} z_i B_i(x), \tag{3.1}$$

where $B_i(x)$, $i = 2,\dots,m-1$, is the linear B–spline which equals one at x_i, zero at x_{i-1}, x_{i+1}, and where $B_1(x)$, $B_m(x)$ are the B–splines with known modification for x_1, x_m, respectively. Now the vector z stands for our varying shape.

Let for convenience $\Omega(z)$, $\Gamma_3(z)$ denote the domain $\Omega(s)$ and $\Gamma_3(s)$ with shape function s given in the form (3.1), h be the mesh discretization parameter, $H^1_h(\Omega(z))$ be an n–dimensional subspace of $H^1(\Omega(z))$ and $V_h(z)$ an N–dimensional subspace of $V(z)$.

We are looking for a semidiscrete approximation of the solution of (2.10) –(2.11) in $C^1[0,T;H_h^1(\Omega(z))]$. Let $\{\varphi_i\}_1^n$ be a basis of $H_h^1(\Omega(z))$. Then the desired solution can be described as

$$\vartheta_h(t,x) = \sum_{i=1}^n \vartheta_i(t)\varphi_i(x),$$

where $\vartheta_i \in C^1[0,T]$, $i = 1,\ldots,n$. Inserting in (2.10)–(2.11), we get for $i = 1,\ldots,n$

$$\begin{aligned}
(\frac{d}{dt}\sum_{j=1}^n \vartheta_j(t)\varphi_j,\varphi_i)_{\Omega(z)} &+ a\ (\sum_{j=1}^n \vartheta_j(t)\nabla\varphi_j,\nabla\varphi_i)_{\Omega(z)} \\
&+ \alpha\ (\sum_{j=1}^n \vartheta_j(t)\varphi_j,\varphi_i)_{\Gamma_3(z)} \\
&= \alpha\ (\vartheta_s,\varphi_i)_{\Gamma_3(z)} \\
\sum_{j=1}^n \vartheta_j(0)(\varphi_j,\varphi_i)_{\Omega(z)} &= (\vartheta_0,\varphi_i)_{\Omega(z)},
\end{aligned}$$

e.g. the system of ordinary differential equations

$$\begin{aligned}
M(z)\vartheta'(t) + K(z)\vartheta(t) &= g(z) \\
M(z)\vartheta(0) &= \theta(z) \qquad (3.2)
\end{aligned}$$

with $M(z) = (m_{ij}(z))_{i,j=1,\ldots,n}$, $K(z) = (k_{ij}(z))_{i,j=1,\ldots,n}$, $g(z) = (g_i(z))_{i=1,\ldots,n}$ and $\theta(z) = (\theta_i(z))_{i=1,\ldots,n}$,

$$\begin{aligned}
k_{ij}(z) &= a\ (\nabla\varphi_j,\nabla\varphi_i)_{\Omega(z)} + \alpha\ (\varphi_j,\varphi_i)_{\Gamma_3(z)} \qquad (3.3) \\
m_{ij}(z) &= (\varphi_j,\varphi_i)_{\Omega(z)} \qquad (3.4) \\
g_i(z) &= \alpha\ (\vartheta_s,\varphi_i)_{\Gamma_3(z)} \qquad (3.5) \\
\theta_i(z) &= (\vartheta_0,\varphi_i)_{\Omega(z)}, \qquad (3.6)
\end{aligned}$$

where $i,j = 1,\ldots,n$ and $\vartheta = (\vartheta_i)_{i=1,\ldots,n}$.

Next, we analogously consider the space $V_h(z)$ mentioned above with basis $\{\psi_i\}_{i=1}^N$. The solution of (2.12) is represented by $u_h(x) = \sum\limits_{i=1}^N u_i\psi_i(x)$. We have to mention that $u_h(x)$ and ψ_i consist of two components. We define $\psi_{2i-1} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}\varphi_i$ and $\psi_{2i} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}\varphi_i$, $u_i \in I\!R$, $i = 1,\ldots,n$. Moreover, we put $\psi_{2i-1} = \psi_{2i} = 0$, if $x_i \in \Gamma_1$, $\psi_{2i-1} = 0$, if $x_i \in \Gamma_2$, and renumber all nontrivial ψ_i. In this way, we arrive at the basis $\{\psi_i\}_{i=1}^N$.

After some transformations in relation (2.12) we arrive at the system of linear equations

$$A(z)u = f(\vartheta_h(T),z) \qquad (3.7)$$

with $A(z) = (a_{ij}(z))_{i,j=1,\dots,N}$, $f(\vartheta, z) = (f_i(\vartheta, z))_{i=1,\dots,N}$,

$$\begin{aligned} a_{ij}(z) &= \lambda(\mathbf{div}\ \psi_i, \mathbf{div}\ \psi_j)_{\Omega(z)} + 2\mu \sum_{k,l=1}^{2} (\varepsilon_{kl}(\psi_i), \varepsilon_{kl}(\psi_j))_{\Omega(z)} \\ f_i(\vartheta_h, z) &= (\rho F, \psi_i)_{\Omega(z)} + (\beta\vartheta_h, \mathbf{div}\ \psi_i)_{\Omega(z)}, \end{aligned}$$

$i, j = 1, \dots, N$. Expanding $\vartheta_h(x) = \sum\limits_{j=1}^{n} \vartheta_j \varphi_j(x)$ we get

$$f_i(\vartheta_h, z) = \rho F_i(z) + \beta \sum_{j=1}^{n} b_{ij}(z)\vartheta_j, \tag{3.8}$$

where

$$\begin{aligned} F_i &= (F, \psi_i)_{\Omega(z)} && (3.9) \\ b_{ij} &= (\mathbf{div}\ \psi_i, \varphi_j)_{\Omega(z)}, && (3.10) \end{aligned}$$

$i = 1, \dots, N$, $j = 1, \dots, n$. After introducing the vector $F_h = (F_i)_{i=1,\dots,N}$ and the $(N \times n)$ – matrix $B = B(z) = (b_{ij})_{\substack{\mathrm{i = 1,\dots,N} \\ \mathrm{j = 1,\dots,n}}}$, we find

$$f(\vartheta_h(T), z) = \rho F_h(z) + \beta B(z)\vartheta(T), \tag{3.11}$$

where $\vartheta(T) = (\vartheta_i(T))_{i=1,\dots,n}$.

Our problem of thermoelasticity can be solved approximately by the system of ordinary differential equations (3.2) and the system of linear equations (3.7).

In addition, we arrive at the discrete objective functional,

$$\Phi(z, u) = \int_0^d \left(\sum_{i=1}^{m} [z_i + u_{\pi(i)} - \hat{z}_i] B_i(x) \right)^2 dx, \tag{3.12}$$

where $\pi : \{1, \dots, N\} \to \{1, \dots, N\}$ is a permutation of all indexes such that $u_{\pi(i)}$, $i = 1, \dots, m$, are the values of the second component u^2 of the displacement vector in the node points on the boundary $\Gamma_3(z)$.

4 Numerical approach for solving the problem

4.1 A Gradient method for solving the programming problem

We have to solve numerically the nonlinear programming problem

$$(P_h)\begin{cases}\text{Minimize} & \\ & \Phi(z,u) \\ \text{subject to } z \in U_h^{ad} \text{ and to the equations (3.2), (3.7),} & \end{cases} \tag{4.1}$$

where

$$\begin{aligned} U_h^{ad} &= \{z \in I\!\!R^m : 0 < \underline{s} \le z_i \le \overline{s},\ i = 1,\dots,m, \\ & \quad |z_{i+1} - z_i| \le h\overline{s}',\ i = 1,\dots,m, \\ & \quad \underline{s} \le z_1 \le \overline{s},\ \underline{s} \le z_m \le \overline{s}\}. \end{aligned}$$

Thus, we have set $r = 1$ (Lipschitz continuity of s') for numerical reasons, in order to obtain linear constraints. Let us denote by T the operator assigning $(u_{\pi(i)})_{i=1,\dots,m}$ to z (according to (3.2), (3.7)) . It is a well–known fact of numerical analysis that a unique solution of (3.2), (3.7) exists for all $z \in U_h^{ad}$. Let a certain (arbitrary) numerical routine be given, which realizes this solution. Then we can handle the operator T numerically.

We reformulate the programming problem as

$$(P_h)\begin{cases} F(z) = \Phi(z,T(z)) = \min! & (4.2) \\ z \in U_h^{ad}. & (4.3) \end{cases}$$

The operator T is defined through the equations (3.2), (3.7). Therefore, T is differentiable from $I\!\!R^m$ to $I\!\!R^m$ provided that the matrices $M(z)$ and $A(z)$ are nonsingular and all entries of M, A, K, g, f, are differentiable with respect to z. This property is satisfied on sufficiently regular meshes. Let us assume that we are able to determine somehow the derivative of T with respect to z in the direction v. Then the derivative of F in the direction v is

$$F'(z)v = \langle \Phi_z(z,T(z)), T'(z)v\rangle_{R^m}, \tag{4.4}$$

thus

$$\mathbf{grad}\ F(z) = T'(z)^*\Phi_z(z,T(z)) \tag{4.5}$$

(by stars we denote transposition). Thanks to this result, we can formulate a gradient method for solving the problem (4.2)–(4.3).

1. Let k iterations be executed with the result $z^k \in U_h^{ad}$. Choose an arbitrary stopping parameter $\varepsilon > 0$.

2. Determine $T'(z^k)$.

3. Determine **grad** $F(z^k)$ to find a direction of descent,

4. solve the linear programming problem

$$h_k = \arg \min_{z^k + h \in U_h^{ad}} \langle \mathbf{grad}\ F(z^k), h\rangle_{R^m}.$$

5. Solve the one–dimensional programming problem (line search)

$$\lambda_k = \arg\min_{\lambda\in[0,1]} F(z^k + \lambda h_k).$$

6. Determine the new solution

$$z^{k+1} = z^k + \lambda_k h_k.$$

7. If $\|z^{k+1} - z^k\| > \varepsilon$, let $z_k := z_{k+1}$ and go to 2.

8. Terminate the method at z^{k+1}.

The exact minimization in step 5 is not practicable. In real computations, a step size λ ensuring a decrease of F is to be found. We applied the bisection method (cf. the numerical examples in section 4.5). The calculation of the derivative $T'(z)$ and step 5 are the most time consuming steps.

In the next sections we discuss and compare several ways to determine $T'(z)$.

4.2 Calculation of $T'(z)$

In order to perform the gradient method, we have to calculate the derivative of the operator T with respect to z. However, this operator is given only in implicite form (through the system (3.2), (3.7)) which makes the differentiation of T difficult. The simplest way to approximate T' is to use finite differences.

By $\dfrac{\partial T}{\partial z_k}$ the k–th column of the Jacobian $T'(z)$ is denoted, hence

$$T'(z) = \left(\frac{\partial T}{\partial z_k}\right)_{k=1,\ldots,m}. \tag{4.6}$$

We approximate this by

$$\frac{\partial T}{\partial z_k} \approx \frac{T(z+\varepsilon e_k) - T(z)}{\varepsilon}, \tag{4.7}$$

where ε is a positive number and e_k is the unit vector in the direction of the k–th coordinate axis.

Using this approach we have to solve the system (3.2), (3.7) at $z+\varepsilon e_k$ and at z to calculate one partial derivative of T. Since there are m such partial derivatives, we are forced to solve the full above mentioned system for $m+1$ times. This is time–consuming. On the other hand, the method can be applied using an arbitrary numerical solver of the thermoelasticity system. We do not need to know how the system matrices and vectors are calculated and how they are processed by the solver. The expensive calls of T can be avoided by other techniques which will be explained below.

4.3 The adjoint–equation method

Using the abbreviation $u_\varepsilon = T(z+\varepsilon e_k)$, k fixed, we have

$$\frac{\partial T}{\partial z_k}(z) = \lim_{\varepsilon\to 0}\frac{1}{\varepsilon}(T(z+\varepsilon e_k) - T(z)) = \lim_{\varepsilon\to 0}\frac{1}{\varepsilon}(u_\varepsilon - u) =: \delta u.$$

The equations defininig u and u_ε read

$$\begin{aligned} M(z)\vartheta'(t) + K(z)\vartheta(t) &= g(z) \\ M(z)\vartheta(0) &= \theta(z) \\ A(z)u &= \rho F(z) + \beta B(z)\vartheta(T) \end{aligned} \tag{4.8}$$

and, respectively,

$$\begin{aligned} M(z+\varepsilon e_k)\vartheta_\varepsilon'(t) + K(z+\varepsilon e_k)\vartheta_\varepsilon(t) &= g(z+\varepsilon e_k) \\ M(z+\varepsilon e_k)\vartheta_\varepsilon(0) &= \theta(z+\varepsilon e_k) \\ A(z+\varepsilon e_k)u_\varepsilon &= \rho F(z+\varepsilon e_k) + \beta B(z+\varepsilon e_k)\vartheta_\varepsilon(T). \end{aligned} \tag{4.9}$$

Subtracting (4.8) and (4.9) and dividing by ε, we obtain in the limit

$$M(z)\delta\vartheta'(t) + K(z)\delta\vartheta(t) = \frac{\partial g}{\partial z_k}(z) - \frac{\partial M}{\partial z_k}(z)\vartheta'(t) - \frac{\partial K}{\partial z_k}(z)\vartheta(t) \tag{4.10}$$

$$M(z)\delta\vartheta(0) = \frac{\partial \theta}{\partial z_k} - \frac{\partial M}{\partial z_k}(z)\vartheta(0) \tag{4.11}$$

$$A(z)\delta u = \rho\frac{\partial F}{\partial z_k} + \beta\frac{\partial B}{\partial z_k}\vartheta(T) + \beta B(z)\delta\vartheta(T) - \frac{\partial A}{\partial z_k}(z)u \tag{4.12}$$

the system for δu. In this way, $\dfrac{\partial T}{\partial z_k} = \delta u$ can be determined through the solution to the semidiscrete thermoelastic system (4.10) – (4.12), provided that the derivatives of A, K, M, B, θ, g and f with respect to z_k are available. To find $T'(z)$, (4.10) – (4.12) must be solved for $k = 1,\dots,m$, i.e. for m times.

Next, we modify the descent method of section 4.3. We avoid the explicit use of $T'(z)$ by an adjoint equation. To derive adjoint equation and gradient technique at the same time, the well known *Lagrange method* is applied.

We consider the semidiscrete problem in the equivalent form

$$(\boldsymbol{P}_h)\left\{\begin{aligned} \Phi(z,u) &= \min! && && (4.13) \\ M(z)\vartheta'(t) + K(z)\vartheta(t) &= g(z), && t\in(0,T] && (4.14) \\ M(z)\vartheta(0) &= \theta(z) && && (4.15) \\ A(z)u &= \rho F_h(z) + \beta B(z)\vartheta(T) && && (4.16) \\ z &\in U_h^{ad}. && && (4.17) \end{aligned}\right.$$

We define the Lagrange function for problem (P_h) with Lagrange multipliers $v \in C^1[0,T;\mathbb{R}^n]$, $y \in \mathbb{R}^N$ and $w \in \mathbb{R}^n$ by

$$\begin{aligned} \mathcal{L}(u,z,\vartheta;v,y,w) &= \Phi(z,u) + \langle v, M(z)\vartheta' + K(z)\vartheta - g(z)\rangle_{L_2(0,T;R^n)} \\ &+ \langle y, \rho F_h(z) + \beta B(z)\vartheta(T) - A(z)u\rangle_{R^N} \\ &+ \langle w, M(z)\vartheta(0) - \theta(z)\rangle. \end{aligned} \tag{4.18}$$

An optimal solution (u^0,z^0,ϑ^0) of (P_h) and the associated *adjoint state* (v^0,y^0,w^0) fulfil the first order necessary optimality conditions

$$\begin{aligned} \mathcal{L}_u(u^0,z^0,\vartheta^0;v^0,y^0,w^0)u &= 0 \qquad \forall u \in \mathbb{R}^N & (4.19)\\ \mathcal{L}_\vartheta(u^0,z^0,\vartheta^0;v^0,y^0,w^0)\vartheta &= 0 \qquad \forall \vartheta \in C^1[0,T;\mathbb{R}^n] & (4.20)\\ \langle \mathcal{L}_z(u^0,z^0,\vartheta^0;v^0,y^0,w^0), z - z^0\rangle &\geq 0 \qquad \forall z \in U_h^{ad}. & (4.21) \end{aligned}$$

The evaluation of (4.19) gives

$$A^*(z^0)y^0 = \Phi_u(z^0,u^0) \tag{4.22}$$

From (4.20) it follows that

$$\begin{aligned} w^0 &= v^0(0) & (4.23)\\ -M^*(z^0)(v^0)'(t) + K^*(z^0)v^0(t) &= 0, \qquad t \in [0,T), & (4.24)\\ -M^*(z^0)v^0(T) &= \beta B^*(z^0)y^0. & (4.25) \end{aligned}$$

Moreover, (4.21) implies

$$\begin{aligned} \sum_{i=1}^m \langle \frac{\partial \Phi}{\partial z_i} &+ \int_0^T -\langle (v^0)'(t), \frac{\partial M}{\partial z_i}(\vartheta^0)(t)\rangle + \langle v^0(t), \frac{\partial K}{\partial z_i}\vartheta^0(t) - \frac{\partial g}{\partial z_i}\rangle\, dt \\ &+ \langle y^0, \rho\frac{\partial F}{\partial z_i} + \beta\frac{\partial B}{\partial z_i}\vartheta^0(T) - \frac{\partial A}{\partial z_i}u^0\rangle - \langle w^0, \frac{\partial \theta}{\partial z_i}\rangle + \langle v^0(T), \frac{\partial M}{\partial z_i}\vartheta^0(T)\rangle, h_i\rangle \\ &\geq 0 \qquad\qquad \forall h \in U_h^{ad} - \{z^0\}, \end{aligned}$$

where the derivatives are taken at z^0.

Now, we are able to modify our gradient method. Let (z^k,u^k,ϑ^k) be an admissible triple of (P_h) and $\varepsilon > 0$ an arbitrary number.

1. Determine v^k and y^k by solving

$$A^*(z^k)y = \Phi_u(z^k,u^k)$$

and

$$\begin{aligned} -M^*(z^k)v'(t) + K^*(z^k)v(t) &= 0 \qquad t \in [0,T) \\ -M^*(z^k)v(T) &= \beta B^*(z^k)y^k. \end{aligned}$$

2. Compute $c \in I\!R^m$ from

$$\begin{aligned} c_i &= \frac{\partial \Phi}{\partial z_i} + \int\limits_0^T -\langle (v^k)'(t), \frac{\partial M}{\partial z_i}(\vartheta^k)(t)\rangle + \langle v^k(t), \frac{\partial K}{\partial z_i}\vartheta^k(t) - \frac{\partial g}{\partial z_i}\rangle \, dt \\ &+ \langle y^k, \rho\frac{\partial F}{\partial z_i} + \beta\frac{\partial B}{\partial z_i}\vartheta^k(T) - \frac{\partial A}{\partial z_i}u^k\rangle - \langle w^k, \frac{\partial \theta}{\partial z_i}\rangle + \langle v^k(T), \frac{\partial M}{\partial z_i}\vartheta^k(T)\rangle, \end{aligned}$$

   $i = 1, \ldots, m.$

3. Solve the linear programming problem

$$h^k = \arg\min\{\langle c, h\rangle | z^k + h \in U^{ad}\}.$$

   If $\langle c, h^k\rangle \geq 0$, then z^k is an optimal solution of (P_h) and the method is terminated. In the other case we

4. determine

$$\lambda^k = \arg\min\{\Phi(z^k + \lambda h^k, u(z^k + \lambda h^k)) | \lambda \in [0,1]\}.$$

   If $\lambda_k < \varepsilon$, then the process ends, else $z^{k+1} = z^k + \lambda^k h^k$, put $z^k := z^{k+1}$, go to 1.

Using the adjoint equation method we have to solve the whole adjoint system of thermoelasticity only once for every calculation of a descent direction, while any explicit use of $T'(z)$ requires the evaluation of the (primal) system of thermoelasticity $m+1$ times. However, we must pay for that. We have to calculate derivatives of various matrices and vectors with respect to z. This last procedure is not so expensive as the direct calculation of the derivative of T.

4.4 Calculation of matrix derivatives

In the gradient algorithm described in the previous subsection certain derivatives of matrices or vectors with respect to z occur. We have to calculate them. The first idea is to use finite difference approximations of these derivatives again. The second one is to calculate them analytically.

4.4.1 Approximation by finite differences

The derivatives of the matrices and vectors M, K, A, B, g, F and θ with respect to z can be approximated in the same way as in (4.7). For instance, the partial derivative of the stiffness matrix K can with sufficient accuracy be determined by

$$\frac{\partial K}{\partial z_k} \approx \frac{1}{\varepsilon}(K(z + \varepsilon e_k) - K(z)) \qquad (4.26)$$

or any other scheme of numerical differentiation. This method can be recommended, if the computation of these matrices is easy to handle (for instance accessible from the FE–method).

4.4.2 Analytical calculation of derivatives

We suppose that, for a given boundary $\Gamma_3(z)$, the algorithm constructing the grid of the whole domain $\Omega(z)$ is known in advance. This assumption is not fulfilled, if a multigrid method with automatic grid generation underlies the FE–technique. Then the derivatives of the system matrices and vectors can be calculated analytically. Here, we describe this method by an example.

Let z (the boundary vector) be given, e.g., let the mesh points on $\Gamma_3(z)$ have the coordinates (x_i, z_i), $i = 1, \ldots, m$. We construct the remaining points of the mesh in the following way. Let $K \in I\!N$, $l_i = \frac{z_i}{K}$, $h_x = \frac{d}{m-1}$, $h_y = \frac{1}{K}$ and $r_i := \frac{l_i}{h}$ $(i = 1, \ldots, m)$. Then we define grid points by $(x_i, l_i \cdot j)$, $i = 1, \ldots, m$, $j = 1, \ldots, K$. Connecting $(x_i, l_i \cdot j)$, $j = 1, \ldots, K$, for every $i = 1, \ldots, m$, and $(x_i, l_i \cdot j)$, $i = 1, \ldots, m$, for every $j = 1, \ldots, K$, we obtain a set of rectangular elements. Dividing each rectangle by two diagonals, we arrive at the set of our finite elements (triangles) $\{\tau_p\}$, $p = 1, \ldots, D$, where D denotes the number of elements. In Figure 2 a grid for $K = m = 11$ is shown.

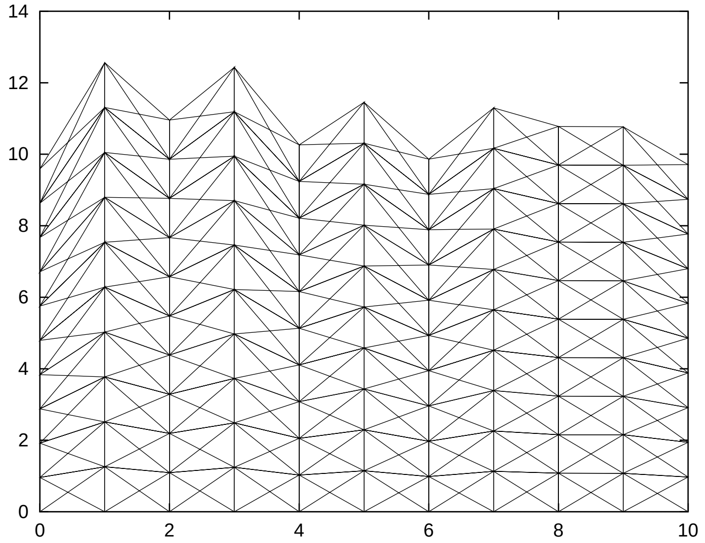

Figure 2: FEM–Grid

This diagonalization was used for symmetry to enhance better numerical stability for the solution of the displacement u.

The system matrices are sums of the element matrices. For instance, $K = \sum\limits_{p=1}^{D} K^p$, where K^p is the $(n \times n)$–element matrix belonging to the triangle with number p. Let i_1^p, i_2^p, i_3^p denote the numbers of mesh points forming this triangle, $I_p = \{i_1^p, i_2^p, i_3^p\}$. Then $k_{ij}^p = 0$, if $i \notin I_p$ or $j \notin I_p$. The nine nontrivial entries of K^p are

$$k_{i_l i_q}^p =: b_{lq}^p,$$

$l, q = 1, 2, 3$.

We obtain

$$
\begin{aligned}
\frac{\partial K}{\partial z_k} &= \sum_{p=1}^{D} \frac{\partial K^p}{\partial z_k} \\
\frac{\partial K^p}{\partial z_k} &\sim \left(\frac{\partial b^p_{lq}}{\partial z_k}\right)_{l,q=1,2,3} \qquad (4.27)\\
\frac{\partial b^p_{lq}}{\partial z_k} &= \sum_{j=1}^{3} \left(\frac{\partial b^p_{lq}}{\partial x_j}\frac{\partial x_j}{\partial z_k} + \frac{\partial b^p_{lq}}{\partial y_j}\frac{\partial y_j}{\partial z_k}\right), \qquad (4.28)
\end{aligned}
$$

where $l, q = 1, 2, 3$. $\left(b^p_{lq}\right)_{l,q=1,2,3}$ $p = 1, \ldots, D$ are the element matrices, which can be determined as functions of the coordinates of the associated mesh points forming the underlying triangle,

$$b^p_{ij} = \varphi^p(x_1, x_2, x_3, y_1, y_2, y_3).$$

These coordinates change if z varies . Hence, we need to find the partial derivatives of φ with respect to x_i, y_i, $i = 1, 2, 3$, as well as the partial derivatives of the node coordinates with respect to z_k, $k = 1, \ldots, m$.

φ is a rational function of the node coordinates. We get these formulas by calculating k_{ij} from (3.3), integrating exactly or applying a certain quadrature formula. The partial derivatives of this rational function can be easily obtained if a routine of symbolic calculation is used.

Next, we have to determine the derivatives of the node coordinates with respect to z.

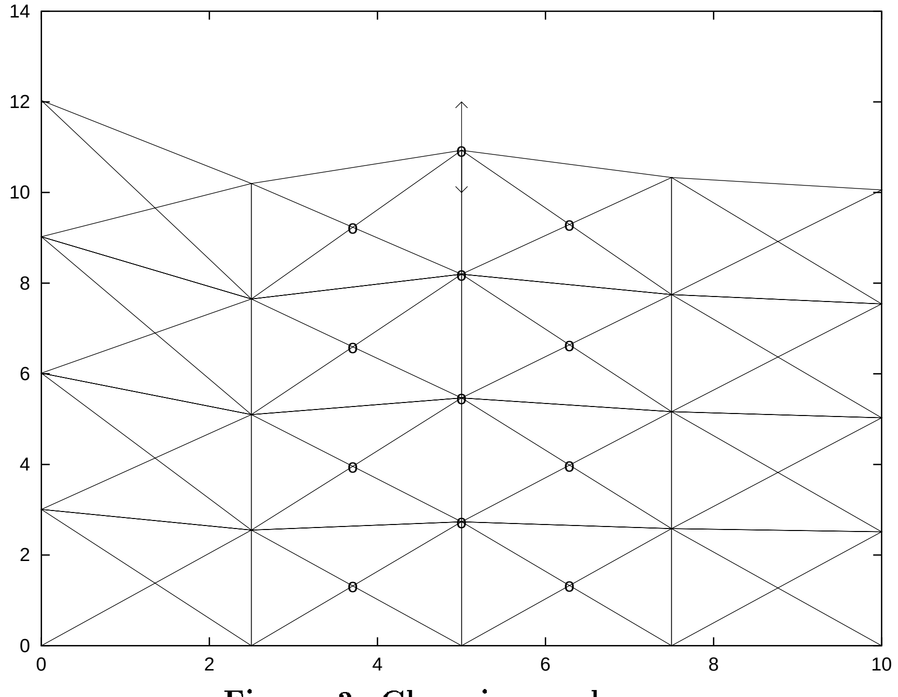

Figure 3: Changing nodes

If z changes in direction e_k, then the nodes with coordinates (x_k, y), $y > 0$ vary. Furthermore, all nodes on diagonals of quadrangles, having one corner at nodes with coordinates (x_k, y) (y arbitrary), are modified. In Figure 3 we see the changing nodes for $k = 3$ and $m = 5$.

Let a quadrangle with the corners (x_a, a), (x_b, b), (x_b, c), (x_a, d) be given , where these points denote the lower left, lower right, upper right, upper left corners, in that order. Then the x–coordinate of the intersection of the two diagonals is

$$x = \xi(x_a, x_b, a, b, c, d) = \frac{x_a(b - c) + x_b(a - d)}{a + b - c - d}, \tag{4.29}$$

while the y–coordinate is

$$y = \eta(a, b, c, d) = \frac{ab - cd}{a + b - c - d}. \tag{4.30}$$

We regard one part of the grid essential for understanding the calculation of the desired derivatives. The nodes are numbered as indicated in Fig. 4. We assume that the midpoint (x_5, y_5) has the coordinates (x_k, il_k).

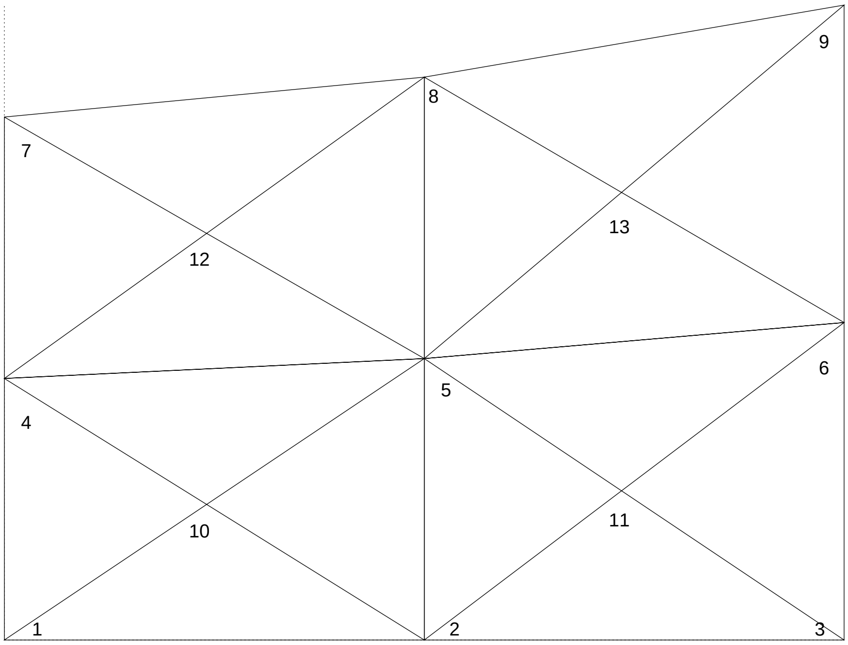

Figure 4: Grid part

Then the other nodes have the coordinates listed on in Table 1.

node	x–coordinate	y–coordinate
1	x_{k-1}	$(i-1)l_{k-1}$
2	x_k	$(i-1)l_k$
3	x_{k+1}	$(i-1)l_{k+1}$
4	x_{k-1}	il_{k-1}
5	x_k	il_k
6	x_{k+1}	il_{k+1}
7	x_{k-1}	$(i+1)l_{k-1}$
8	x_k	$(i+1)l_k$
9	x_{k+1}	$(i+1)l_{k+1}$
10	$\xi(x_{k-1},x_k,(i-1)l_{k-1},(i-1)l_k,il_k,il_{k-1})$	$\eta((i-1)l_{k-1},(i-1)l_k,il_k,il_{k-1})$
11	$\xi(x_k,x_{k+1},(i-1)l_k,(i-1)l_{k+1},il_{k+1},il_k)$	$\eta((i-1)l_k,(i-1)l_{k+1},il_{k+1},il_k)$
12	$\xi(x_k,x_{k+1},il_k,il_{k+1},(i+1)l_{k+1},(i+1)l_k)$	$\eta(il_k,il_{k+1},(i+1)l_{k+1},(i+1)l_k)$
13	$\xi(x_{k-1},x_k,il_{k-1},il_k,(i+1)l_k,(i+1)l_{k-1})$	$\eta(il_{k-1},il_k,(i+1)l_k,(i+1)l_{k-1})$

Table 1: Node coordinates

Now, the coordinates of the perturbed nodes associated to $z+\varepsilon e_k$ are easy to express (see Table 2).

node	x–coordinate	y–coordinate
1	x_{k-1}	$(i-1)l_{k-1}$
2	x_k	$(i-1)l_k^\varepsilon$
3	x_{k+1}	$(i-1)l_{k+1}$
4	x_{k-1}	il_{k-1}
5	x_k	il_k^ε
6	x_{k+1}	il_{k+1}
7	x_{k-1}	$(i+1)l_{k-1}$
8	x_k	$(i+1)l_k^\varepsilon$
9	x_{k+1}	$(i+1)l_{k+1}$
10	$\xi(x_{k-1},x_k,(i-1)l_{k-1},(i-1)l_k^\varepsilon,il_k^\varepsilon,il_{k-1})$	$\eta((i-1)l_{k-1},(i-1)l_k^\varepsilon,il_k^\varepsilon,il_{k-1})$
11	$\xi(x_k,x_{k+1},(i-1)l_k^\varepsilon,(i-1)l_{k+1},il_{k+1},il_k^\varepsilon)$	$\eta((i-1)l_k^\varepsilon,(i-1)l_{k+1},il_{k+1},il_k^\varepsilon)$
12	$\xi(x_k,x_{k+1},il_k^\varepsilon,il_{k+1},(i+1)l_{k+1},(i+1)l_k^\varepsilon)$	$\eta(il_k^\varepsilon,il_{k+1},(i+1)l_{k+1},(i+1)l_k^\varepsilon)$
13	$\xi(x_{k-1},x_k,il_{k-1},il_k^\varepsilon,(i+1)l_k^\varepsilon,(i+1)l_{k-1})$	$\eta(il_{k-1},il_k^\varepsilon,(i+1)l_k^\varepsilon,(i+1)l_{k-1})$

Table 2: Perturbed node coordinates

Here we denote by l_k^ε the term $(1+\frac{\varepsilon}{z_k})l_k$.

In this way the formulas for all nodes with respect to z and to $z+\varepsilon e_k$ are written down and the derivatives can be calculated.

1. Varying nodes with coordinates (x_k, il_k), $k = 1, \dots, m$, $i = 0, \dots, K$:

$$\begin{align}
x_p(z + \varepsilon e_k) &= (k-1)h \nonumber \\
y_p(z + \varepsilon e_k) &= il_k^\varepsilon \nonumber \\
\frac{\partial x_p}{\partial z_k} &= 0 \tag{4.31} \\
\frac{\partial y_p}{\partial z_k} &= ih_y \tag{4.32}
\end{align}$$

2. Varying nodes at the diagonals right from (x_5, y_5):

$$(\xi(x_k, x_{k+1}, il_k, il_{k+1}, (i+1)l_{k+1}, (i+1)l_k), \eta((i-1)l_{k+1}, (i-1)l_k, il_k, il_{k+1})),$$

$k = 1, \dots, m-1$, $i = 0, \dots, K-1$:

$$\begin{align}
x_p(z + \varepsilon e_k) &= \frac{x_k(il_{k+1} - (i+1)l_{k+1}) + x_{k+1}(il_k^\varepsilon - (i+1)l_k^\varepsilon)}{il_k^\varepsilon + il_{k+1} - (i+1)l_{k+1} - (i+1)l_k^\varepsilon} \nonumber \\
&= \frac{x_k l_{k+1} + x_{k+1} l_k^\varepsilon}{l_{k+1} + l_k^\varepsilon} \nonumber \\
y_p(z + \varepsilon e_k) &= \frac{il_k^\varepsilon il_{k+1} - (i+1)l_{k+1}(i+1)l_k^\varepsilon}{il_k^\varepsilon + il_{k+1} - (i+1)l_{k+1} - (i+1)l_k^\varepsilon} \nonumber \\
&= \frac{l_k^\varepsilon l_{k+1}(2i+1)}{l_k^\varepsilon l_{k+1}} \nonumber \\
\frac{\partial x_p}{\partial z_k} &= \frac{h_y h_x l_{k+1}}{(l_k + l_{k+1})^2} \tag{4.33} \\
\frac{\partial y_p}{\partial z_k} &= \frac{h_y l_{k+1}^2 (2i+1)}{(l_k + l_{k+1})^2} \tag{4.34}
\end{align}$$

3. Varying nodes at the diagonals left from (x_5, y_5) :

$$(\xi(x_{k-1}, x_k, il_{k-1}, il_k, (i+1)l_k, (i+1)l_{k-1}), \eta((i-1)l_k, (i-1)l_{k-1}, il_{k-1}, il_k)),$$

$k = 2, \dots, m$, $i = 0, \dots, K-1$:

$$\begin{align}
x_p(z + \varepsilon e_k) &= \frac{x_{k-1}(il_k^\varepsilon - (i+1)l_k^\varepsilon) + x_k(il_{k-1} - (i+1)l_{k-1})}{il_{k-1} + il_k^\varepsilon - (i+1)l_k^\varepsilon - (i+1)l_{k-1}} \nonumber \\
&= \frac{x_{k-1} l_k^\varepsilon + x_k l_{k-1}}{l_k^\varepsilon + l_{k-1}} \nonumber \\
y_p(z + \varepsilon e_{k-1}) &= \frac{il_{k-1} il_k^\varepsilon - (i+1)l_k^\varepsilon(i+1)l_{k-1}}{il_{k-1} + il_k^\varepsilon - (i+1)l_k^\varepsilon - (i+1)l_{k-1}} \nonumber \\
&= \frac{l_{k-1} l_k^\varepsilon (2i+1)}{l_{k-1} l_k^\varepsilon} \nonumber \\
\frac{\partial x_p}{\partial z_k} &= -\frac{h_y h_x l_{k-1}}{(l_k + l_{k-1})^2} \tag{4.35} \\
\frac{\partial y_p}{\partial z_k} &= \frac{h_y l_{k-1}^2 (2i+1)}{(l_k + l_{k-1})^2} \tag{4.36}
\end{align}$$

4.5 Numerical examples

The numerical tests below were perfomed for the following set of parameters:

$$\begin{aligned}
d &= 10 \text{ cm}\\
\underline{s} &= 5 \text{ cm}\\
\overline{s} &= 15 \text{ cm}\\
\overline{s}' &= 2\\
s_a = s_b &= 10 \text{ cm}\\
\vartheta_s(x) &= 727 \text{ e}^{-\frac{(5-x)^2}{9}}\text{K}\\
\kappa &= 0.18 \text{ cal cm}^{-1}\text{K}^{-1}\\
c &= 0.11 \text{ cal g}^{-1} \text{ K}\\
\rho &= 7.5 \text{ g cm}^{-3}\\
\alpha_w &= 0.12 \text{ cal cm}^{-2} \text{ s}^{-1} \text{ K}^{-1}\\
E &= 216000 \text{ g cm}^{-2}\\
\nu &= 0.3\\
\alpha_{el} &= 0.0005 \text{ K}^{-1}.
\end{aligned}$$

ϑ_s is depicted in Fig. 5.

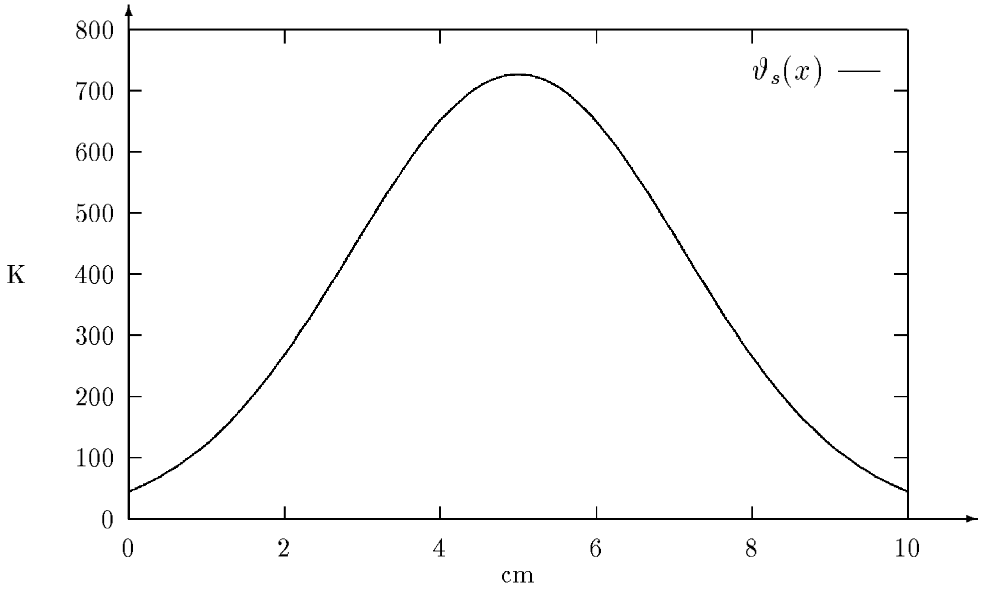

Figure 5: Outside temperature

The desired shape function $\hat{s}$ will be defined later. In our first approach, the derivatives were approximated numerically by the relation (4.7), while a finite element multigrid method was used to solve the system (2.6)–(2.8) of thermoelastic equations.

We used the software package FEMGP developed by the group of Prof. U. Langer (Linz, formally Chemnitz). To approximate $\hat{s}$, the interval $[0, d]$ was divided by 21 nodes. The finest grid generated by the solver FEMGP contained more than 40 000 nodes.

In the gradient method of section 4.1 the direction h_k of descent was determined by solving the associated programming problem by a simplex routine.

The step size λ was calculated in the following way (bisection method):

1. Put $\lambda = 1$.
2. Determine $F(z_k + \lambda h_k)$.
3. While $F(z_k + \lambda h_k) > F(z_k)$ (STEP)
   (a) If $\lambda > \varepsilon$, put $\lambda = \frac{\lambda}{2}$, goto 3
4. Put $\lambda_k = \lambda$.

The desired boundary vector $\hat{z}$ and the initial boundary vector were chosen in the following manner.

Example 1 Let $u^2 = u^2(x_1, s(x_1))$ be the second component of the displacement vector u on Γ_3 at $t = T$, which belongs to the initial shape $s = s(x_1) \equiv 10$. The corresponding deformed shape at $t = T$ is $\hat{s}(x_1) = s(x_1) + u^2(x_1)$. Owing to this, the optimal shape should be $s_{opt}(x_1) \equiv 10$, if this goal function $\hat{s}$ is prescribed in the objective functional (3.12). The optimal value is exactly zero. In the numerical implementation, s was represented by the vector $z = (z_i)$, $z_i = 10$, $i = 1, \ldots, 21$. According to our notations, $\hat{s}$ is realized by $\hat{z} = z + Tz$. One optimal discrete solution must be $z_{opt} \equiv 10$. We started the gradient algorithm at the (very poor) initial guess $z_0 = \hat{z}$. After four iterations a good approximation of the optimal solution was achieved (see Fig. 6 and Fig. 7).

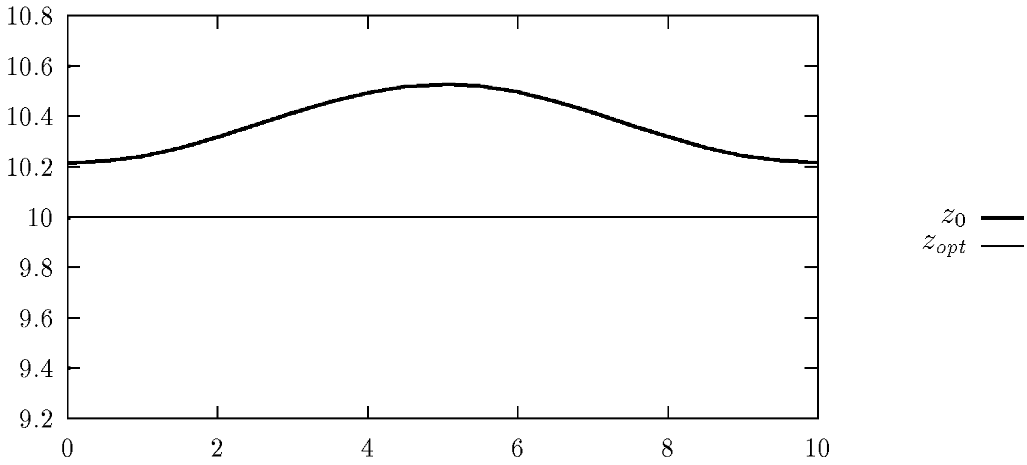

Figure 6: Initial solution

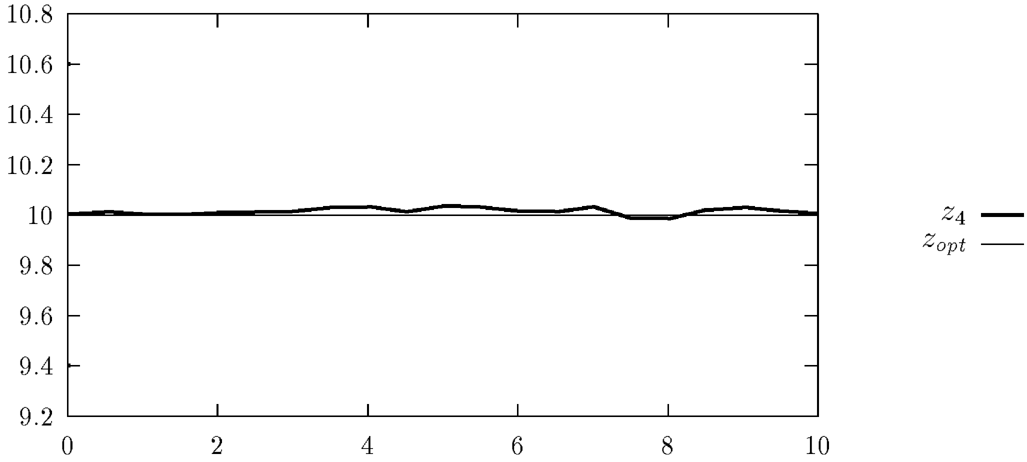

Figure 7: 4–th Iterate

Example 2 Let $\hat{z}$ and, therefore, $z_{opt} \equiv 10$, as before. To obtain a better initial guess, we calculate $T\hat{z}$ and put $z_0 := \hat{z} - T\hat{z}$. We cannot expect that $Tz_0 = T\hat{z}$.However, Tz_0 is certainly very close to $T\hat{z}$, so z_0 should be close to z_{opt}. Therefore, we have a better initial guess as in the first example. The results are represented in Figure 8 and Figure 9.

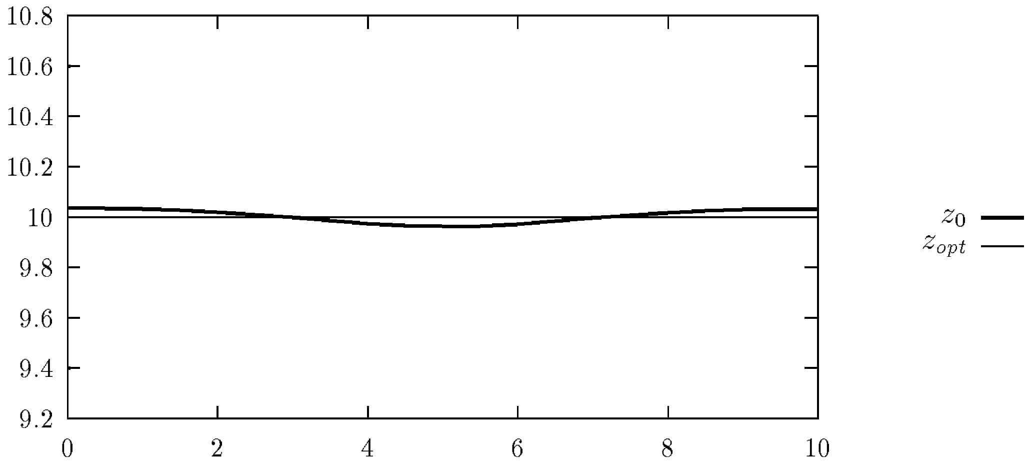

Figure 8: Initial Iteration

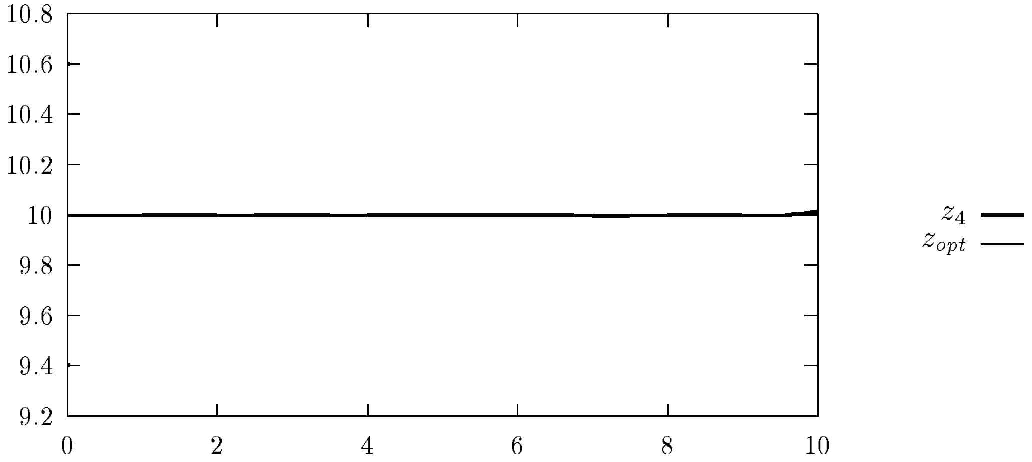

Figure 9: 4–th Iteration

The gradient method using (4.7) worked well for all meaningful test problems. It is easy to implement. Moreover, the very sophisticated FEMGP– solver could be used as a black box. However, the method is time–consuming due to the numerous calls of FEMGP to determine one gradient. The running time to solve the above examples on an HP 9000/730 amounted to some minutes.

Example 3 Now, we investigate a problem without knowledge of its solution. Let $\hat{z} \equiv 10$. The optimal shape is expected to have a deepening centered at $x = 5$. The result after 4 iterations is represented in Figure 10.

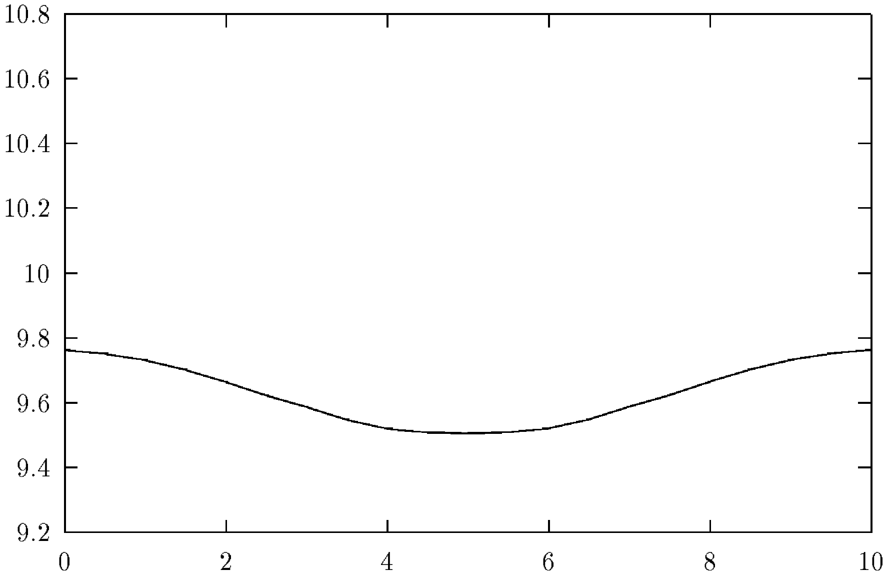

Figure 10: 4-th Iteration

Next, we are going to compare the gradient method and the adjoint equation method. To this end, we used a simple FEM solver for the thermoelastic equations. This solver is much slower than FEMGP, but it is well–suited to perform the comparison. We can easily make the necessary modifications of the system matrices and vectors. Therefore, we have done computations by applying the simple solver to the finite difference method and to the adjoint equation method. In Figure 11, we compare the average difference $\sigma = \frac{1}{m} \sum\limits_{i=1}^{m} |z_{opt} - z_k|$ between the actual solution z_k and the optimal solution z_{opt} for both methods and for different discretization parameters m.

m	k	σ adj. equation method	σ fin. difference method
9	1	0.1478715556	0.1478715556
9	2	0.0349815556	0.1044260000
9	3	0.0181947778	0.0490795556
9	4	0.0344570000	0.0301845556
9	5	0.0268452222	0.0036713333
9	6	0.0003841111	0.0050666667
Time		3:35:57.3	5:03:44.9

m	k	σ adj. equation method	σ fin. difference method
10	1	0.1245350000	0.1245350000
10	2	0.0707486000	0.0082486000
10	3	0.0252120000	0.0079911000
10	4	0.0068256000	0.0077411000
10	5	0.0089058000	0.0076202000
10	6	0.0010242000	0.0075012000
10	7	0.0088280000	0.0081816000
10	8	0.0048874000	0.0081176000
10	9	0.0000022000	0.0002741000
Time		6:53:16.8	9:06:31.7

Figure 11: Comparison

We observe that the iteration process of the two methods develops almost identically.

The essential difference occurs in the calculation time. The running time of the finite difference method is approximately 50 % larger than that of the adjoint equation method. The theoretical assertions stated in section 4.2 are emphasized by the numerical results given here. Our computational experience showed that for large m the adjoint equation method should be preferred.

Moreover, we regarded a slightly modified adjoint equation method. The displacement arising in our model problem occurs mainly at the upper boundary Γ_3. Therefore, it is natural to assume that only node points in a certain neighbourhood of the boundary Γ_3 are moving, while the other part of the grid remains fixed. This approach is often used in the literature. To compare this technique with moving the whole grid, we fixed all node points with exeption of all (x_i, y_i) located at the boundary $\Gamma_3(z)$ or being cross points of the diagonals of the quadrangle Q with $Q \cap \Gamma_3(z) \neq \emptyset$.

We noticed that both methods generated the same defects σ . This is due to the fact that the computed gradients turned out to be almost identical. Even the differences of running times (less than one minute) can be neglected. However, the use of a partial grid may lower the needed storage capacity.

References

[1] R. Butt and Rubio J. E. Optimal shapes for the control of nonlinear heat conduction. *J. Franklin Inst. , No. 1*, 327:151–163, 1990.

[2] T. Cheu. Procedures for shape optimization of fas turbine disks. *Comput. Struct.*, 34 No. 1:1–4, 1990.

[3] M. Delfour, G. Payre, and J.-P. Zolésio. Optimal design of a minimum weight thermal diffuser with constraints on the output thermal power flux. *Appl. Math. Optimization*, 9:225–262, 1983.

[4] K. Dems and Z. Mròz. Variational approach by means of adjoint systems to structural optimization and sensitivity analysis. ii. *Int. J. Solids Struct.*, 20:527–552, 1984.

[5] J. Haslinger and P. Neittaanmäki. *Finite element approximation for optimal shape design: theory and applications.* John Wiley and Sons, Chichester, 1988.

[6] J. Haslinger, P. Neittaanmäki, and K. Salmenjoki. Sensitivity analysis for some optimal shape design problems. *ZAMM*, 67:403–405, 1987.

[7] N. Kamiya and E. Kita. Local shape optimization of a two–dimensional elastic body. *Finite Elem. Anal. Des.*, 6 No. 3:207–216, 1990.

[8] A. Karafiat. On the shape optimization of plane elastic bodies. *Wiss. Z. Tech. Hochsch. Leipz.*, 12 No. 2:131–139, 1988.

[9] R. Mäkinen. On numerical methods for state constrained optimal shape design problems. In *Control and estimation of distributed parameter systems*, volume 91 of *ISNM 4th Int. Conf.*, pages 283–299, Vorau/Austria, 1989.

[10] R. A. Meric. Sensitivity analysis of functionals with respect to shape for dynamically loaded nonlocal thermoelastic solids. *Int. J. Eng. Sci. , No. 26*, 26:703–711, 1988.

[11] A. Myślinski. Finite element approximation of a shape optimization problem for von Kármán system. *Numer. Funct. Anal. Opt.*, 10 No. 7/8:691–717, 1989.

[12] O. Pironneau. *Optimal shape design for elliptic systems.* Springer, New York, 1984.

[13] K. Salmenjoki. *On numerical methods for shape design problems.* PhD thesis, University of Jyväskylä, Department of mathematics, 1991.

[14] J. Sokolowski and J. P. Zolesio. Introduction to shape optimization: shape sensitivity analysis. volume 16 of *Springer Series in Computational Math.*, Springer Berlin, 1991.

[15] J. Sprekels and F. Tröltzsch. On an optimal shape design problem connected with the heating of elastic bodies. To appear.

[16] T. Yao and Choi K. K. 3–d shape optimal design and automatic finite element regridding. *Int. J. Numer. Methods Eng.*, 28 No. 2:369–384, 1989.