Computer Assisted Mechanics and Engineering Sciences, 6: 337–343, 1999.

Advanced solving techniques in optimization of machine components[1]

Gundolf Haase and Ewald H. Lindner
Institute for Analysis and Computational Mathematics
Johannes Kepler University of Linz, Linz, Austria

(Received March 8, 1999)

We consider the optimal design of a machine frame under several stress constraints. The included shape optimization is based on a Quasi-Newton Method and requires the solving of the plain stress state equations in a complex domain for each evaluation of the objective therein. The complexity and robustness of the optimization depends strongly on the solver for the pde. Therefore, solving the direct problem requires an iterative and adaptive multilevel solver which detects automatically the regions of interest in the changed geometry. Although we started with a perfected type frame we achieved another 10 % reduction in mass.

1. Introduction

Some 20 kilometers away from Linz the industrial company ENGEL Maschinenbau GesmbH in the field of mechanical engineering is situated. One of their fields in constructing, fabricating and selling industrial machine components is in injection moulding machines. The classical construction is based on bars, but ENGEL Maschinenbau holds a patent for the barless construction. The mass of the frame of a mid range injection moulding machine amounts to 4 tons, hence limits on transportation facilities hamper marketing of these machine components, i.e., increase costs for transportation.

The following statements are correct for construction in lines of products, but they became more and more important for construction in single units also:

- An industrial machine component, i.e. in our application the frame of an injection moulding machine, has to meet certain requirements that are fixed in a contract between the enterprise and the purchaser.
- Due to lack of time the mechanical engineer who designs the industrial machine component has to stop his designing process after two or three drafts and take the best draft obtained so far.
- There is no more time left for more drafts that would either meet the requirements to a larger extent or be constructed more cheaply.

Becoming aware of these facts leads ultimately to the use of mathematics, resp. optimization, in industry, and, as a first step, to

2. The modeling of the design of the frame of an injection moulding machine

The frame of an injection moulding machine is briefly described by its 2D-cut given in Fig. 1. Typical dimensions are:

[1]This work has been partially granted by the *Forschungsförderung der Allgemeinen Sparkasse Oberösterreich.*

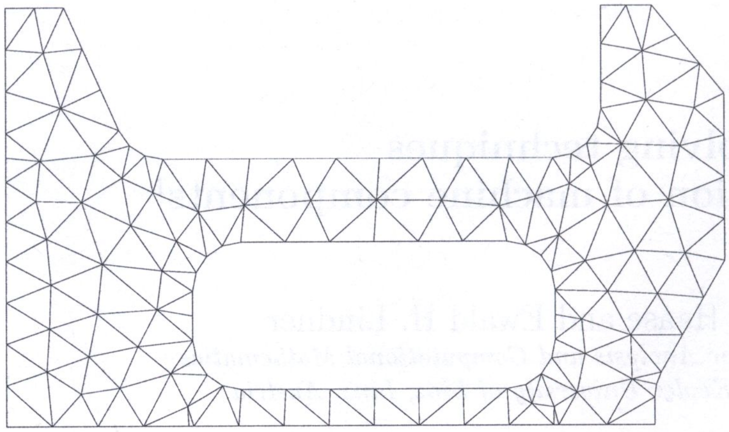

Fig. 1. Initial grid of the original shape (204 elements)

- thickness of one plate: 180 mm
- mass of one plate: 3.8 tons
- clumping force (line force): 300 tons $\approx$ 2943 kN$\widehat{=}$ 16 N/mm^2
- length / height: 2.8 m / 1.7 m
- 2 supporting points (areas).

Our first goal is to minimize the mass of the frame of the injection moulding machine subject to certain requirements (not all of them are constraints in the classical nomenclature of optimization), namely

- maximal von Mises stress: $\sigma_{\max}$
- maximal tensile stress: $\tau_{\max}$
- shrinking angle of clumping unit (vertical edges on top, called *wings*): $\alpha_{\max}$
- handling of the machine, feeding mechanism
- easy manufacturing,

where the boundary of the cross section is described by parts of arcs and straight lines. Hence, we use 2D coordinates of corner points and their radii of curvature as design variables, i.e., $(x_i, y_i, r_i) =: v_{D,i}$.

$$-\mu \Delta \mathbf{u} - (\lambda + \mu) \nabla \operatorname{div} \mathbf{u} = b \qquad \text{in } \Omega\,, \qquad (1)$$

$$+ B.C.$$

$$\lambda = \frac{E\nu}{(1+\nu)(1-\nu)}\,, \qquad \mu = \frac{E}{2(1+\nu)}\,,$$

with

λ, μ	Lamé's elasticity constants
E	modulus of elasticity
ν	Poisson's ratio [2,7].

This pde governs the following shape optimization problem:

Minimize

$$\mathcal{F}(\mathbf{u}(v_D), v_D)$$

with respect to

$$v_D := (v_{D,1}, \dots, v_{D,N})$$

taking into account the constraints

$$\begin{aligned} \sigma(x) &\le \sigma_{\max} \quad \forall x \in \Omega(v_D) \\ \alpha(\text{wings}) &\le \alpha_{\max} \\ \tau(x) &\le \tau_{\max} \end{aligned}$$

admissible geometry $\Omega(v_D) \iff$ box constraints on v_D

Here, $\mathbf{u}$ is the (discrete) solution of the linear elasticity equations.

Most requirements are not really constraints, e.g., a larger angle will be acceptable if the mass decreases significantly. Hence, all requirements but the geometrical constraints on the design variables themselves are put into the objective

$$\begin{aligned} \mathcal{F}(v_D, \mathbf{u}(v_D), \sigma, \alpha) = w_m \cdot \text{mass}(v_D)/\text{m}_0 + w_\sigma \cdot \sum \{\max[\sigma(x(v_i))/\sigma_{\max} - 1,\, 0]\}^2 \\ + w_{\alpha,m} \cdot [\max(\alpha(\text{wings})/\alpha_{\max} - 1,\, 0)]^2 \\ + w_{\alpha,s} \cdot [\max(\alpha(\text{wings})/\alpha_{\text{soft}} - 1,\, 0)]^2 \end{aligned} \tag{2}$$

- The term $\{\max[\sigma(x(v_i))/\sigma_{\max} - 1,\, 0]\}^2$ fulfills the requirement $\mathcal{F} \in C^1$.
- $\max(\alpha(\text{wings})/\alpha_{\text{soft}} - 1,\, 0)$ denotes the quality of the shrinking angle with respect to a lower bound. Values of α in the interval $[\alpha_{\text{soft}}, \alpha_{\max}]$ are accepted but increase the functional $\mathcal{F}$.
- w_m , $w_{\alpha,m}$, $w_{\alpha,s}$, w_σ are the weights of importance of a criteria for the engineer.
- Geometrical constraints are handled by an active index set strategy. A regular geometry is guaranteed by the choice of box constraints and by additional regularity tests.

3. Numerical treatment

3.1. Advanced solver for the direct problem

Replacing the displacements $\mathbf{u}$ by the approximate FEM-solution $\mathbf{u_h}$ [3,7] the solving of Eq. (1) reduces to the solving of the huge linear system of equations

$$K \cdot \underline{\mathbf{u}} = \underline{b}\ . \tag{3}$$

Due to complexity and robustness reasons in the solution process of (3) we need

- iterative Multigrid/Multilevel solvers with adaptive grid refinement, which
- find automatically (also unexpected) areas of interest due to the
- accuracy of derived values (σ) at near-by singularities.

These requirements are met by software developed at our institute in the past years, coded in C++, see NETGEN [5], FEPP [6]. The control of the adaptive FEM-grid refinement on each level k is done by an a posteriori error estimator for the error of the solution $\mathbf{u_h}^k$, i.e., $\tilde{z}_h^k \approx z_h^k := |\mathbf{u_h}^k - \mathbf{u}_\mathbf{h}^*|$ where $\mathbf{u_h}^*$ denotes the f.e. interpolation of the exact solution $\underline{\mathbf{u}}$ of system (3).

The numerical examples for the solution of the direct problem, i.e., solving the plain stress state equations, reveal that uniform refinement of the grid immediately leads to huge requirements on storage space and consequently to enormous CPU-consumption.

If using an iterative solver for the direct problem, then we have to pose the question how many adaptive levels are necessary?

In the optimization code (see below) we use numerical differentiation, hence an accuracy of at least 1% in functional $\mathcal{F}$ is desirable. This implies that

$$||\sigma_{\max}^{(k)} - \sigma_{\max}^*|| < 0.01 \cdot \sigma_{\max}^* \ ,$$

(similar for $\alpha^{(k)}$, α^*) where the exact values α^*, $\sigma_{\max}^*$ are unknown. An error estimator working on these criteria is not implemented so far. So we use an heuristic criteria in combination with a conventional error estimator, i.e.

Stop the adaptive solver, if

$$\begin{aligned} &||\tilde{z}_h^k|| && < 0.01 \cdot ||\tilde{z}_h^0|| && \text{and} \\ &||\sigma_{\max}^{(k)} - \sigma_{\max}^{(k-1)}|| && < 0.01 \cdot \sigma_{\max}^{(k)} && \text{and} \\ &||\sigma_{\max}^{(k-1)} - \sigma_{\max}^{(k-2)}|| && < 0.01 \cdot \sigma_{\max}^{(k-1)} \ . \end{aligned} \tag{4}$$

Here, we omit the condition on α because the accuracy of the angle is not critical in the iteration.

If the supporting points/edges are modeled as Dirichlet B.C. with no displacement in that region ($\mathbf{u} = (0,0)^T$), then we achieve high stresses in these areas. This is due to the fact that zero Dirichlet B.C. do not reflect the flexibility of the supporting points, i.e., the frame is not strictly fixed there. To avoid solving a contact problem we handle these Dirichlet B.C. as Robin B.C. with a function as f.e.-penalty parameter. Assuming x_s and x_e as starting and end point of the supporting edge the boundary conditions are

$$0 = \underbrace{10^4 \cdot (x - x_s)(x - x_e)}_{\text{f.e.-penalty}(x)} \cdot (\mathbf{u}(x) - 0) \ ,$$

which ensures support in the center of the edge and increasing flexibility away from it.

3.2. Shape optimization

The variables are arranged in groups of discrete and continuous variables. Mathematically, our optimization problem is of the form

minimize

$$f(x_I, x_C)$$

with respect to

$$x = (x_I, x_C)$$

subject to

$$\begin{aligned} &\underline{x_I} \le x_I \le \overline{x_I} \\ &\underline{x_C} \le x_C \le \overline{x_C} \\ &x_{I,k} \in \mathbb{N} && \text{for } 1 \le k \le D_I \\ &x_{I,k} = \underline{x_{I,k}} + m s_k && \text{for some } m \in \mathbb{N}_0 \ , \ \text{for } 1 \le k \le D_I \\ &x_{C,k} \in \mathbb{R} && \text{for } 1 \le k \le D_C \, . \end{aligned}$$

s_k is a given integer step size. D_I and D_C , denotes the dimension of the vector x_I and x_C , respectively. At first we generate a list of all continuous box constrained optimization problems by totally enumerating all discrete feasible points (in our case there is for the moment only one discrete variable, the number of wholes in the frame). In the second part of the code all continuous problems are treated as follows.

As there are no equality constraints and as the gradients of the inequality constraints are equal to either e_k or $-e_k$, where e_k denotes the k^{th} unit vector, the regularity conditions and hereby the Kuhn-Tucker-Theorem hold. All continuous variables are scaled, i.e.

$$\tilde{x}_{C,k} := x_{C,k} \bigg/ \frac{|\underline{x_{C,k}}| + |\overline{x_{C,k}}|}{2}$$

For sake of simplicity we will always use the notation x_C instead of $\tilde{x}_C$. Starting from a feasible point for the continuous problem, which is always known – e.g. the central point of the feasible region that is formed by the box constraints – we use a Quasi-Newton method (see e.g. [1, pp.138]) which takes advantage of the simple structure of the constraints during generating and solving the corresponding quadratic optimization problems by the active index set strategy. For updating the Quasi-Newton matrix we use a modified BFGS formula, following Powell [4]. The optimal solution of one of the generated continuous problems is denoted by $x^*_C(x_I)$.

After solving all generated continuous problems, the solution of the mixed integer problem is easily found by

minimize

$$f(x_I, x^*_C(x_I))$$

with respect to x_I , subject to

$$\underline{x_I} \le x_I \le \overline{x_I}$$

$$x_{I,k} \in \mathbb{N} \qquad \text{for } 1 \le k \le D_I$$

$$x_{I,k} = \underline{x_{I,k}} + m s_k \qquad \text{for some } m \in \mathbb{N}_0\,, \quad \text{for } 1 \le k \le D_I$$

i.e., just by comparing values of the objective already calculated.

4. Application

During testing the above code we realized that there are more requirements to be met by the design of a frame for an injection moulding machine which were not stated in the beginning. Now this requirement on the tensile stress has already been included in the model. During the *automatic* optimization process we achieved temporary frame geometries resulting in very high stresses in single points or areas which would lead to a damage of the machine. To catch such regions without a very fine grid in the whole domain an adaptive solver is strictly recommended – otherwise we may run into non-admissible solutions.

Note, that part of our objective looks like applying a penalty method. In our model these weight factors are used and interpreted as factors of importance, the "penalty parameters" are about of the same magnitude as the coefficients of the elements in the objective. As our penalty parameters do not tend to infinity, as in the penalty method for constrained optimization, but are fixed given weights, we do not have to cope with the problem of ill-conditioning for larger and larger penalty parameters.

5. Numerical Examples

We used a SUN-ULTRA 170MHz (9 SPEC fp95) for optimizing the initial geometry from Fig. 1. The mesh generation and solving of one direct problem took us 40–60 sec.

The shape of the frame was optimized with respect to the hole in the center, i.e., we had 12 design variables. Our automatic optimization code stopped after 9 hours with the geometry presented in Fig. 2, here the adaptive code needed 7 levels to fulfill stopping criteria (4) in the direct problem.

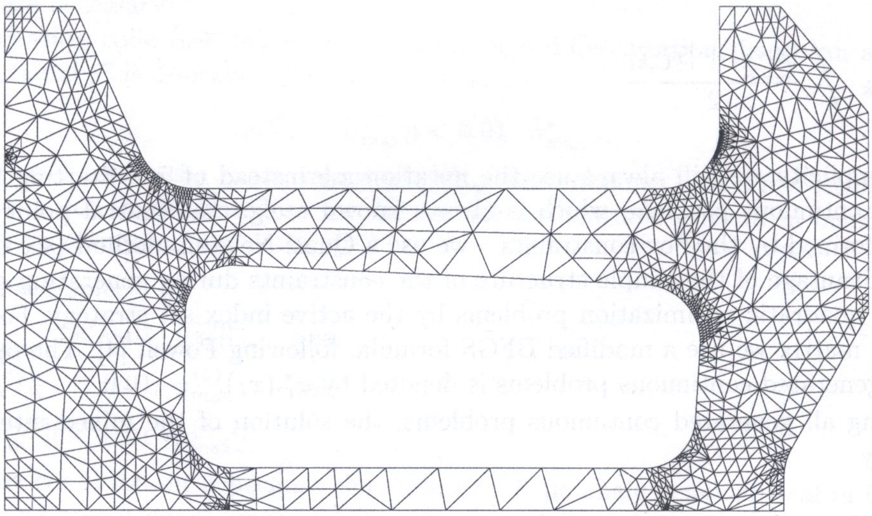

Fig. 2. 7th adaptive grid of the optimized shape (2257 linear elements)

The weighting factors in (2) have been chosen as $w_m = w_\sigma = w_{\alpha,m} = 1$ and $w_{\alpha,s} = 0.5$. Because of the linear finite elements, the edge based residual error estimator used in the code results in a quite dense refinement in the areas of interest. The usage of quadratic test f.e. functions results in a sparser grid but comparable solution time. Taking into account that the original frame has already been produced for several years a mass reduction of 10 % is not bad. Additionally, a smaller weight $w_{\alpha,s}$ in the soft limit for the angle would result in a bigger decrease of mass. The value of that weight is still in discussion. Actually, we choose the soft limit α_{soft} for the angle by 95 % of the strict limit $\alpha_{\max}$.

Table 1. Mass reduction and fulfilling of constraints in per cent

	orig. geom.	opt. geom.
mass	100 %	90 %
$\max \sigma/\sigma_{\max}$	75 %	71 %
$\max \tau/\tau_{\max}$	81 %	100 %
$\alpha/\alpha_{\text{soft}}$	88 %	96 %
$\alpha/\alpha_{\max}$	92 %	102 %

6. Conclusions

Using advanced mathematical techniques in the solving of the direct problem opens the gate to fast 2D optimization. Due to the fact that the whole optimization code has to run automatically, one has to guarantee that the solver for the direct problem will detect all critical stresses, displacements, etc. without any interaction by the user. This can only be achieved by adaptive codes – maybe by implementing problem specific error estimators. Otherwise, the optimization process will return some geometry leading to serious damages of the machine, i.e., hours or days of computational work become worthless.

The handling of 3D optimization problems will be only feasible in future if these advanced techniques, i.e., multilevel and adaptivity, are transfered to the optimization algorithms.

Acknowledgment

We would like to thank our colleague Joachim Schöberl for the assistance in adapting the code FEPP to our needs and Michael Bauer from ENGEL Maschinenbau for stimulating discussions.

References

[1] R. Fletcher. *Practical Methods for Optimization*, Vol. 2: *Constrained Optimization*. John Wiley & Sons, Chichester, 1981.

[2] V.G. Korneev, U. Langer. *Approximate Solution of Plastic Flow Theory Problems*. Vol. 69 of Teubner-Texte zur Mathematik, Teubner, Leipzig, 1984.

[3] J.T. Oden E.B. Becker, G.F. Carey. *Finite Elements, The Finite Element Series I–VI.* Prentice Hall, Englewood Cliffs, N.J., 1982.

[4] M.J.D. Powell. A fast algorithm for nonlinear constrained optimization calculations. In: G.A. Watson (ed.), *Numerical Analysis*, Dundee 1977. *Lecture Notes in Mathematics* 630, Springer-Verlag, Berlin, 1978.

[5] J. Schöberl. *NETGEN*. Technical Report No 95-3, Institut für Mathematik, Johannes Kepler Universität Linz, 1995.

[6] J. Schöberl. *FEPP*: `www.numa.uni-linz.ac.at/projects/doc/`

[7] O.C. Zienkiewicz, R.L. Taylor. *The Finite Element Method*, Volume 1/2. McGraw-Hill Book Company, New York, 1991