TOPOLOGY OPTIMIZATION OF SHEETS IN CONTACT BY A SUBGRADIENT METHOD *

JOAKIM PETERSSON
Department of Mechanical Engineering, Division of Mechanics, Linköping Institute of Technology, S-581 83 Linköping, Sweden

and

MICHAEL PATRIKSSON
Department of Mathematics, Box 354350, University of Washington, Seattle, WA 98195-4350, U.S.A.

Abstract. We consider the solution of finite element discretized optimum sheet problems by an iterative algorithm. The problem is that of maximizing the stiffness of a sheet subject to constraints on the admissible designs and unilateral contact conditions on the displacements. The model allows for zero design volumes, and thus constitutes a true topology optimization problem.

We propose and evaluate a subgradient optimization algorithm for a reformulation into a nondifferentiable, convex minimization problem in the displacement variables. The convergence of this method is proven, and its low computational complexity is established. An optimal design is derived through a simple averaging scheme which combines the solutions to the linear design problems solved within the subgradient method.

To illustrate the efficiency of the algorithm and investigate the properties of the optimal designs, the algorithm is numerically tested on some medium and large scale problems.

Key words: optimum sheet; unilateral contact; nondifferentiable convex programming; subgradient optimization algorithm

1 INTRODUCTION

In this paper we propose a subgradient optimization algorithm for the solution of finite element discretized stiffest sheet problems, including unilateral contact. The method applies to any structure with a linear energy-design dependence, but the interpretation of the abstract formulation as a variable thickness sheet problem is the one preferred. There is an abundance of works in adjoining research fields, such as topology structural design and mathematical programming, whence we next provide a background to the problem under consideration and the context of our work.

1.1 Models of stiffness maximization and topology design

We start by giving a short background to the common choice of cost functional in topology optimization.

Prager and Taylor[1] considered in 1968 the problem of maximizing the stiffness of distributed parameter elastic systems, and derived the sufficient optimal-

* This work was supported by the Swedish Research Council for Engineering Sciences (TFR), grants 92-426 and 282-93-1195.

ity criterion of constant specific strain energy per unit specific stiffness. In 1973 Rossow and Taylor[2] used this optimality criterion and a modified Newton–Raphson method to obtain numerical solutions to a maximum stiffness problem for the special case of variable thickness sheets. Benedict[3] considered in 1982 the maximum stiffness problem in case of unilateral contact, and proposed a nested optimality criteria approach to solve an example with a beam on Winkler foundation. He also concluded that, in the presence of contact, the maximization of equilibrium potential energy (stiffness) is not equivalent to minimizing the work of the external loads (compliance), which is otherwise the case.

Among the different types of structural optimization problems, *topology optimization* seems to be the one which yields the greatest improvements in structural performance. The problem formulation most frequently encountered is that of *compliance minimization*, in which the design variables are allowed to take (close to) zero values. The design variables enter the bilinear form of the total potential energy functional. If microstructures are included, then this dependence is usually nonlinear and determined by homogenization formulas. The potential energy-design relation is, however, linear both for variable thickness sheets and for trace constrained locally optimized material, as well as for sandwich plates and truss structures. Céa and Malanowski[4] established in 1970 the existence of solutions to problems with linear energy-design relations. They did, however, consider neither zero lower design bounds nor contact conditions; in order to cover topology optimization of contacting structures, existence and uniqueness results were established in Reference 5 using saddle point theory. This study was followed by that in Reference 6, in which the convergence of finite element solutions was studied. The present work, in turn, can be seen as a continuation of Reference 6.

The continuous sheet problem is discretized by finite elements such that displacements are approximated as elementwise linear functions and designs as elementwise constant. The discretized topology optimization problem is a convex-concave saddle point problem over the product of the sets defined by the unilateral contact and design constraints. The saddle function is convex quadratic in the displacement variables and linear in the design variables. In true topology optimization, zero volumes are admissible; in these cases, the saddle function is not strictly convex in the displacement variables for some feasible designs.

1.2 Computational procedures

Probably the most common solution strategy for general structural optimization is the *nested approach*, in which one alternates between the solution of an *equilibrium* (or, *analysis*) problem given a fixed design, and the solution of a problem in the design variables (the *design* problem) given fixed state (displacement) variables. In topology optimization an optimality criteria method is frequently used[7]. In such a method the design problem includes a design update which utilizes the difference in strain energy density in different positions of the structure, and yields more material at positions with high energy and less material where the energy is low.

The value of the Lagrange multiplier for the volume constraint determines the size of the update[8].

We next describe some of the computational difficulties that are associated with allowing zero design volumes and with the presence of displacement constraints.

The non-strict convex-concavity of the saddle function constitutes a computational challenge: first, because the analysis problem may fail to have a solution or have multiple solutions for certain designs; second, because it implies that an optimal design cannot be generated through the solution of the linear design problem given an optimal state vector. (The latter remains true even if the lower bounds on the volumes were positive.) An immediate consequence is that a nested approach cannot be shown to produce an optimal solution.

In the absence of contact conditions, the analysis problem is equivalent to a system of linear equations which, under the additional assumption that the lower bound on the design volume is positive, is guaranteed to have a (unique) solution. The inequality contact constraints give rise to complementarity conditions, which make the analysis problem significantly more difficult to solve, and also demands for quite different solution techniques than the analysis problem in the non-contact case.

We next describe the three (non-contact topology optimization) algorithms most relevant to the present work. They are the nested approach of Céa and Malanowski[4], the ε-steepest descent approach suggested by Ben-Tal and Bendsøe[9], and the barrier approach of Ringertz[10]. In the nested approach of Céa and Malanowski the design update consists of a relaxation step in the design variable space. The direction of the step is determined as the difference between the solution of the linear design problem (the maximization of potential energy keeping the displacement variables fixed) and the current design iterate. This algorithm is proven convergent for strictly positive lower bounds, but it is not applicable to true topology problems.

The ε-steepest descent approach of Ben-Tal and Bendsøe utilizes an unconstrained reformulation of the saddle point problem in terms of the displacement variables. The sparsity of truss structures—the number of displacement variables is much lower than the number of truss bars—implies that the quadratic programs iteratively solved to obtain a descent direction are relatively small; an optimal design, which is allowed to include zero volumes, is also generated simultaneously with the optimal displacement vector through the solution of these quadratic programs. This relation between the number of elements and displacement variables present in truss topologies does however not hold for variable thickness sheet topologies, and the presence of the displacement constraints further significantly complicates the generation of a descent direction. If one chooses to determine the descent direction using a dual reformulation, then the problem size is determined by the number of (almost) active elements; this number is expected to be large compared to the ones appearing for trusses in Reference 9.

Ringertz transforms the original problem by introducing barrier functions for the volume bound constraints; the resulting problem is, for a given penalty parameter, solved using a Newton algorithm. An improper choice of penalty parameters can make it laborious to identify the vanishing designs, and, in addition, the presence of displacement constraints would probably make also this approach impractical due to the introduction of complementarity conditions.

In addition to these algorithms, classical saddle point algorithms may also be considered. They fall into two categories: in the first, one alternates between finding a solution to one of two inequalities defining the saddle point problem followed by an update. (The two inequalities correspond to the linear design problem and the equilibrium problem, respectively.) This strategy is the basis for the nested approaches described above, of which the algorithm of Céa and Malanowski is a special case. (In the absence of strict convex-concavity, the saddle function may be iteratively regularized, i.e., augmented by a strictly convex-concave function, to obtain this property; one example is the proximal point algorithm[11].) The optimality criteria method is similar to the algorithm in this category, since the updating formula can be derived from the optimality conditions for the first inequality of the saddle point problem. Since for certain designs the equilibrium problem may fail to have a solution, for our purposes these approaches are invalid in their original statements.

The second category recognizes the strong relationship between saddle point problems and *minimax* problems[12,13]. A classical result (see, e.g., Reference 14, Theorem VII.4.2.5) states that the set of saddle points, when nonempty, coincides with the product of the solutions to its associated minimax and maximin problems. (In this context, the minimax problem, referred to as the *primal* problem, is the equilibrium problem of a fictitious elastic material with a certain convex strain energy potential, and the maximin formulation, or, the *dual* problem, is the problem of finding a design with the maximal equilibrium potential energy.) In general, both these problems must be solved in order to yield a saddle point. (In our context, for example, it is not sufficient to solve the primal problem, and use the corresponding optimal displacement vector to generate an optimal design through the second inequality of the saddle point problem.) In the strictly convex-concave case, however, the inner problem of one of these reformulations provides the solution to the other one (see Reference 14, Theorem VII.4.3.1); this property of the saddle function also validates effective step length rules based on the improvement of merit functions. Ben-Tal and Bendsøe[9] introduce an alternative approach, in which an optimizing sequence of dual (design) solutions is generated in the process of solving the primal problem. This is however achieved at the cost of solving a quadratic programming problem in every iteration, which, in our context, is too expensive computationally.

We end this subsection with an outline of the new method.

The approach we take is reminiscent to that of Ben-Tal and Bendsøe in the sense that also we solve the saddle point problem through the solution of the primal problem. The algorithm has its foundations in subgradient optimization techniques in nondifferentiable convex programming. Such techniques have been frequently and successfully applied in some practical contexts, especially in the context of combinatorial optimization and Lagrangean relaxation. The properties of the saddle problem can be used to establish the convergence to an optimal displacement vector using this approach. From existing convergence results one can however not infer the existence of a convergent sequence of design solutions. (Essentially, this is due to the non-strict concavity of the saddle function with respect to the design variables, and is in the context of Lagrangean relaxation known as the *non-coordinability phenomenon.*)

Our main technical contribution is the result that, under certain conditions on the step length rule used, the simple average of the designs, generated in the subgradient optimization scheme as part of the construction of subgradients, converge to an optimal design; the design and displacement solutions are hence automatically and simultaneously obtained through the use of the subgradient optimization scheme. A second, minor, contribution is a finite termination criterion for the algorithm, based on the averaged design solution.

A main improvement over previous algorithms (in addition to the fact that the proposed algorithm is applicable to the general problem with possibly zero design volumes and with unilateral contact conditions) is that the optimal design is obtained without having to solve a quadratic program (especially, we never solve the analysis problem during the algorithmic process); as a result of this fact—in combination with the mild memory requirement of the algorithm—the proposed algorithm can be utilized to solve very large-scale problems. In this context, we remark that in previous engineering works on topology optimization, it is common to solve problems with 2000–6000 design variables (at least if trusses are excluded), while, in the present article, numerical results are presented where the number of design variables, as well as the number of state variables, exceeds 10,000.

1.3 Preview

The rest of the paper is organized as follows. Section 2 summarizes the formulation of the optimum sheet problem and its corresponding finite-dimensional saddle problem, together with their most important properties. Section 3 outlines and establishes the convergence of a subgradient optimization algorithm for the generation of an optimal displacement vector. In Section 4 we describe an augmentation of this algorithm, which constructs the simple average of all tentative designs generated; the convergence of this sequence to an optimal design is established, and we describe and validate a simple criterion for the finite termination of the overall algorithm. Finally, in Section 5 we present numerical results. Practical considerations and numerical experience concerning the algorithm implementation are given, we discuss the possible instabilities in the design sequences and their mitigation, and provide computed solutions with mechanical interpretations.

2 FORMULATION OF THE OPTIMUM SHEET PROBLEM

2.1 The Original Problem in Infinite Dimension

The optimum sheet problem with unilateral contact was formulated and mathematically studied by Petersson[5], whose presentation is summarized below.

Consider a sheet with a mid-surface occupying a two-dimensional domain Ω, and with a thickness $h(x)$ at the point x in Ω. The boundary is partitioned into the parts Γ_d, Γ_t and Γ_c, the first two being the parts where zero displacements and tractions are prescribed, respectively, and the last being the candidate contact boundary.

Each point x in Ω has an in-plane displacement $u(x)$. The corresponding field belongs to

$$W = \left\{ u = (u_1, u_2) \in (H^1(\Omega))^2 \mid u_i = 0 \text{ on } \Gamma_d, \quad i = 1, 2 \right\},$$

and, since the considered contact is with a rigid frictionless obstacle, to the convex set

$$\mathcal{U} = \{ u \in W \mid u_i n_i - g \leq 0 \text{ on } \Gamma_c \}.$$

Here, n_i are the components of a unit length vector indicating contact directions, the summation convention in which the indices take the values 1 and 2 is used, and g is the initial normal distance in the potential contact region.

The total potential energy functional is

$$L^\infty(\Omega) \times (H^1(\Omega))^2 \ni (h, u) \longmapsto J_h(u) \in \Re,$$

where

$$J_h(u) = \frac{1}{2} a_h(u, u) - L(u),$$

and where the linear and bilinear forms are

$$L(u) = \int_\Omega f \cdot u \, d\Omega \; + \int_{\Gamma_t} t \cdot u \, d\Gamma \quad \text{and} \quad a_h(u, v) = \int_\Omega h u_{i,j} E_{ijkl} v_{k,l} d\Omega.$$

Here, E_{ijkl} are the elasticity constants satisfying the usual ellipticity, symmetry and boundedness conditions and $u_{i,j}$ means $\frac{\partial u_i}{\partial x_j}$. The functional $L(\cdot)$ depends on the body forces $f \in (L^2(\Omega))^2$ and the tractions $t \in (L^2(\Gamma_t))^2$.

The set of permissible designs can be described as

$$\mathcal{H} = \left\{ h \in L^\infty(\Omega) \middle| \, h_{min} \leq h \leq h_{max} \quad \text{a. e. in } \Omega; \quad \int_\Omega h \, d\Omega = V \right\},$$

where h_{min} and h_{max} are bound functions whose properties are such that $\mathcal{H}$ is nonempty, convex and weakly* compact, and V is the available volume of material to be distributed in the domain Ω.

The equilibrium problem can be formulated as the variational inequality

$$(\mathcal{V}) \quad u \in \mathcal{U}: \; a_h(u, \varphi - u) \geq L(\varphi - u), \qquad \forall \varphi \in \mathcal{U},$$

or, equivalently, as a minimum of total potential energy:

$$(\mathcal{M}) \quad u \in \mathcal{U}: \; J_h(u) \leq J_h(\varphi), \qquad \forall \varphi \in \mathcal{U}.$$

If a solution, u, to the variational inequality $(\mathcal{M})$ is sufficiently regular, and if $h_{min} > 0$, then

$$\begin{array}{l} \{h\sigma_{ij}(u)\}_{,j} + f_i = 0 \quad \text{in } \Omega, \\ \sigma_{ij}(u) \equiv E_{ijkl} u_{k,l}, \\ u_i = 0 \quad \text{on } \Gamma_d, \\ \sigma_{ij}(u) n_j = \frac{1}{h} t_i \quad \text{on } \Gamma_t, \\ \left.\begin{array}{l} \sigma_{Ti} = 0, \quad \sigma_N(u_N - g) = 0 \\ \sigma_N \leq 0, \quad u_N - g \leq 0 \end{array}\right\} \text{ on } \Gamma_c, \\ \sigma_N \equiv \sigma_{ij}(u) n_i n_j, \quad u_N = u_i n_i, \\ \sigma_{Ti} \equiv \sigma_{ij} n_j - \sigma_N n_i, \end{array}$$

among which the last four rows describe the classical Signorini contact conditions.

The structural optimization problem to be solved can be formulated as

$$(SJ) \quad \left\{ \begin{array}{l} \text{Find } (\tilde{u}, \tilde{h}) \in \mathcal{U} \times \mathcal{H} \; : \\ J_h(\tilde{u}) \leq J_{\tilde{h}}(\tilde{u}) \leq J_{\tilde{h}}(u), \qquad \forall (u, h) \in \mathcal{U} \times \mathcal{H}. \end{array} \right.$$

A solution $(\tilde{u}, \tilde{h})$ maximizes the equilibrium potential energy, and solves

$$\begin{array}{ll} \min\limits_{u,\, h} & L(u) + \int_{\Gamma_c} h |\sigma_N(u)| g \, d\Gamma, \\ \text{subject to:} & \text{equilibrium } (\mathcal{V}), \\ & \text{design constraints } h \in \mathcal{H}, \end{array}$$

which is a more traditional form than (SJ). The appearance of $\sigma_N(u) \equiv E_{ijkl} u_{k,l} n_i n_j$ (the contact stresses) is according to the extended Clapeyron's theorem for unilateral contact. The objective functional to be minimized is a sum of two terms; one is a weighted measure of displacements and the other of contact stress magnitudes. For further discussions on (SJ), we refer to Reference 15.

According to results in Reference 5, there exist solutions to (SJ), and the solution set is a closed, bounded and convex subset of $\mathcal{U} \times \mathcal{H}$; a solution is not necessarily unique unless the lower bound h_{min} is strictly positive. (The more traditional formulation of (SJ) has the drawback that the optimal displacements in regions of zero thickness are unbounded.)

2.2 The Finite Element Discretized Problem

When introducing the finite element discretization, the notation u, h, g, $\mathcal{U}$ and $\mathcal{H}$ will be kept; from now on, it is understood to be associated with the discretized formulation.

As finite elements we choose linear triangles, and hence approximate the displacement components as elementwise linear functions; the approximate thickness distribution is elementwise constant. The nodal values of the approximate displacement field are collected in a vector u of dimension n, and t is a vector containing m thickness values, m being the number of triangles in the mesh. After discretization, the saddle function J is replaced by the saddle function $\Pi : \Re^n \times \Re^m \to \Re$ defined by

$$\Pi(u,h) = \frac{1}{2}u^T K(h)u - f^T u,$$

where

$$K(h) = \sum_{i=1}^{m} h_i K_i \tag{1}$$

is the structural stiffness matrix and f the structural load vector. In (1), $h_i = t_i S_i$, where t_i and S_i denote the thickness and area, respectively, of the ith element, and K_i is the global element stiffness matrix per unit volume. The matrices K_i, $i = 1, 2, \ldots, m$, are symmetric and positive semidefinite in $\Re^{n\times n}$.

The discretized version of the saddle problem (SJ) is

$$(S\Pi) \quad \begin{cases} \text{Find } (u^*, h^*) \in \mathcal{U} \times \mathcal{H} \ : \\ \Pi(u^*, h) \le \Pi(u^*, h^*) \le \Pi(u, h^*), \qquad \forall (u,h) \in \mathcal{U} \times \mathcal{H}, \end{cases}$$

where the admissible set of the variables u is the nonempty polyhedral set

$$\mathcal{U} = \{\, u \in \Re^n \mid Cu \le g \,\},$$

and the admissible set of the variables h is the bounded polyhedral set

$$\mathcal{H} = \left\{ h \in \Re^m \;\middle|\; \sum_{i=1}^{m} h_i \equiv 1^T h = V, \quad \underline{h} \le h \le \overline{h} \right\}.$$

The vector g of dimension r contains initial contact distances from nodes in Γ_c to the boundary of the rigid obstacle. The ith row of the matrix $C \in \Re^{r\times n}$ is the transpose of a unit length vector ν_i which indicates the direction from the ith contact node to the corresponding obstacle point. If no more than one contact condition is imposed for each node of the structure, then the vectors ν_i will not have any nonzero components at the same positions; hence, $\nu_i^T \nu_j = \delta_{ij}$ holds, where δ_{ij} is the Kronecker delta. This implies that CC^T is the unit matrix, and leads us to the following

Definition 2.1 (Quasi-orthogonality) *An $r \times n$ matrix C is said to be* quasi-orthogonal *if $CC^T = I$, where I is the $r \times r$ unit matrix.*

The available material volume is V and $\underline{h}$ and $\overline{h}$ contain the elementwise lower and upper design bounds, respectively. We assume that $0 \leq \underline{h}_i \leq \overline{h}_i \leq V$ for all i, and that $1^T\underline{h} < V \leq 1^T\overline{h}$; thus, $\mathcal{H}$ is nonempty. It should be noted that the matrix $K(h)$ is positive definite for each strictly positive vector h. These properties of $\mathcal{U}$, $\mathcal{H}$ and Π are assumed to hold throughout the paper.

For completeness, we state the necessary and sufficient conditions for (u^*, h^*) to solve $(S\Pi)$, cf. References 16,9:

$$(j) \qquad K(h^*)u^* = f - C^T\lambda^* \tag{2}$$

$$\lambda^* \geq 0 \tag{3}$$

$$\lambda^{*T}(Cu^* - g) = 0 \tag{4}$$

$$Cu^* \leq g \tag{5}$$

$$(jj) \qquad h^* \in \mathcal{H} \quad \text{and} \quad \exists\, \kappa \in \Re :$$

$$u^{*T}K_iu^* < \kappa \implies h_i^* = \underline{h}_i$$

$$u^{*T}K_iu^* > \kappa \implies h_i^* = \overline{h}_i$$

The first part (j) stems from the second inequality of $(S\Pi)$ and constitutes the conditions for equilibrium. Here, (2) describes the force equilibrium conditions and λ_i^* is the *contact force magnitude* at contact node i. The inequality (3) represents non-adhesion, (5) non-penetration, and (4) states that a contact force is developed only if contact occurs. In (jj), which are the KKT-conditions for the linear program defined by the first inequality of $(S\Pi)$, we recognize the classical optimality condition of uniform strain energy density.

Petersson and Haslinger[6] study the convergence of the fields $\tilde{u}_m$ and $\tilde{h}_m$, generated from u^* and $t^* = (\frac{h_i^*}{S_i})$, to the solutions $(\tilde{u}, \tilde{h})$ of (SJ) as $m \to \infty$. In the case where $\tilde{u}$ and $\tilde{h}$ are unique, $\tilde{u}_m$ tends weakly in W to $\tilde{u}$ and $\tilde{h}_m$ weakly* in $L^\infty(\Omega)$ to $\tilde{h}$; in the case of non-uniqueness, convergence hold in the sense of subsequences. When zero designs are not allowed, $\tilde{u}_m$ tends strongly in W to the unique solution $\tilde{u}$.

Truss structures also fit into the problem description, in which case h is a vector of bar volumes and $K_i = \frac{E}{L_i^2}\gamma_i\gamma_i^T$, where E is Young's modulus, L_i the length of the ith bar, and $\gamma_i \in \Re^n$ is a vector of direction cosines describing the orientation of the bar. As we shall see, the most time-consuming part of our algorithm has an $O(m)$ complexity, and m is often very large compared to n in truss topology optimization. Moreover, the method does not take advantage of the dyadic structure of K_i present for trusses. It may, however, still be comparable to more specialized truss topology optimization methods when multiple load cases and/or general design bounds are considered, since, in these cases, the dyadic structure is destroyed, with the consequence that the problem cannot be solved through linear programs[17].

The saddle function Π is convex-concave on the closed and convex product set $\mathcal{U} \times \mathcal{H}$; further, the function $u \mapsto \Pi(u, h)$ is weakly coercive (or, 0-coercive) for all $h > 0$, and $\mathcal{H}$ is bounded. Hence, by Theorem 4.3.1 of Reference 14 (or, Theorem 4.1 of Reference 16), there exist solutions to $(S\Pi)$, and, further, the solution set $\mathcal{U}^* \times \mathcal{H}^*$ is convex, closed and bounded. We let Π^* denote the saddle value of $(S\Pi)$, i.e., for every saddle point (u^*, h^*) of Π, $\Pi(u^*, h^*) = \Pi^*$ holds.

Associated with the saddle problem $(S\Pi)$ is the dual pair of convex programs

$$(P) \quad \begin{cases} \text{Find } u^* \in \mathcal{U} \; : \\ \psi(u^*) \leq \psi(u), \qquad \forall u \in \mathcal{U}, \end{cases}$$

where $\psi(u) \stackrel{\text{def}}{=} \sup_{h \in \mathcal{H}} \Pi(u,h)$, $u \in \mathcal{U}$, and

$$(D) \quad \begin{cases} \text{Find } h^* \in \mathcal{H} \; : \\ \varphi(h^*) \geq \varphi(h), \qquad \forall h \in \mathcal{H}, \end{cases}$$

where $\varphi(h) \stackrel{\text{def}}{=} \inf_{u \in \mathcal{U}} \Pi(u,h)$, $h \in \mathcal{H}$.

The primal problem may be interpreted as an equilibrium problem of an elastic material with nondifferentiable but convex and positive definite strain energy potential. The dual problem corresponds to a nested formulation of the design goal of maximum equilibrium potential energy.

By Theorem VII.4.2.5 in Reference 14, the solutions to (P) and (D) are $\mathcal{U}^*$ and $\mathcal{H}^*$, respectively. Hence, a solution to $(S\Pi)$ may be obtained by finding optimal solutions to the two convex programs (P) and (D).

In the next section, we formulate and establish the convergence of an iterative algorithm for the solution of $(S\Pi)$ through the solution of the primal problem (P). Reasons for choosing (P) as the basis for our algorithm were given in Section 1.2; we also remark here that the evaluation of $\psi(u)$, $u \in \mathcal{U}$, amounts to solving a very simple linear program, and that the remainder of the iterative scheme consists of even simpler matrix-vector operations. Most important is also the fact that an optimal solution h^* to the dual problem (D) may be easily obtained as a byproduct of the algorithm; the construction of this solution is presented in Section 4.

3 CONVERGENCE OF THE DISPLACEMENTS

When considering iterative algorithms for finding a saddle point of Π, we must take into account that in order to find a good approximation to the solution to (SJ), one may need to resolve the discretized problem $(S\Pi)$ with smaller and smaller element sizes; therefore, the algorithm chosen must be able to solve large-scale instances of $(S\Pi)$. This requires that the algorithm is able to utilize the structure of $(S\Pi)$, and, most importantly, that it is economical with respect to its memory requirements.

As remarked in Section 1.2, classical algorithms for the solution of saddle point problems often require that the saddle function is *strictly* convex-concave (e.g., References 13,18). In these cases, a descent direction with respect to either ψ or φ is obtained from the evaluation of the respective function. It must then be observed that Π is *not* strictly convex-concave. A descent direction (e.g., in extensions of steepest descent algorithms) may still be obtained, at the cost of solving an additional quadratic program. Such a technique has been suggested by Ben-Tal and Bendsøe[9] for the solution of an optimal truss topology design problem, which has a structure reminiscent to ours. In their application there are, however, no displacement constraints, so the primal problem (P) is unconstrained. Further, the truss topology problem is sparse in the sense that $n \ll m$, making the search-direction generating quadratic programs comparatively easy to solve iteratively. Such a scheme would not be as practical in our setting because of these differences. In another class of saddle point algorithms, strict convex-concavity is obtained through the additive augmentation by strictly convex-concave regularization functions (e.g., the proximal method of multipliers[11]); their practical use in our application is questionable, however, due to the difficulty of solving the resulting subproblems.

The algorithm chosen is a *subgradient optimization* algorithm for the solution of the program (P). This algorithm has several advantages; it is simple to implement, it utilizes the problem structure efficiently, and it has a low computational complexity of operation. The algorithm is also chosen because it is possible to augment with a very simple scheme for generating an optimal design (see Section 4).

3.1 The subgradient optimization algorithm

Definition 3.1 (Euclidean projection) *We denote by $P_{\mathcal{A}}(x)$ the Euclidean projection of $x \in \Re^n$ onto the closed and convex set $\mathcal{A} \subset \Re^n$, i.e., for any $x \in \Re^n$, $P_{\mathcal{A}}(x) \in \mathcal{A}$ is uniquely given by*

$$\|P_{\mathcal{A}}(x) - x\| \leq \|y - x\|, \qquad \forall y \in \mathcal{A}, \tag{6}$$

or, equivalently,

$$(x - P_{\mathcal{A}}(x))^T (y - P_{\mathcal{A}}(x)) \leq 0, \qquad \forall y \in \mathcal{A}. \tag{7}$$

Given a feasible starting point u^1, the subgradient optimization algorithm[19,20,21] for solving (P) generates a sequence $\{u^k\}$ of feasible solutions by the formula

$$u^{k+1} = P_{\mathcal{U}}(u^k - \alpha_k \xi_\psi(u^k)), \qquad k = 1, 2, \ldots, \tag{8}$$

where

$$\xi_\psi(u^k) \in \partial\psi(u^k) \tag{9}$$

is an element of the subdifferential (i.e., a subgradient) of ψ at u^k, and the step length α_k is chosen according to a rule which guarantees the convergence of the algorithm (8). To be able to obtain optimal designs (implicitly) through the use of the algorithm (8), we cannot choose the step lengths α_k with the full generality allowed by the existing convergence results for subgradient optimization algorithms. We will consider instances of the divergent series step length rule

$$\alpha_k > 0, \quad \forall k, \qquad \lim_{k \to \infty} \alpha_k = 0, \qquad \sum_{k=1}^{\infty} \alpha_k = \infty, \qquad \sum_{k=1}^{\infty} \alpha_k^2 < \infty. \tag{10}$$

The known convergence characteristic of subgradient optimization is summarized below.

Theorem 3.1 (Convergence of subgradient optimization algorithms) *Let $\{u^k\}$ be generated by the algorithm (8), (9), (10). If the sequence $\{\xi_\psi(u^k)\}$ is bounded, then the sequence $\{u^k\}$ converges to a solution to (P).*

Proof See, e.g., Reference 22. ∎

We remark here the important fact that the boundedness of the sequence $\{\xi_\psi(u^k)\}$ is guaranteed by letting the value of $\sup_k\{\alpha_k\}$ be sufficiently small, since the set $\mathcal{U}^*$ of optimal solutions is bounded, cf. Reference 21, page 27. (The maximal step length allowed is, however, unknown *a priori* in general.)

The projection and subgradient generation steps of the algorithm are studied next.

3.2 Generating subgradients

Using a well-known result on subgradients of max-functions[12] ,

$$\partial\psi(\overline{u}) = \texttt{conv} \left\{ \nabla_u \Pi(\overline{u}, \overline{h}) \,\middle|\, \overline{h} \in \arg\max_{h \in \mathcal{H}} \Pi(\overline{u}, h) \right\}, \qquad \overline{u} \in \mathcal{U}.$$

At u^k, a subgradient of ψ is therefore given by

$$\xi_\psi(u^k) = K(h^k)u^k - f, \tag{11}$$

where

$$h^k \in \mathcal{H}(u^k) \stackrel{\text{def}}{=} \arg\max_{h \in \mathcal{H}} \Pi(u^k, h). \tag{12}$$

[Hence, h^k is a solution to the linear program defining the value $\psi(u^k)$.]

To solve the linear program (12), we observe that it may be written as

$$\max_{h \in \mathcal{H}} \sum_{i=1}^{m} c_i^k h_i, \tag{13}$$

where

$$c_i^k = {u^k}^T K_i u^k \geq 0, \qquad i = 1, 2, \ldots, m, \tag{14}$$

by the positive semidefiniteness of each matrix K_i. From the structure of the feasible set $\mathcal{H}$, it follows that (13) is a continuous knapsack problem, which belongs to the class of problems solvable by greedy algorithms[23].

Greedy algorithm for (13)

Step 0 (Initialization): Let $\hat{h} = \underline{h}$.

Step 1 (Iterative step): Repeat the following until $1^T\hat{h} = V$:

1. Let $\tilde{\imath} \in \arg\max_{i\in\{1,\ldots,m\}} \left\{ c_i^k \,\middle|\, \hat{h}_i < \overline{h}_i \right\}$.
2. Let $\hat{h}_{\tilde{\imath}} := \hat{h}_{\tilde{\imath}} + \min\{V - 1^T\hat{h}, \overline{h}_{\tilde{\imath}} - \hat{h}_{\tilde{\imath}}\}$.

The solution to (13) then is $h^k = \hat{h}$.

To carry out Step 1 it is necessary to sort the m elements c_i^k in descending order. The best complexity of a sorting algorithm is known to be $O(m \log m)$, (see chapter 8 in Reference 24). We may, however, observe that as u^k approaches a solution to (P), the variations in the coefficients c_i^k will be smaller and smaller; thus, the permutation defined at iteration k will be a near-optimal one at iteration $k+1$ for k large enough, so by always using the previous permutation as the starting solution, the practical performance can be expected to be better than $O(m \log m)$.

In the interesting special case when $\underline{h}_i = 0$ for all i and all $\overline{h}_i$ are equal (to, say, $\check{h}$), we can, however, do even better. Indeed, to solve the linear program (13) it is only necessary to select the $\lceil V/\check{h} \rceil$:th largest element of the vector c^k, where $\lceil x \rceil$ denotes the smallest integer larger than or equal to x, and (as a byproduct of this procedure) partition the remaining elements into two groups depending on if they would precede the $\lceil V/\check{h} \rceil$:th largest element or not in a completely sorted list. The solution is then obtained by letting $h_i^k = \check{h}$ for all i corresponding to elements in c^k that precede the breakpoint element, and to let that element receive the remainder of the total volume $V - \check{h}(\lceil V/\check{h} \rceil - 1)$. The complexity of this selection operation [and therefore the complexity of the linear program (13)] is $O(m)$, (see chapter 8.7 in Reference 24). (When $\check{h} \geq V$ the selection operation reduces to finding the maximum element in c^k, and it follows that

$$\psi(u^k) = \frac{V}{2} \max_{i\in\{1,\ldots,m\}} \left\{ {u^k}^T K_i u^k - f^T u^k \right\},$$

which is the well-known form of ψ for truss topology optimization with free design variables, cf. section 6 in Reference 9.)

3.3 Performing projections

Given $u^k \in \mathcal{U}$, a new displacement vector u^{k+1} is calculated by projecting $u^k - \alpha_k \xi_\psi(u^k)$ onto $\mathcal{U}$. Due to the quasi-orthogonality of the matrix C, we can show that the Euclidean projection of any vector in $\Re^n$ onto $\mathcal{U}$ is performed (essentially) by projecting a certain vector onto the nonnegative orthant $\Re^r_+$. This is a significant computational improvement over a Euclidean projection onto a general polyhedral set, which is an unstructured quadratic programming problem. The projection of a point $z \in \Re^r$ onto $\Re^r_+$ is given by $P_{\Re^r_+}(z) = \max\{0, z\}$, where the maximum is taken component-wise.

Lemma 3.1 *Suppose that C is quasi-orthogonal. Then, for any $x \in \Re^n$,*

$$P_{\mathcal{U}}(x) = x - C^T P_{\Re^r_+}(Cx - g). \tag{15}$$

Proof We first establish that the right hand side of (15) is admissible, i.e., belongs to $\mathcal{U}$. With the simplified notation $P = P_{\Re^r_+}(Cx - g)$ we have

$$C(x - C^T P) = Cx - CC^T P = Cx - P.$$

Examining the ith component, one gets

$$(C(x - C^T P))_i = \nu_i^T x - P_{\Re_+}(\nu_i^T x - g_i) \leq g_i,$$

where the inequality follows by separately considering the two possible cases of the sign of $\nu_i^T x - g_i$, and hence $x - C^T P \in \mathcal{U}$.

We next show that (15) is a projection, i.e., that (7) is satisfied. Let $y \in \mathcal{U}$ be arbitrary. We have

$$[x - (x - C^T P)]^T [y - (x - C^T P)] = P^T C(y - x + C^T P)$$

$$= P^T (Cy - g) - P^T (Cx - g) + P^T CC^T P. \tag{16}$$

Since $P \geq 0$ and $Cy - g \leq 0$, the first term in (16) is clearly nonpositive. Concerning the second one, we have that $P^T(Cx - g) = \|P\|^2$. In fact, from Definition 3.1, the projection of $Cx - g$ on $\Re^r_+$ is characterized by

$$(Cx - g - P)^T (\overline{y} - P) \leq 0, \qquad \forall \overline{y} \in \Re^r_+,$$

from which $P^T(Cx - g) = \|P\|^2$ follows by choosing $\overline{y} = 2P$ and $\overline{y} = \frac{1}{2}P$. Proceeding with (16) we have that

$$[x - (x - C^T P)]^T [y - (x - C^T P)] \leq -\|P\|^2 + P^T CC^T P = 0, \tag{17}$$

and we have established (7) with $P_{\mathcal{U}}(x) = x - C^T P$, i.e., (15). ∎

4 CONVERGENCE OF THE DESIGNS

4.1 The averaging scheme

In the course of the subgradient optimization algorithm, candidate designs are generated when calculating subgradients of ψ. The sequence $\{h^k\}$ given by (12) is, however, not optimizing, since $\mathcal{H}(u^*)$ is not a singleton set in general. (The sequence $\{h^k\}$ consists of extreme points of $\mathcal{H}$, and it is very unlikely that the optimal design is an extreme point of $\mathcal{H}$.) An equivalent explanation is that in general ψ is nondifferentiable at u^*.

That an optimal solution to (D) cannot be found by solving (12) at an optimal solution to (P) is well-known in the use of Lagrangean relaxation in linear programming, where it is known as the *non-coordinability phenomenon*[25]. To obtain an optimal solution to (D) by utilizing the subproblem solutions h^k, one may consider using the Dantzig–Wolfe[26] decomposition principle, in which one seeks an optimal convex combination of the subproblem solutions generated; this procedure is, however, computationally expensive. An alternative is to introduce coordinability (i.e., differentiability) by introducing a strictly concave function of h in Π (e.g., Reference 11). Although this modification has the effect that the corresponding sequence $\{h^k\}$ of subproblem solutions is optimizing in the limit, one disadvantage is that the simple linear program (12) is replaced by the solution of a computationally more demanding strictly convex program at each iteration.

Shor[21] shows that, in the context of Lagrangean relaxation of linear programs, one can obtain an optimal solution by forming predetermined convex combinations of the subproblem solutions (using the step lengths α_k of the subgradient optimization scheme). The convex combinations used in his scheme have the practical disadvantage that the first subproblem solutions (which obviously are of lesser quality than latter ones) receive the largest weights. Larsson and Liu[27] show that one may replace this weighting scheme by a simple averaging of the subproblem solutions, provided that the step lengths α_k used are of a special form. Based on numerical experiments, their conclusion is that the averaging of the subproblem solutions is a much better strategy than that proposed by Shor. Subsequently, the averaging scheme is applied to a specially structured convex program[28], also in the context of Lagrangean relaxation.

The approach taken here is to extend the use of the averaging scheme of Larsson and Liu to obtain an optimal design vector in the saddle problem $(S\Pi)$.

We thus define the auxiliary sequence $\{h(l)\}$ by

$$h(l) = \frac{1}{l} \sum_{k=1}^{l} h^k, \qquad l = 1, 2, \ldots, \tag{18}$$

where h^k is given by (12). (In practice, we calculate $h(l)$ recursively as $h(l) = (1 - \frac{1}{l})h(l-1) + \frac{1}{l}h^l$.) Clearly, $h(l) \in \mathcal{H}$ for all l. We note here that it is unnecessary to introduce the averaging scheme from the very beginning of the subgradient optimization scheme, and it was in References 27,28 indeed observed to be of practical benefit to postpone the invocation of the averaging scheme to a later stage in the algorithmic process. For the simplicity of presentation of the theoretical results we shall, however, assume that the averaging scheme is used from the start of the subgradient optimization scheme.

In the subgradient optimization algorithm, as in Reference 27 we choose the step lengths α_k according to the formula

$$\alpha_k = \frac{1}{a+bk}, \qquad a \geq 0, \qquad b > 0, \qquad k = 1, 2, \ldots, \tag{19}$$

which clearly satisfies (10). [Since the sequence $\{\alpha_k\}$ is decreasing, it also satisfies the upper boundedness requirement (cf. the discussion following Theorem 3.1) if the value of b is large enough.]

4.2 Convergence results

By the boundedness of $\mathcal{H}$, the sequence $\{h(l)\}$ has at least one accumulation point. We will show that each such point h^∞, together with the solution u^* obtained by the subgradient optimization algorithm applied to (P), is a saddle point of Π.

Our first theorem establishes that any accumulation point of the sequence $\{h(l)\}$ solves the linear program (12) at $u = u^*$; the averaging therefore, in a sense, introduces a coordinability in the limit.

Theorem 4.1 ($\Pi(u^*, h) \leq \Pi(u^*, h^\infty)$ for all $h \in \mathcal{H}$) *Let the sequence $\{u^k\}$ be generated by the algorithm (8), (9), (10), and assume that the sequence $\{\xi_\psi(u^k)\}$ is bounded. Let u^* be the limit of $\{u^k\}$, and h^∞ be any accumulation point of the sequence $\{h(l)\}$ defined by (18). Then,*

$$h^\infty \in \arg\max_{h\in\mathcal{H}} \Pi(u^*, h).$$

Proof We wish to establish that

$$\frac{1}{2} u^{*T} K(h^\infty) u^* \geq \frac{1}{2} u^{*T} K(h) u^*, \qquad \forall h \in \mathcal{H}.$$

Let h^∞ be an accumulation point of the sequence $\{h(l)\}$, corresponding to a convergent subsequence $\mathcal{L}$. We then have that

$$\frac{1}{2} u^{*T} K(h^\infty) u^* = \frac{1}{2} \lim_{l \xrightarrow{\mathcal{L}} \infty} \frac{1}{l} \sum_{k=1}^{l} u^{*T} K(h^k) u^*.$$

Consider

$$\frac{1}{2} \lim_{l \xrightarrow{\mathcal{L}} \infty} \frac{1}{l} \sum_{k=1}^{l} (u^* - u^k)^T K(h^k)(u^* + u^k). \tag{20}$$

We have that

$$|(u^* - u^k)^T K(h^k)(u^* + u^k)| \leq \|u^* - u^k\| \cdot \|K(h^k)\| \cdot \|u^* + u^k\|,$$

and the right-hand-side tends to zero, since both $\{u^k\}$ and $\{K(h^k)\}$ are bounded. This implies that the value of the term in (20), when l tends to ∞, is 0, and that

$$\begin{aligned}
\frac{1}{2}u^{*T}K(h^\infty)u^* &= \frac{1}{2}\lim_{l\xrightarrow{\mathcal{L}}\infty}\frac{1}{l}\sum_{k=1}^{l} u^{*T}K(h^k)u^* \\
&= \frac{1}{2}\left\{\lim_{l\xrightarrow{\mathcal{L}}\infty}\frac{1}{l}\sum_{k=1}^{l} {u^k}^T K(h^k)u^k\right. \\
&\quad \left. + \lim_{l\xrightarrow{\mathcal{L}}\infty}\frac{1}{l}\sum_{k=1}^{l}(u^* - u^k)^T K(h^k)(u^* + u^k)\right\} \\
&= \frac{1}{2}\lim_{l\xrightarrow{\mathcal{L}}\infty}\frac{1}{l}\sum_{k=1}^{l} {u^k}^T K(h^k)u^k \\
&\geq \frac{1}{2}\lim_{l\xrightarrow{\mathcal{L}}\infty}\frac{1}{l}\sum_{k=1}^{l} {u^k}^T K(h)u^k \\
&= \frac{1}{2}u^{*T}K(h)u^*, \qquad \forall h \in \mathcal{H},
\end{aligned}$$

where the inequality stems from the fact that h^k solves (12) at iteration k. ■

To show that (u^*, h^∞), for any accumulation point h^∞ of the sequence $\{h(l)\}$, is a saddle point of Π, it remains to show that

$$u^* \in \arg\min_{u\in\mathcal{U}} \Pi(u, h^\infty).$$

Lemma 4.1 *Let h^∞ be an accumulation point of the sequence $\{h(l)\}$, defined by (18), corresponding to a convergent subsequence $\mathcal{L}$. Under the assumptions of Theorem 4.1,*

$$\left\{\frac{1}{l}\sum_{k=1}^{l}\xi_\psi(u^k)\right\} \xrightarrow{\mathcal{L}} K(h^\infty)u^* - f. \tag{21}$$

Proof The proof is a modification of that of a part of Theorem 3 of Reference 27, stated for a Lagrangean relaxation of a linear program. We first have

$$\frac{1}{l}\sum_{k=1}^{l}\xi_\psi(u^k) = \frac{1}{l}\sum_{k=1}^{l}(K(h^k)u^k - f)$$

$$
\begin{aligned}
&= \frac{1}{l}\sum_{k=1}^{l}(K(h^k)u^* - f + K(h^k)(u^k - u^*)) \\
&= K(h(l))u^* - f + \frac{1}{l}\sum_{k=1}^{l} K(h^k)(u^k - u^*). \qquad (22)
\end{aligned}
$$

Since $\mathcal{H}$ is bounded, $\sup_k\{\|K(h^k)\|\}$ clearly is finite. As $l \to \infty$, the last term of (22) therefore tends to zero, and we obtain (21). ∎

Lemma 4.2 *Suppose that $\{u^k\}$ converges, and let the sequence $\{\alpha_k\}$ of step lengths be given by (19). Then,*

$$
\left\{\frac{1}{l}\sum_{k=1}^{l}\frac{u^{k+1}-u^k}{\alpha_k}\right\} \to 0.
$$

Proof Follows from exactly the same arguments as those in the proof of Theorem 3 in Reference 27. ∎

Theorem 4.2 ($\Pi(u^*, h^\infty) \le \Pi(u, h^\infty)$ for all $u \in \mathcal{U}$) *Let the sequence $\{u^k\}$ be generated by the algorithm (8), (9), (19), and assume that the sequence $\{\xi_\psi(u^k)\}$ is bounded. Let u^* be the limit of $\{u^k\}$. Let h^∞ be any accumulation point of the sequence $\{h(l)\}$ defined by (18). Then,*

$$
u^* \in \arg\min_{u\in\mathcal{U}} \Pi(u, h^\infty). \qquad (23)
$$

Proof To establish the theorem, we will show that the necessary and sufficient Karush–Kuhn–Tucker conditions for the convex quadratic program (23) hold.

The Karush–Kuhn–Tucker conditions for the optimality of u^* in (23) are (cf. Theorem 4.3.7 in Reference 29)

$$
\begin{aligned}
K(h^\infty)u^* - f + C^T\lambda^* &= 0, && (24)\\
\lambda^* &\ge 0, && (25)\\
\lambda^{*T}(Cu^* - g) &= 0, && (26)\\
Cu^* - g &\le 0. && (27)
\end{aligned}
$$

From (8) and Lemma 3.1 we have, for each k, that

$$
\begin{aligned}
u^{k+1} &= P_{\mathcal{U}}\left\{u^k - \alpha_k\xi_\psi(u^k)\right\} \\
&= u^k - \alpha_k\xi_\psi(u^k) - C^T P_{\Re^r_+}\left\{Cu^k - g - \alpha_k C\xi_\psi(u^k)\right\}.
\end{aligned}
$$

By rearranging terms, summing the equality for all k from 1 to l, and dividing the result by l, we obtain

$$\frac{1}{l}\sum_{k=1}^{l}\xi_\psi(u^k) = -\frac{1}{l}\sum_{k=1}^{l}\frac{u^{k+1}-u^k}{\alpha_k} - C^T\frac{1}{l}\sum_{k=1}^{l}P_{\Re^r_+}\left\{\frac{Cu^k-g}{\alpha_k} - C\xi_\psi(u^k)\right\}. \tag{28}$$

Define

$$\lambda^l = \frac{1}{l}\sum_{k=1}^{l}P_{\Re^r_+}\left\{\frac{Cu^k-g}{\alpha_k} - C\xi_\psi(u^k)\right\}, \tag{29}$$

whereby (28) becomes

$$\frac{1}{l}\sum_{k=1}^{l}\xi_\psi(u^k) = -\frac{1}{l}\sum_{k=1}^{l}\frac{u^{k+1}-u^k}{\alpha_k} - C^T\lambda^l. \tag{30}$$

By Lemmata 4.1 and 4.2, (30) implies that

$$\{C^T\lambda^l\} \xrightarrow{\mathcal{L}} f - K(h^\infty)u^*; \tag{31}$$

invoking the quasi-orthogonality property of C yields that

$$\{\lambda^l\} \xrightarrow{\mathcal{L}} \lambda^* \stackrel{\text{def}}{=} C(f - K(h^\infty)u^*), \tag{32}$$

which is nonnegative by (29). With this choice of multiplier values, (25) holds. Moreover, from (31) and (32), we obtain that (24) holds, since the limit of $\{C^T\lambda^l\}$ is unique.

To establish complementarity, i.e., (26), suppose that $\lambda^*_j > 0$ for some $j \in \{1,\dots,r\}$. Then, from (29) and (32), we obtain

$$\frac{1}{l}\sum_{k=1}^{l}P_{\Re_+}\left\{\frac{\nu_j^Tu^k-g_j}{\alpha_k} - \nu_j^T\xi_\psi(u^k)\right\} \xrightarrow{\mathcal{L}} \lambda^*_j > 0, \tag{33}$$

where $\nu_j^Tu^k - g_j = (Cu^k-g)_j$. The average of a sequence of nonnegative real numbers converges to a strictly positive limit only if there exists a convergent subsequence of these numbers which are strictly positive. Let $\mathcal{K}$ denote such a subsequence. Then

$$\frac{\nu_j^Tu^k-g_j}{\alpha_k} - \nu_j^T\xi_\psi(u^k) > 0 \Longleftrightarrow \nu_j^Tu^k - g_j > \alpha_k\nu_j^T\xi_\psi(u^k), \quad \forall k \in \mathcal{K}.$$

By assumption, $\{\alpha_k\} \xrightarrow{\mathcal{K}} 0$, $\{u^k\} \xrightarrow{\mathcal{K}} u^* \in \mathcal{U}$, and $\{\xi_\psi(u^k)\}_{k\in\mathcal{K}}$ is bounded, so

$$\{\nu_j^Tu^k - g_j\} \xrightarrow{\mathcal{K}} \nu_j^Tu^* - g_j \geq 0. \tag{34}$$

Since the condition (27) holds by the construction of $\{u^k\}$, (34) implies that

$$\nu_j^T u^* - g_j = (Cu^* - g)_j = 0.$$

Since j was arbitrary, we have therefore obtained that (26) holds. ∎

We have thus established that any pair (u^*, h^∞) defined by the solution u^* to (P) given by the subgradient optimization algorithm (8), (9), (19), and an accumulation point h^∞ of the auxiliary sequence $\{h(l)\}$ defined by the simple average of the solutions to (12), is a saddle point of Π. We summarize the convergence properties of the sequences $\{u^k\}$ and $\{h(l)\}$ below.

Theorem 4.3 (Convergence properties of the overall scheme) *Let the sequence $\{u^k\}$ be generated by the algorithm (8), (9), (19), and assume that the sequence $\{\xi_\psi(u^k)\}$ is bounded. Let the sequence $\{h(l)\}$ be defined by (18). Then,*

$$\{u^k\} \to u^* \in \mathcal{U}^*,$$

and

$$\{d_{\mathcal{H}^*}(h(l))\} \to 0,$$

where

$$d_{\mathcal{H}^*}(x) = \inf_{h \in \mathcal{H}^*} \|x - h\|.$$

Proof The first result is Theorem 3.1. The second result follows from Theorems 4.1 and 4.2, and the boundedness of $\mathcal{H}$ (cf. also Theorem 14.1.4 in Reference 30). ∎

4.3 A lower bound

The convergence results of Section 4.2 are *asymptotic*, that is, a solution to $(S\Pi)$ is obtained only in the limits of the sequences $\{u^k\}$ and $\{h(l)\}$. For the algorithm presented above to be practical, it therefore needs to be augmented with the iterative inspection of one or more criteria for finite termination. It is important that the criteria used are based on measures of the deviation from optimality of the pair $(u^k, h(l))$ that are *well-defined*, in the sense that the measures tend to zero as $(u^k, h(l))$ tends to a saddle point. Further, any measure used should be relatively easy to evaluate. (Should the computational cost of evaluating any measure be high, it must otherwise be amortized by invoking the corresponding termination test with regular intervals only.)

In *differentiable* convex optimization several such measures are available. As an example, we consider the use of the relative error in the objective value of the convex program. Any feasible solution provides an upper bound on the optimal value; a lower bound is obtained, for example, by solving an approximation of the original problem, in which the objective is linearized at the current iterate. (The latter is known as the Frank–Wolfe bound.) The lower bound is optimizing, i.e., as the iterate tends to a solution, the lower bound tends to the optimal value of the program, and hence the relative error tends to zero. This lower bound is, however, not necessarily optimizing if the solution is a point of non-differentiability[31].

The value $\psi(u^k)$, which is supplied automatically from the solution of the linear program (12), is an upper bound on Π^*. The perhaps obvious choice of lower bound on Π^* is $\varphi(h)$, evaluated at $h = h(l)$; although this is of course a possible choice, we have earlier argued that the evaluation of φ is too expensive computationally to be practical in a large-scale setting. We propose to replace it by its Lagrangean lower bound, where the vector of multipliers is an approximate dual solution to the quadratic program defining $\varphi(h^\infty)$, i.e., an approximation to the contact force magnitude λ^*. Given an iterate u^k and a tentative design $h(l)$, the lower bound is

$$\underline{\varphi}(u^k, h(l)) = \inf_{u \in \Re^n} \left\{ \Pi(u, h(l)) + {\lambda^k}^T (Cu - g) \right\}, \tag{35}$$

where we let

$$\lambda^k = P_{\Re^r_+}(C[f - K(h(l))u^k]). \tag{36}$$

(The value of $\underline{\varphi}(u^k, h(l))$ is obtained through the solution of a sparse system of linear equations.) In the below theorem, we demonstrate that the lower bound $\underline{\varphi}(u^k, h(l))$ is well-defined.

Theorem 4.4 (A lower bound) *Let the sequence $\{u^k\}$ be generated by the algorithm (8), (9), (19), and assume that the sequence $\{\xi_\psi(u^k)\}$ is bounded. Let the sequence $\{h(l)\}$ be defined by (18). Let the sequence $\{\underline{\varphi}(u^k, h(l))\}$ be generated by (35), (36). Then,*

$$\underline{\varphi}(u^k, h(l)) \le \Pi^*, \quad \forall k, l.$$

Assume further that the function

$$\underline{\tilde{\varphi}}(h, \lambda) = \inf_{u \in \Re^n} \left\{ \Pi(u, h) + \lambda^T (Cu - g) \right\} \tag{37}$$

is finite on a relatively open, convex set in $\Re^{m+r}$ which includes all except at most finitely many elements of the sequence $\{(h(l), \lambda^k)\}$ and all its accumulation points. Then,

$$\{\underline{\varphi}(u^k, h(l))\} \to \Pi^*.$$

Proof The value $\underline{\varphi}(u^k, h(l))$ is a lower bound on $\varphi(h(l))$ for every k, l by weak duality (e.g., Theorem 6.2.1 in Reference 29); the first result then follows immediately.

The function $\underline{\tilde{\varphi}}$ is concave on $\Re^{m+r}$, since it is the pointwise supremum of a collection of concave (actually affine) functions. It is also finite on a relatively open, convex set $S \subset \texttt{dom}\,\underline{\tilde{\varphi}}$, and, by Theorem 10.1 in Reference 32, therefore continuous on this set. By hypothesis, $(h(l), \lambda^k) \in S$ for all sufficiently large l, k. By the continuity on $\Re^{n+m}$ of the mapping

$$(u, h) \mapsto (h, \lambda(u, h)) = (h, P_{\Re^r_+}(C[f - K(h))u]),$$

it follows that for large enough l, k, the sequence $\{(u^k, h(l))\}$ lies in the domain of continuity of the function

$$(u, h) \mapsto \underline{\varphi}(u, h) = \underline{\tilde{\varphi}}(h, \lambda(u, h)).$$

The sequence $\{\underline{\varphi}(u^k, h(l))\}$ is clearly bounded in $\Re^1$. Let φ^* be any of its accumulation points. Then, due to the boundedness of $\mathcal{H}$ and continuity of $(u, h) \mapsto \underline{\varphi}(u, h)$, there exists an accumulation point h^∞ of $\{h(l)\}$ such that

$$\underline{\varphi}(u^*, h^\infty) = \varphi^*,$$

where u^* is the limit point of $\{u^k\}$. The proof of Theorem 4.2 establishes that $\lambda^* = P_{\Re^r_+}(C[f - K(h^\infty)u^*])$ is a vector of Lagrange multipliers for the constraints $Cu \leq g$ in (23). Strong duality (e.g., Theorem 6.2.4 of Reference 29) then establishes that $\underline{\varphi}(u^*, h^\infty) = \Pi^*$. Hence, $\varphi^* = \Pi^*$, and since φ^* was an arbitrary point of accumulation of $\{\underline{\varphi}(u^k, h(l))\}$, the second result follows. ■

Note that the assumption concerning the second result holds automatically when $\underline{h} > 0$, since then $u \mapsto \Pi(u, h)$ is coercive for every $h \in \mathcal{H}$. (The relatively open, convex set can, e.g., be chosen as

$$S = \left\{ (h, \lambda) \in \Re^{m+r} \;\middle|\; \sum_{i=1}^{m} h_i = V, \quad \underline{h}/2 < h < 2\overline{h}, \quad \lambda > -\varepsilon 1 \right\},$$

where $\varepsilon > 0$ is arbitrary and 1 is the vector of ones in $\Re^r$.) In practice, the computation of a lower bound is made only late in the process of the algorithm, when not much progress is longer made.

5 NUMERICAL RESULTS

In this section some numerical experience with the proposed algorithm and analyses and mechanical evaluations of the solutions to $(S\Pi)$ are reported. In addition to observations on the practical performance of the subgradient optimization algorithm and on the averaging scheme, we investigate the mechanical features of the optimal sheets. Especially demonstrated are the improvements in the displacements and the von Mises stresses compared with those of sheets with uniform thickness; we also compare their respective contact stresses. The analysis shows that when the obstacle is remote, the optimal structure avoids contact, while for obstacles closer to the elastic body the optimal topology alters drastically in order to improve the contact pressures and displacements.

5.1 PROBLEM DATA AND ALGORITHM IMPLEMENTATION

To be able to solve large-scale instances of $(S\Pi)$, it is crucial that both the storage and computational requirements do not increase rapidly with the size of the problem. We begin by showing that the storage requirements are linear in the parameters m, n and r, and that the same holds for the execution of one iteration of the subgradient optimization algorithm. For our analysis purposes, it is desirable to be able to solve $(S\Pi)$ for a small granularity of the finite element mesh; our numerical experience is that the number of iterations needed to obtain a given accuracy is a function which is nearly linear in the parameter m.

The matrices $C \in \Re^{r\times n}$ and $K_i \in \Re^{n\times n}$, $i = 1, 2, \ldots, m$, constitute problem data that potentially has a large storage requirement. In the case of the matrix C, we recall that its rows correspond to unit normals; thus, at most two elements of each row are non-zero, and it suffices with $2r$ real elements to store this matrix. (We need also $2r$ integers to store the elements' column positions.) For a linear triangle, the element stiffness matrix is of the order 6×6 (cf. pages 90-91 in Reference 33). Also taking the symmetry of the matrix K_i into account, only 21 real elements are needed to store it. Moreover, if two edges of each triangle are coaligned with the axes of the Cartesian coordinate system, 17 non-zero elements are sufficient for each element. When performing a topology optimization, it is very hard to determine *a priori* where the mesh needs to be comparably fine, and where a rather coarse mesh is sufficient; using a uniform mesh, we need only store 17 reals in total for all the matrices K_i. (Also, $34m$ integers are necessary to store the positions of the elements.) In total, therefore, utilizing the sparsity of the matrices C and K_i, $i = 1, 2, \ldots, m$, only $2r + 17$ real elements (and $2r + 34m$ integers) of storage are required. If the lower bounds $\underline{h}_i = 0$ and the upper bounds $\overline{h}_i = \check{h}$ for all i (as we shall always assume), then the storage requirement for one instance of the problem $(S\Pi)$ is only $n + 3r + 19$ reals (and $2r + 34m + 3$ integers).

One iteration of the subgradient optimization algorithm consists of three main steps: the construction and solution of the linear program (12), the calculation of the subgradient (11), and the updating operation (8). The linear program is set up by calculating the coefficients c_i^k [cf. (14)]; this is performed through $62m$ floating point operations (flops). As analyzed in Section 3.2, the linear program (12) is solved in $O(m)$ time (i.e., the number of operations needed is bounded above by a polynomial of degree one in m); the subsequent construction of the solution h^k is clearly also linear in m. The subgradient is then calculated through $84m$ flops. Finally, the updating step, in which the projection is performed utilizing Lemma 3.1, is performed through $6r + 3n$ flops. Hence, one iteration of the subgradient scheme has a *linear time complexity* in m, n and r.

Ben-Tal and Nemirovskii[34,35] have considered complexity issues for interior point methods for the *truss* problem. They obtain a complexity per iteration which is linear in m and cubic in n, a desirable result when $n \ll m$. In our setting, i.e., for FE problems, n/m is nearly constant; in this case, the complexity per iteration is $O(m^3)$.

We next discuss the remaining parts of the overall algorithm and provide some numerical experience.

5.2 Numerical experience

We implemented the algorithm in Fortran-77 and executed it on an HP 9000 Model 712/60 (Envizex X station), using single (about seven digits) precision arithmetic and second level compiler optimization.

The initial solution, $u^1 \in \mathcal{U}$, is generated by combining two heuristics. First, a (mechanically motivated) guess is made on the displacement field to be expected for a uniform sheet. A low order polynomial is used to imitate a plausible deformed shape, and it is scaled depending on the load and volume magnitudes. Second, we use a few steps of a descent approach for differentiable optimization problems: instead of letting u^{k+1} be determined by the projection (8), a line search between the point u^k and the projection is introduced, in order to minimize the value of ψ; this is performed using the Armijo step length rule (e.g., section 8.3 in Reference 29). In the case where ψ is differentiable at u^k (and this is true with probability one according to Rademacher's theorem), this strategy yields a decreased value of ψ, and is in differentiable optimization a slight extension of the Goldstein-Levitin-Polyak gradient projection algorithm[36]. We use this heuristic for a maximum number of 30 iterations. (In the tests performed, no problems associated with the possible nondifferentiability of ψ were ever encountered.) For the large-scale problems analyzed in this paper, the improvements over the use of simpler starting point generation strategies [such as the formula $u^1 = P_{\mathcal{U}}(0)$] of the combined heuristic were moderate, however.

The proper choice of step lengths α_k was investigated through a series of experiments on small problems. The conclusion drawn from these experiments is that a small initial step is necessary to avoid an unbounded sequence of subgradients (see the discussion following Theorem 3.1), and that the subsequent steps should decrease slowly. While choices in the order of $a = 10^5$ and $b = 10^3$ work well for small problems, in a large-scale setting the initial step needs to be even shorter; the problems presented in this paper have been solved using $a = 5 \cdot 10^6$ and $b = 8 \cdot 10^2$. Their values need to be larger for stiffer structures than those studied here.

As discussed in Section 4.1 the averaging scheme should not be initialized at the beginning of the subgradient algorithm, but be postponed until the displacement variables are near-optimal. In the applications of averaging schemes described in References 27,28, the dual functions are piece-wise linear, and it is there shown that the corresponding solutions h^k after a finite number of iterations are among the subproblem solutions at u^*. We have not been able to establish such a result for the problem $(S\Pi)$, where the function ψ is piece-wise quadratic, but have drawn similar conclusions regarding the best initialization policy. What seems to maximize the quality of the design at termination for large-scale problems is to initialize the averaging scheme as late as after about 90–95% of the total number of iterations.

As a practical termination criterion we have chosen a small difference between the lower and upper bounds on the saddle value. Terminating when $(\mathrm{UBD} - \mathrm{LBD})/|\mathrm{LBD}| \leq 0.01$, where $\mathrm{UBD} = \min_k\{\psi(u^k)\}$ and $\mathrm{LBD} = \max_k\{\underline{\varphi}(u^k, h(l))\}$ was most often found to be quite satisfactory; in fact, in few cases does a relative error of much less than one percent produce a noticeable alteration in the design. (This is an indication that the accuracy of both the design and the displacements are, at termination, better than shown by the relative error.) For generating the lower bound, we solve the unconstrained quadratic program (35) by the conjugate direction algorithm of Fletcher and Reeves (e.g., section 8.8 in Reference 29), always using the current iterate u^k as the starting point. (The use of a direct method for the corresponding linear system of equations is precluded by the possible singularity of the coefficient matrix $K(h(l))$, and that most direct methods do not take sparsity into account efficiently. Conjugate direction methods also have the advantage of being very memory efficient; the Fletcher-Reeves algorithm requires only one additional vector of size n.) Even without restarts or other modifications for speeding it up, the algorithm is efficient, and in general finds a solution (with a residual proportional to the norm of the right-hand-side times the square root of the machine precision) within $2n$ iterations. A lower bound is generated only very late in the overall procedure, and in total less than ten times. (For the largest problems solved, the solution time for generating one lower bound is in the order of five minutes.)

The problems solved in this paper are presented in Section 5.3. The complexity figures presented above indicates that for large-scale problems, the construction and solution of the linear program (12) and the construction of the subgradient (11) are the most time consuming parts of the algorithm. Indeed, in our experience, they completely dominate in computational complexity. In our implementation, the CPU time per thousand iterations needed for these parts of the algorithm is roughly $0.023m$ second. (The largest example solved in this paper takes, on average, 3.9 minutes per thousand iterations, meaning that each iteration requires about one fifth of a second.)

The number of iterations needed to reach a given accuracy was found to be nearly linear in m. On average, about $400\text{–}500 \cdot 10^3$ iterations were spent on each example in Table I (see below), usually reaching a relative error in the objective below one percent. Combining this result with the linear complexity of each iteration yields a practical complexity which is close to quadratic in m. (Problems of size $m = 1000$ are solved within 15–20 minutes, while solving problems of the size $m = 10,000$ takes in the order of at least 24 hours.) It should be noted that according to a result of Poljak[19], the subgradient optimization algorithm has theoretically a sublinear convergence rate, which is among the weakest possible.

Our numerical experience with the algorithm is reported in fuller detail in Reference 37.

5.3 Computed solutions and mechanical evaluation

We solve $(S\Pi)$ for the symmetric sheet shown in Fig. 1.

[Place of Fig. 1]

The elasticity coefficients are given by plane stress and for the Young modulus and Poisson ratio, the values $E = 2.1 \cdot 10^5$ MPa and $\nu = 0.3$, respectively, are used. The load t is a downward constant line load of 10 N/mm and the gap function (for the lower left edge) is

$$g(x_1) = k_1 + k_2 \cdot x_1^4, \qquad 0 \le x_1 \le 50, \tag{38}$$

where we use $k_2 = 6.4 \cdot 10^{-10}$ (mm)$^{-3}$ and vary the value of k_1. We choose $\underline{h}_i = 0$ and $\overline{h}_i = |\Omega|\alpha_U/m$ where $\alpha_U = 5$ mm, for all i, and consider the five cases shown in Table I.

TABLE I
Problem data for case 1–5.

Case	m	n	k_1 (mm)	Volume (% of upper design bound)	Π^*
1	4096	4160	0.0003	58.0	−0.1194
2	5040	5110	0.0000	58.0	−0.1166
3	7568	7654	−0.0003	58.0	−0.0841
4	10080	10185	−0.0015	58.0	0.2587
5	7568	7654	−0.0015	7.5	−0.4255

To represent optimal thickness distributions we use a grayscale, where black and white represent a 5 mm and 0 mm thickness, respectively. Checkerboards are treated as described in Appendix A. Fig. 2 shows optimal designs for cases 1 and 2. Clearly, in case 1 no contact occurs, and Fig. 2 (a) shows a traditional minimum compliance solution.

[Place of Fig. 2]

For all cases we also solved the equilibrium problem for a uniform thickness sheet with the same total volume, in order to compare the displacements, von Mises stresses and contact pressures. (This was done using our method by simply avoiding the sorting procedure, and the CPU time was hereby roughly halved). Fig. 3 shows the deformed configurations for both optimal and uniform thickness sheets for case 2.

[Place of Fig. 3]

In this figure, the gray scale illustrates von Mises effective stress. The appearance is typical: the stresses in the optimal sheet is much more evenly distributed, and the (downward) displacements are smaller than in the uniform sheet.

In cases 3 and 4, the unilateral supports are pushed upwards, and in case 4 the contact forces are dominating over the external loads applied, cf. Fig. 4.

[Place of Fig. 4]

Concerning the alteration of the material layout as the unilateral supports are moved closer to the elastic body, we can draw conclusions similar to those in References 42, 16. If the supports are remote, one obtains classical minimum compliance optima. Moving the supports closer to the structure, the optimum layout changes quickly, however continuously, and the displacements are considerably reduced at the expense of non-zero contact pressures. However, these contact pressures are favourable compared to those of a structure with uniform design. (The computation of a piece-wise linear approximation of the distribution of the contact pressure $p(x)$, through the use of the quantities u^*, h^* and λ^*, is described in Appendix B.)

Since Fig. 3 is quite representative for the examples studied, not all the cases are illustrated. For the resulting contact pressure, we observed an improvement in all cases; Fig. 5 illustrates the results for cases 2 and 4.

[Place of Fig. 5]

The fifth case is included mainly to show that an optimal design might very well be semi-coercive. In this case, the optimized structure is not attached to the "ceiling", but is supported by the unilateral obstacles only, cf. Fig. 6.

[Place of Fig. 6]

Since case 5 is much less constrained, the gain in structural performance is much larger, as is seen in Fig. 6 (b) and Fig. 7. Of course, the scale for the displacement in Fig. 7 is different from the one in Fig. 3.

[Place of Fig. 7]

6 CONCLUSIONS

We have presented a solution approach to finite element discretized optimum sheet problems in unilateral contact. General upper and lower design constraints can be included, and convergence is guaranteed even when allowing vanishing designs. Upper and lower bounds on the saddle value determine the quality of solutions at termination.

The proposed algorithm was found to be satisfactory in solving large-scale problems. Its computational efficiency is contributed to by the fact that the algorithm, although working essentially only in the space of displacement variables, also generates an optimal design through a simple averaging scheme. The practical efficiency is also supported by its low storage requirements and computational complexity per iteration, which, in turn, partly stems from the fact that the subgradient optimization scheme utilizes the problem structure very efficiently. (For example, the Euclidean projection can be performed very easily thanks to the quasi-orthogonality of the constraint matrix C.)

The overall time complexity was found to be nearly quadratic in the number of design variables. Compared to the time needed to solve the equilibrium problem for a uniform thickness sheet, it is roughly double. This should be compared to the ratio of up to at least 100 which is often encountered in optimality criterion algorithms[8]. Our interpretation of these results is that our approach seems to be a practical and efficient tool for the solution of the topology optimization problem at hand, although not the best equilibrium solver.

For the computed solutions it is observed that, in comparison to a uniform thickness sheet, the optimum designs generate significantly improved distributions of displacements, stresses and contact pressures.

Appendix

A RAPIDLY OSCILLATING DESIGN FIELDS

The results of Reference 6 guarantee a rather weak convergence of the designs as we refine our mesh [convergence in the weak* topology of $L^\infty(\Omega)$]. We will here try to clarify what consequences this has on our problem. To demonstrate the connection between weak* convergence and *checkerboard patterns* for the approximating design field, and to validate a mean-value strategy for adjacent elements, we study a one dimensional example below.

Consider a sequence of functions $\{f_\ell\}_{\ell=1}^\infty$ in $L^\infty(0,1)$, where

$$f_\ell(x) = \begin{cases} 1, & \text{if } x \in \bigcup_{k=0}^{2^{\ell-1}-1} \left(\frac{2k}{2^\ell}, \frac{2k+1}{2^\ell}\right), \\ 0, & \text{otherwise}, \end{cases}$$

i.e., f_ℓ is a step function which takes the value one in half of the 2^ℓ intervals that $(0,1)$ is uniformly divided into. This represents a one dimensional checkerboard pattern as the mesh size $2^{-\ell}$ is decreased. It is straightforward to show that, as $\ell \to \infty$,

$$f_\ell \xrightarrow{w^*} f \quad \Longleftrightarrow \quad \int_0^1 f_\ell \varphi \, dx \to \int_0^1 f \varphi \, dx, \qquad \forall \varphi \in L^1(0,1),$$

where $f(x) \equiv \frac{1}{2}$ corresponds to the optimal design. Hence, checkerboards are not excluded by the weak* convergence. However, in the above example the desired limit is the mean value of the two extreme values occuring in $\{f_\ell\}$. The (weak*) convergence is here drastically "improved" by creating a new sequence $\{\overline{f}_\ell\}_{\ell=1}^\infty$ such that, for each ℓ, $\overline{f}_\ell$ is the average of the two values of f_ℓ in any two adjacent segments in their union. In this example we obtain $\overline{f}_\ell = f$ for all ℓ. It is also straightforward to show that for any sequence of step functions (with the above choice of domain decomposition) which converges weakly* to a function $f \in L^\infty(0,1)$, this convergence property is retained for $\overline{f}_\ell$, i.e., $\overline{f}_\ell \overset{w^*}{\to} f$. This mean value strategy for adjacent elements is applied to the sheet problems of this paper.

Suppose, for simplicity, that an even number (m) of triangles of equal size are such that contiguous triangles Ω_i and Ω_j can be paired together to form a rectangle $R_i = \Omega_i \cup \Omega_j$. We treat the case where the lower and upper thickness bounds are constant and equal to α_L and α_U, respectively. Further, suppose that a thickness distribution $h_m \in L^\infty(\Omega)$ is given by $(S\Pi)$ and is such that $h_m \overset{w^*}{\to} \tilde{h} \in L^\infty(\Omega)$ as $m \to \infty$, where $\tilde{h}$ is part of an optimum pair to (SJ). For each rectangle R_i, h_m takes two different values: ϱ_m^i in Ω_i, and ϱ_m^j in Ω_j. We define $\overline{h}_m$ such that $\overline{h}_m|_{R_i} \equiv \frac{1}{2}(\varrho_m^i + \varrho_m^j)$ for all rectangles R_i. Clearly, under the above assumptions, $\overline{h}_m$ belongs to (the original definition of) $\mathcal{H}$ whenever h_m does. We next establish that $\overline{h}_m \overset{w^*}{\to} \tilde{h}$ as $m \to \infty$.

Let $\varphi \in C(\overline{\Omega})$ and $\varepsilon > 0$ be arbitrary. There is an integer $M(\varepsilon)$ such that

$$m \geq M(\varepsilon) \;\Rightarrow\; |\varphi(x) - \varphi(y)| < \frac{2\varepsilon}{|\Omega|\alpha_U}, \qquad \forall x, y \in R_i \text{ and any } i,$$

where $|\Omega|$ is the area of the domain Ω. This implies that for any rectangle $R_i = \Omega_i \cup \Omega_j \subset \Omega$,

$$\left| \int_{\Omega_i} \varphi \, d\Omega - \int_{\Omega_j} \varphi \, d\Omega \right| \leq \frac{2\varepsilon}{|\Omega|\alpha_U} |\Omega_i| = \frac{2\varepsilon}{|\Omega|\alpha_U} \frac{|\Omega|}{m} = \frac{2\varepsilon}{m\alpha_U}. \tag{39}$$

Further,

$$\left| \int_\Omega \varphi \overline{h}_m \, d\Omega - \int_\Omega \varphi h_m \, d\Omega \right| \leq \sum_{i=1}^{m/2} \left| \int_{R_i} \varphi(\overline{h}_m - h_m) \, d\Omega \right|, \tag{40}$$

and

$$\begin{aligned} \left| \int_{R_i} \varphi(\overline{h}_m - h_m) \, d\Omega \right| &= \left| \frac{\varrho_m^i + \varrho_m^j}{2} \int_{\Omega_i \cup \Omega_j} \varphi \, d\Omega - \varrho_m^i \int_{\Omega_i} \varphi \, d\Omega - \varrho_m^j \int_{\Omega_j} \varphi \, d\Omega \right| \\ &\leq \frac{\varrho_m^i + \varrho_m^j}{2} \left| \int_{\Omega_i} \varphi \, d\Omega - \int_{\Omega_j} \varphi \, d\Omega \right|. \end{aligned} \tag{41}$$

Now, using (39) in (41), and the resulting inequality in (40), we arrive at

$$\left| \int_\Omega \varphi \overline{h}_m \, d\Omega - \int_\Omega \varphi h_m \, d\Omega \right| \leq \varepsilon, \qquad \forall m \geq M(\varepsilon). \tag{42}$$

By (42) it follows that, for any $\varphi \in C(\overline{\Omega})$

$$\int_\Omega \varphi(\overline{h}_m - \tilde{h}) \, d\Omega \to 0 \quad \text{as } m \to \infty. \tag{43}$$

If we can establish (43) with $\varphi \in C(\overline{\Omega})$ replaced by any $\overline{\varphi} \in L^1(\Omega)$, then we are done. This is accomplished through the density of $C(\overline{\Omega})$ in $L^1(\Omega)$. Let $\overline{\varphi} \in L^1(\Omega)$ and $\varepsilon > 0$ be arbitrary. There exists $\varphi \in C(\overline{\Omega})$ such that

$$\int_\Omega |\varphi - \overline{\varphi}| \, d\Omega < \frac{\varepsilon}{2\alpha_U}. \tag{44}$$

Further, by (44) and the Hölder and triangle inequalities,

$$\begin{aligned} \left| \int_\Omega \overline{\varphi}(\overline{h}_m - \tilde{h}) \, d\Omega \right| &= \left| \int_\Omega \overline{h}_m(\overline{\varphi} - \varphi) \, d\Omega + \int_\Omega (\overline{h}_m - \tilde{h})\varphi \, d\Omega + \int_\Omega (\varphi - \overline{\varphi})\tilde{h} \, d\Omega \right| \\ &\leq \alpha_U \frac{\varepsilon}{2\alpha_U} + \left| \int_\Omega (\overline{h}_m - \tilde{h})\varphi \, d\Omega \right| + \alpha_U \frac{\varepsilon}{2\alpha_U}. \end{aligned} \tag{45}$$

By (43) and (45),

$$0 \leq \left| \int_\Omega \overline{\varphi}(\overline{h}_m - \tilde{h}) \, d\Omega \right| \leq \varepsilon + \left| \int_\Omega (\overline{h}_m - \tilde{h})\varphi \, d\Omega \right| \to \varepsilon \quad \text{as } m \to \infty,$$

which gives the desired result.

We remark that the analogue of the above result can be shown for the checkerboard control proposed in Reference 38, where four-noded quadrilateral elements are grouped into patches (consisting of four quadrilaterals instead of two triangles). Whenever the original design sequence converges weakly* then so does the new design sequence and their limits are equal.

In almost all of our numerical experiments noticeable checkerboard patterns were obtained when the mean-value strategy was not used, the exception being the non-contact case 1 in Table I; in all other cases, we have incorporated the above mean-value strategy. We note that no rapid design changes apart from the expected ones were observed when using this strategy.

A finite element convergence theory which includes a strong convergence of designs (in some norm) is highly desirable, since then checkerboards could be proven not to occur. Main differences with the LBB theory[39], which is such a theory for, e.g., Stokes flows, are that the functional is trilinear (rather than linear-quadratic) and that the design (which here corresponds to the pressure) belongs to the non-reflexive space $L^\infty(\Omega)$. Jog and Haber[40] have proposed to use a Patch test to determine which elements will be stable. Although no convergence proof is established, the test is reported to agree well with numerical experiments.

We conclude that if a strong convergence result is possible to achieve, then it is certainly not in the norm of $L^\infty(\Omega)$. It is the authors belief that, in addition to a proper norm selection, there is a need for a characterization and clearer understanding of the possible non-uniqueness of the optimal designs.

B CALCULATION OF CONTACT PRESSURE

It is known[41] that if the work of the contact forces is approximated by

$$\int_{S_c} pw\, dS \approx \sum_{i=1}^{G} W_i(pw)(x_i), \tag{46}$$

where x_i are the integration points and W_i weighting factors, then the relation between contact force magnitudes λ_i^* and contact pressures $p_i = p(x_i)$ is

$$\lambda_i^* = W_i p_i. \tag{47}$$

In our case we calculate

$$\int_{S_c} pw\, dS = \int_{x_1}^{x_r} (pw)(x)h(x)\, dx.$$

We take the nodes to be the integration points and x_i the x-value of the ith contact node in Γ_c. We approximate h as an elementwise constant function, to obtain

$$\int_{S_c} pw\, dS \approx \sum_{i=1}^{r-1} h_i \int_{x_i}^{x_{i+1}} (pw)(x)\, dx, \tag{48}$$

where $h_i = h|_{(x_i,x_{i+1})}$. Applying the trapezoidal rule on the integral in (48),

$$\int_{S_c} pw\, dS \approx \sum_{i=1}^{r-1} \frac{h_i L_i}{2}(p_i w_i + p_{i+1} w_{i+1}), \tag{49}$$

where $L_i = x_{i+1} - x_i$, $p_i = p(x_i)$ and $w_i = w(x_i)$. With the choices $G = r$ and

$$W_i = \frac{1}{2}(h_{i-1}L_{i-1} + h_i L_i), \qquad i = 2, \ldots, r-1, \tag{50}$$

$$W_1 = \frac{1}{2}h_1 L_1, \qquad W_r = \frac{1}{2}h_{r-1}L_{r-1}, \tag{51}$$

the relation (49) implies (46). Hence, the relation (47), with the weighting factors given by (50) and (51), is used to obtain the results in Fig. 5 and Fig. 6 (b).

References

1. W. Prager and J. E. Taylor, 'Problems of optimal structural design', *Trans. ASME J. Appl. Mech.* **35**, 102–106 (1968).
2. M. P. Rossow and J. E. Taylor, 'A finite element method for the optimal design of variable thickness sheets', *AIAA J.* **11**, 1566–1568 (1973).
3. R. L. Benedict, 'Maximum stiffness design for elastic bodies in contact', *J. Mech. Design* **104**, 825–830 (1982).
4. J. Céa and K. Malanowski, 'An example of a max-min problem in partial differential equations', *SIAM J. Control* **8**, 305–316 (1970).

5. J. Petersson, 'On stiffness maximisation of variable thickness sheet with unilateral contact', *Quart. Appl. Math.* (to appear).
6. J. Petersson and J. Haslinger, 'An approximation theory for optimum sheets in unilateral contact', (submitted).
7. G. I. N. Rozvany, *Structural Design via Optimality Criteria*, Kluwer Academic Publishers, Dordrecht, 1989.
8. M. P. Bendsøe and J. M. Guedes, 'Some computational aspects of using extremal material properties in the optimal design of shape, topology and material', *Control Cybernet.* **3**, 327–350 (1994).
9. A. Ben-Tal and M. P. Bendsøe, 'A new method for optimal truss topology design', *SIAM J. Optim.* **3**, 322–358 (1993).
10. U. T. Ringertz, 'On finding the optimal distribution of material properties', *Struct. Optim.* **5**, 265–267 (1993).
11. R. T. Rockafellar, 'Augmented Lagrangians and applications of the proximal point algorithm in convex programming', *Math. Oper. Res.* **1**, 97–116 (1976).
12. J. M. Danskin, *The Theory of Max-Min*, Springer-Verlag, Berlin, 1967.
13. V. F. Dem'yanov and V. N. Malozemov, *Introduction to Minimax*, John Wiley & Sons, New York, 1974.
14. J.-B. Hiriart-Urruty and C. Lemaréchal, *Convex Analysis and Minimization Algorithms I: Fundamentals*, Springer-Verlag, Berlin, 1993.
15. J. Petersson, 'Stiffness optimization of general structure in Signorini-type contact', *Proc. 2nd Contact Mech. Int. Symp.* Marseille, France, 1994 (to appear).
16. J. Petersson and A. Klarbring, 'Saddle point approach to stiffness optimization of discrete structures including unilateral contact', *Control Cybernet.* **3**, 461–480 (1994).
17. W. Achtziger, M. P. Bendsøe, A. Ben-Tal and J. Zowe, 'Equivalent displacement based formulations for maximum strength truss topology design', *IMPACT Comput. Sc. Engrg.* **4**, 315–345 (1992).
18. B. N. Pshenichny and Yu. M. Danilin, *Numerical Methods in Extremal Problems*, MIR Publishers, Moscow, 1978.
19. B. T. Poljak, 'A general method of solving extremum problems', *Soviet Math. Dokl.* **8**, 593–597 (1967).
20. B. T. Polyak, 'Minimization of unsmooth functionals', *U.S.S.R. Comput. Math. and Math. Phys.* **9**, 14–29 (1969).
21. N. Z. Shor, *Minimization Methods for Non-Differentiable Functions*, Springer-Verlag, Berlin, 1985.
22. M. A. Shepilov, 'Method of the generalized gradient for finding the absolute minimum of a convex function', *Cybernet.* **12**, 547–553 (1976).
23. C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*, Prentice-Hall, Englewood Cliffs, 1982.
24. A. V. Aho, J. E. Hopcroft and J. D. Ullman, *Data Structures and Algorithms*, Addison-Wesley, Reading, 1985.
25. Y. M. Dirickx and L. P. Jennergren, *System Analysis by Multilevel Methods*, John Wiley & Sons, Chichester, 1979.
26. G. B. Dantzig and P. Wolfe, 'Decomposition principle for linear programs', *Oper. Res.* **8**, 101–111 (1960).
27. T. Larsson and Z.-W. Liu, 'A Lagrangian relaxation scheme for structured linear programs with application to multicommodity network flows', Report LiTH-MAT-R-89-24, Department of Mathematics, Linköping Institute of Technology, Linköping, Sweden, 1989. Revised 1992.
28. T. Larsson, Z.-W. Liu and M. Patriksson, 'A dual scheme for traffic assignment problems', Report LiTH-MAT-R-92-21, Department of Mathematics, Linköping Institute of Technology, Linköping, Sweden, 1992.
29. M. S. Bazaraa, H. D. Sherali and C. M. Shetty, *Nonlinear Programming: Theory and Algorithms*, 2nd edn, John Wiley & Sons, New York, 1993.
30. J. M. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several*

Variables, Academic Press, New York, 1970.
31. T. Larsson, M. Patriksson and A.-B. Strömberg, 'Ergodic results in subgradient optimization', Report LiTH-MAT-R-95-21, Department of Mathematics, Linköping Institute of Technology, Linköping, Sweden, 1995.
32. R. T. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, New Jersey, 1970.
33. T. J. R. Hughes, *The Finite Element Method - Linear Static and Dynamic Finite Element Analysis*, Prentice-Hall, Englewood Cliffs, 1987.
34. A. Ben-Tal and A. Nemirovskii, 'An interior point algorithm for truss topology design', in M. P. Bendsøe and C. A. Mota Soares (eds), *Topology design of Structures*, Kluwer Academic Publishers, Dordrecht, 1993, pp. 55–70.
35. A. Ben-Tal and A. Nemirovskii, 'Potential reduction polynomial time method for truss topology design', *SIAM J. Optim.* **4**, 596–612 (1994).
36. E. S. Levitin and B. T. Polyak, 'Constrained minimization methods', *U.S.S.R. Comput. Math. and Math. Phys.* **6**, 1–50 (1966).
37. M. Patriksson and J. Petersson, 'A subgradient method for contact structural optimization', *Proc. Int. Conf. Compl. Prob.* Baltimore, USA, 1995 (to appear).
38. M. P. Bendsøe, A. Diaz and N. Kikuchi, 'Topology and generalized layout optimization of elastic structures', in M. P. Bendsøe and C. A. Mota Soares (eds), *Topology design of Structures*, Kluwer Academic Publishers, Dordrecht, 1993, pp. 159–205.
39. F. Brezzi and M. Fortin, *Mixed and Hybrid Finite Element Methods*, Springer-Verlag, New York, 1991.
40. C. S. Jog and R. B. Haber, 'Checkerboard and other spurious modes in solutions to distributed-parameter and topology design problems', *Proc. 1st World Congr. Struct. Multidisc. Optim.* Goslar, Germany, 1995 (to appear).
41. A. Klarbring and G. Björkman, 'A mathematical programming approach to three-dimensional contact problems with friction', *Comput. Methods Appl. Mech. Engrg.* **58**, 175–200 (1986).
42. A. Klarbring and M. Rönnqvist, 'Nested approach to structural optimization in nonsmooth mechanics', *Struct. Optim.* (to appear).