On the Solution of a Minimum Weight Elastoplastic Problem Involving Displacement and Complementarity Constraints

M.C. Ferris[*] F. Tin-Loi[†]

12 July 1997

Abstract

This paper deals with a special class of minimum weight design problems involving discretized structures, holonomic (reversible) plasticity, and constraints on displacements. A key feature of the optimization problem is the presence of complementarity conditions, involving the orthogonality of two sign-constrained vectors. This problem falls within the important class of so-called Mathematical Programs with Equilibrium Constraints (MPECs). Two simple algorithms are proposed to solve our synthesis problem and application is illustrated by some examples concerning truss-like structures.

1 Introduction

After more than three decades of research, the area of structural optimization has become in its own right a well-established and fruitful discipline within structural engineering. Formal methods, typically based on either a mathematical programming approach or some optimality criteria method, have been developed to tackle both fixed geometry layouts as well as the more topical topology optimization problem. A vast number of papers including comprehensive surveys of various generic problem classes, as found for example in such recent publications as [1]–[7], attest to the vigor with which research has been and is still being carried out in the area.

---

[*] Computer Sciences Department, University of Wisconsin – Madison, 1210 West Dayton Street, Madison, WI 53706, USA (ferris@cs.wisc.edu).

[†] School of Civil and Environmental Engineering, University of New South Wales, Sydney 2052, Australia (f.tinloi@unsw.edu.au).

In this paper, we revisit a problem considered briefly in the early 80s by Maier and co-workers [8]–[10]. It concerns an important class of structural optimization problems, commonly referred to by mathematical programmers as Mathematical Programs with Equilibrium Constraints (MPECs for short) [11] in which it is required to minimize the weight (volume) of discretized elastoplastic structures subject to displacement limitations and some technological constraints. The feature (and difficulty) of this problem lies in the presence of nonlinear, nonconvex complementarity constraints, characterized by the orthogonality of two sign-constrained vectors.

The challenge posed by our optimization problem in point lies in the development of suitable (efficient and robust) numerical algorithms rather than on its formulation which is straightforward. The iterative branch-and-bound based numerical procedure proposed in [8] is reported [9] to have failed with some of the problems tried, even though they were of very small sizes. Cinquini and Contro [10] developed optimality criteria through the Lagrange multiplier technique and claim that these criteria may be useful in the solution of practical size problems. Unfortunately, they only illustrate their approach by means of a three-bar truss example. Some related recent work by Kaliszky and Lógó [12], recognizing the computational difficulty inherent in a direct approach, proposed an alternative, albeit approximate, scheme of controlling deformations by introducing a constraint on the complementary strain energy. This avoids the need for a complementarity constraint and, we must admit, is useful for repeated loading processes in that only bounds, rather than actual values, on deflections are possible in many situations. However, a major shortcoming is that there is no direct relationship between deflections and complementary strain energy. Use of the energy bound approach has also been investigated by Muralidhar and Rao [13].

We propose in this computation-oriented paper two simple mathematical programming based approaches with the potential of solving efficiently the minimum weight design problem, even for large-size, practical structures. A primary motivation for the proposed, almost heuristic, algorithms is to exploit the availability of sophisticated general purpose nonlinear programming solvers such as MINOS [14] and CONOPT [15], especially through the use of the modeling language GAMS [16]. We thus aim to facilitate the solution of such problems by explicitly demonstrating how to easily and concisely formulate them using a modeling language and to transparently call the required mathematical programming solver.

The organization of this paper is as follows. In the next section, we review the analysis problem for a class of discretized models exhibiting piecewise linearized holonomic (reversible) constitutive laws. In essence, the displacement response for a given load is sought for a structure of known geometric and material properties. For the sake of simplicity, as in [8], we present our formulations and discussions with reference to truss-like structures which in effect possess all essential features of practical and more complex bar structures. As is well-known (see references in [17], [18]), the governing relations can be cast in fundamentally three forms by suitable elimination of variables. All three formulations, however, involve complementarity as a key mathematical structure. The minimum weight design problem is considered in the ensuing Section 3. Briefly, we now aim to minimize the weight of the topologically known elastoplastic discretized structure under displacement and technological constraints for a given load level. We assume that stiffnesses, yield limits and hardening parameters can all be expressed as some functions of the design variable, cross-sectional areas for instance. It is shown that this synthesis problem can be formulated as an MPEC. In Section 4, we outline two algorithms for solution of our particular optimization problem. We also illustrate the use of GAMS by modeling and solving a simple example using one of the algorithms proposed. Finally, three numerical examples are solved in Section 5; two of these have been considered in [9], [10] while the third concerns a more realistic structure such as would be encountered in practice.

A word regarding notation is in order. Column vectors are assumed; a real vector x of size m is indicated by $x \in \Re^n$ and a real $m \times n$ matrix A by $A \in \Re^{m \times n}$. Transpose is indicated by the superscript T and the inverse of a matrix by the superscript -1. The complementarity relationship between nonnegative vectors w and z is written as $w^T z = 0$ which implies the componentwise condition $w_i z_i = 0$ for all i.

2 The Analysis Problem

Consider a suitably space-discretized skeletal (trusses, frames and grids) structural system for which the constitutive behavior is directly reflected by the element behavior [19]. The simplest model, and the one we consider herein in our examples, is that of a truss. Under a holonomy assumption, in the spirit of the deformation theory of plasticity, we formulate the single step analysis problem simply by collecting and manipulating, if necessary, the relations describing the three key ingredients of the structural behavior: statics, kinematics and constitutive laws. We further assume a small deformation theory and adopt, without undue loss of generality, suitably piecewise linearized yield surfaces.

In terms of well-known notation and description [20], the governing relations for the whole structure can be written as

$$\begin{aligned}
F &= C^T Q, && (1)\\
q &= Cu, && (2)\\
q &= e + p, && (3)\\
Q &= Se, && (4)\\
p &= Nz, && (5)\\
w &= -N^T Q + Hz + r, && (6)\\
w &\geq 0, \quad z \geq 0, \quad w^T z = 0. && (7)
\end{aligned}$$

As is typical in finite element methodology, vector and matrix quantities represent the unassembled contributions of corresponding elemental entities, as concatenated vectors and block diagonal matrices, respectively. For example, with n finite elements and denoting the corresponding element index by a subscript, we have $Q^T = (Q_1^T \ldots Q_n^T)$, $S = diag(S_1, \ldots, S_n)$, etc.

For a structure with d degrees of freedom, m member generalized quantities and y yield functions, equilibrium between the nodal loads $F \in \Re^d$ and the natural generalized stresses $Q \in \Re^m$ is expressed by (1) through the geometric or compatibility matrix $C \in \Re^{m \times d}$ of full column rank. Equation (2) represents linear compatibility of strains $q \in \Re^m$ with the nodal displacements $u \in \Re^m$. Relations (3)–(7) embody the holonomic constitutive laws: additivity of elastic $e \in \Re^d$ and plastic $p \in \Re^d$ strains in (3); linear elasticity in (4), where $S \in \Re^{m \times m}$ is an elastic matrix of unassembled symmetric positive definite element stiffnesses; plastic strains p in (5) defined by an associated flow rule and expressed as functions of the plastic multipliers $z \in \Re^y$ through the constant matrix of outward normals $N \in \Re^{m \times y}$ to the yield surface; a piecewise linear yield function $w(Q(z), z) : \Re^y \to \Re^y$ in (6) which accommodates, through $H \in \Re^{y \times y}$, a class of hardening models with known yield limits $r \in \Re^y$; and finally, a complementarity relationship in (7) between the sign-constrained total quantities w and z. Condition (7) simply implies that when yielding occurs or $w = 0$, plastic flow can happen or $z \geq 0$, whereas an elastic condition or $w > 0$ precludes the formation of plastic strains or $w = 0$.

Relation set (1)–(7) can be simplified in three possible ways to give corresponding formulations in (Q, u, z), (u, z) or z variables. We will not detail the simple algebraic manipulations involved but will simply present these three formulations in the following. We adopt a tableau format for highlighting the key mathematical structures belonging to these alternative forms.

The (Q, u, z) formulation is

$$\begin{bmatrix} -S^{-1} & C & -N \\ C^T & 0 & 0 \\ -N^T & 0 & H \end{bmatrix} \begin{bmatrix} Q \\ u \\ z \end{bmatrix} + \begin{bmatrix} 0 \\ -F \\ r \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ w \end{bmatrix}, \qquad (8)$$

$$w \geq 0, \quad z \geq 0, \quad w^T z = 0.$$

Variables Q can now be eliminated to furnish the (u, z) formulation

$$\begin{bmatrix} K_{uu} & K_{uz} \\ K_{zu} & K_{zz} \end{bmatrix} \begin{bmatrix} u \\ z \end{bmatrix} + \begin{bmatrix} -F \\ r \end{bmatrix} = \begin{bmatrix} 0 \\ w \end{bmatrix}, \qquad (9)$$

$$w \geq 0, \quad z \geq 0, \quad w^T z = 0$$

where $K_{uu} \equiv C^T S C$, $K_{uz} \equiv K_{zu}^T \equiv -C^T S N$, and $K_{zz} \equiv H + N^T S N$.

The most widely used approach [20] is to retain only z variables in the complementarity problem. This leads to the following standard Linear Complementarity Problem (LCP) which, for hardening matrices H, can be solved, for instance, by Lemke's classical algorithm [21]:

$$\begin{bmatrix} M \end{bmatrix} \begin{bmatrix} z \end{bmatrix} + \begin{bmatrix} r - N^T Q^e \end{bmatrix} = \begin{bmatrix} w \end{bmatrix}, \qquad (10)$$

$$w \geq 0, \quad z \geq 0, \quad w^T z = 0$$

where both $M \equiv H - N^T(SCK_{uu}^{-1}C^T S - S)N$ and $Q^e \equiv SCK_{uu}^{-1}F$ (the elastic response to F) are known from the given data.

At variance with Kaneko and Maier [8], we will formulate our optimization problem, in spite of the apparent compactness of LCP (10), using the (Q, u, z) set of relations. There are two main reasons for this.

First, we intend to use the high level mathematical programming language GAMS to model the minimum weight problem and hence have immediate and automatic access to sophisticated inbuilt tools for efficiently manipulating relations while maintaining sparsity. Automatic differentiation capabilities are additionally provided.

Second, and perhaps more importantly, the (Q, u, z) approach consists of small-size uncoupled complementarity problems expressed at the element level, instead of the single coupled LCP (10). This is particularly important in optimal design problems where, for instance, the various quantities making up matrix M are dependent on certain design variables. In particular, note that we ideally need an explicit inverse of the conventional structure stiffness matrix K_{uu} to form M and Q^e. Such an inverse has yet to be found although Fuchs [22] recently derived an analytic form which unfortunately, as he himself admits, appears to defy any practical application except for the simplest of problems. On the other hand, a (Q, u, z) approach involves, if required, the inverse of S at the element level and is a tractable task. Moreover, as will be seen later, our minimum weight design formulation avoids the necessity of explicitly computing an inverse.

Remark 1 For our adopted class of hardening laws, the theory behind the above three symmetric fully equivalent complementarity formulations is well-known [21]. Problem (8), for instance, is a saddle-point problem (convex in u, z and concave in Q) which can also be formulated as a pair of dual convex quadratic programming problems with corresponding extremum meanings, namely minimum total potential energy or minimum complementary energy. Further, for given loads, the positive semidefiniteness of H (and hence of M in (10)), as would be expected of a mix of perfectly-plastic and hardening element behavior, implies nonuniqueness of z and consequently of deflections u . However, uniqueness of stresses Q is guaranteed. Of course, positive definiteness of H, due to strict hardening behavior of all yield modes, would ensure uniqueness of u.

Remark 2 The primary reason for adopting an approximate holonomy, instead of the exact nonholonomy, hypothesis lies in the fact that holonomic (reversible, path-independent) laws have the appeal of simplicity and, in practice for realistic structures, are often accurate enough when loads monotonically increase, even when some manifestations of nonholonomic plasticity (namely local unloading) are likely to occur. A holonomic assumption is almost mandatory in our minimum weight problem since, as indicated in [8], full account of the irreversibility of the constitutive laws would lead to a complex nonlinear optimal control problem.

Remark 3 For "well-proportioned" realistic structures, violations of nonholonomy, as indicated in Remark 2, are expected not to be significant. However, when member sizes differ by large amounts, as would be the case for topology optimization where member cross-sections are allowed to vary freely, holonomic (reversible) unloading, especially if they start at an early stage of the loading process and occur extensively, may lead to significant reversals of deflections. It may then be possible that some deflection constraint satisfied for the given load level may be violated at a lower load level. Further, pathological phenomena such as the limited mechanism motion (pseudomechanism) described in [23] may become significant. However, such occurrences may still be checked *a posteriori* after an optimal design has been achieved.

3 The Minimum Weight Problem

The synthesis problem can be described briefly as follows. A skeletal elastoplastic structure of fixed topology has to be designed to resist certain specified loads and to have its corresponding displacements (and plastic deformations, if required) kept within certain assigned serviceability limits. The yield limits r, stiffnesses S and hardening parameters H of the constituent members of the structure are all regarded as unknown but assumed to be (continuous) functions of the cross-sectional areas of all n elements. The element areas, collected in vector $a \in \Re^n$, need to be chosen so as to minimize the total weight (or equivalently volume) of the structure while perhaps also satisfying certain so-called "technological" constraints (e.g. same areas for all members, minimum area requirements, etc.).

We assume a holonomy hypothesis since the loads can be conceived to be increased proportionally to their design levels. In fact, for known a, we can analyze the structure precisely using any of the three complementarity formulations (8), (9), or (10).

Let us collect all element lengths in vector $\ell \in \Re^n$. The minimum volume problem can then be formally stated as the following constrained optimization problem:

$$
\begin{array}{ll}
\text{minimize} & \ell^T a \\
\text{subject to} & -Q + SCu - SNu = 0, \\
 & C^T Q - F = 0, \\
 & -N^T Q + Hz + r = w, \\
 & w \geq 0, \quad z \geq 0, \quad w^T z = 0, \\
 & -\hat{u} \leq u \leq \hat{u}, \\
 & z \leq \hat{z}, \\
 & a_\ell \leq a \leq a_u, \\
\text{and} & Ta = 0
\end{array}
\tag{11}
$$

where $\hat{u} \in \Re^d$ is a vector of nonnegative deflections limits; $\hat{z} \in \Re^y$ is a vector of upper bounds on plastic multipliers used to model the limited ductility of the members; $a_\ell \in \Re^n$ and $a_u \in \Re^n$ are, respectively, lower and upper bounds on the cross-sectional areas; and $T \in \Re^{t\times n}$ is a technological matrix imposing t constraints on the design variables (areas).

Optimization problem (11), in variables (a, Q, u, z) is a special case of an MPEC [11] in which the equilibrium system takes the form of a complementarity condition. Clearly, the first four sets of constraints directly represent the static, kinematic and constitutive requirements as given in (8). The remaining constraints mathematically describe the additional design requirements (namely limit on deflections, ductility requirements and technological constraints), all or some of which are imposed depending on the particular problem being investigated.

Remark 4 Whilst an extensive theory of first and second order optimality conditions for MPECs has been developed in [11], still relatively little is known about the numerical solution of practical, large-scale MPECs likely to arise in realistic applications. The most prominent feature of an MPEC, and one that distinguishes it from a standard nonlinear program, is the presence of complementarity constraints. These constraints classify this class of mathematical programs as a nonlinear disjunctive (or piecewise) program and therefore carries with it a "combinatorial curse". This makes it very difficult to solve, especially if one wishes to compute a global optimal solution. A branch-and-bound technique, as basically used in [8], can be adopted to perform an exhaustive enumeration in the search for a global optimum, but is obviously severely limited in the size of problem it can handle. Nearly all methods proposed to date [11] are aimed at finding stationary solutions and/or local optima and are categorized roughly by the way the complementarity condition is handled.

Remark 5 Formulation (11) is fully equivalent to the one in (a, z) variables given by Kaneko and Maier [8]. Computationally, however, our form should be much easier to solve, especially via a modeling package. Note in particular, as indicated earlier, that we do not need to form K_{uu} (or worse its inverse) whose elements are complex functions of a; in our case, S consists of block-diagonal element matrices (each of size 1×1 for a truss, 3×3 for a plane frame, etc.) with terms which are immediately available (albeit not necessarily linear in a).

Remark 6 As indicated in [9], under certain conditions (primarily the positive definiteness of M, feasibility of the constraints, and all functions of a are continuous), the minimum weight problem has an optimal solution which needs not, however, be unique. Theoretical results for positive semidefiniteness of M were not given.

4 Solution Algorithms and Modeling

We propose, in the following, two simple and intuitive reformulations of (11) involving the use of standard, readily available nonlinear programming solvers. A primary motivation behind these schemes is to exploit the availability of sophisticated solvers such as MINOS [14] and CONOPT[15], especially from within the powerful GAMS modeling language [16] which will facilitate the modeling task. The use of GAMS on a simple truss example is also explicitly illustrated.

The attempt to formulate and solve an MPEC as a nonlinear program, it must be noted, is carried out in spite of the fact that traditional constraint qualifications are never satisfied, with the implication that the usual numerical methods for solving nonlinear programming problems may be expected to have some difficulties in their solution. Whilst there is no guarantee that the solution provided represents a local minimum to the MPEC (let alone a global minimum), we wish to investigate numerically if our simple schemes can provide reasonable solutions in practice.

4.1 Algorithm 1: penalty approach

The basic idea underlying the penalty method for solving MPEC (11) consists in choosing a "penalty parameter" ρ, and converting the MPEC to the following problem:

$$\begin{array}{ll} \text{minimize} & \ell^T a + \rho w^T z \\ \text{subject to} & -Q + SCu - SNu = 0, \\ & C^T Q - F = 0, \\ & -N^T Q + Hz + r = w, \\ & w \geq 0, \quad z \geq 0, \\ & -\hat{u} \leq u \leq \hat{u}, \\ & z \leq \hat{z}, \\ & a_\ell \leq a \leq a_u, \\ \text{and} & Ta = 0 \end{array} \tag{12}$$

The algorithm is simple. This penalized problem is solved for successively higher values of ρ to force the complementarity term, which is nonnegative at feasible points of (12), to approach zero. The attraction of this method is that each penalty subproblem is a standard nonlinear program and general purpose codes such as CONOPT [15] can be used. The following pseudocode further clarifies the algorithm:

```
Set: initial $\rho$ (e.g. $10^{-4}$), maximum number of iterations ($maxiter$), and $w^T z = 100$.
for $i = 1$ to $maxiter$
    if $w^T z \leq 10^{-8}$ exit
    solve nonlinear program (12)
    $\rho = 10\rho$
end
```

Remark 7 A difficulty with the penalty approach is that solving subproblem (12) for "large" values of ρ may cause numerical problems such as round-off and numerical instability. In this regard, the exact penalization results recently derived in [24] may be useful in helping to circumvent such a difficulty.

4.2 Algorithm 2: parametric approach

Our second algorithm, and one that we have successfully used in solving a class of structural identification problems formulated as MPECs [25], essentially treats the complementarity term $w^T z$ as the only objective, with the volume $\ell^T a$ relegated to and parametrically constrained in the constraints set.

More explicitly, we reformulate MPEC (11) as the following parametric (in μ) constrained nonlinear programming problem:

$$
\begin{aligned}
\text{minimize} \quad & w^T z \\
\text{subject to} \quad & -Q + SCu - SNu = 0, \\
& C^T Q - F = 0, \\
& -N^T Q + Hz + r = w, \\
& w \geq 0, \quad z \geq 0, \\
& -\hat{u} \leq u \leq \hat{u}, \\
& z \leq \hat{z}, \\
& a_\ell \leq a \leq a_u, \\
& Ta = 0, \\
\text{and} \quad & \ell^T a \leq \mu.
\end{aligned}
\tag{13}
$$

In essence, we now try to force the complementarity term to zero in the objective function, while iteratively relaxing a specified volume bound via the parameter μ through a series of major iterations. The following pseudocode elucidates this procedure:

```
Set: initial $\mu$, maximum number of iterations ($maxiter$), and $w^T z = 100$.
for $i = 1$ to $maxiter$
    if $w^T z \leq 10^{-8}$ exit
    solve nonlinear program (13)
    $\mu = 1.2\mu$
end
```

The initial value of μ and its increase for each major iteration i is of course problem-dependent. Obviously, too small an increase in μ will lead to slow convergence but possibly a better optimum. Moreover, a simple refinement of the above algorithm can be carried out. Basically, after having found a solution, further iterations are carried out, this time involving small reductions in μ (e.g. $\mu = 99.9\%$ of current optimum weight) till complementarity is violated. We have successfully implemented this scheme in all our computational testing.

4.3 Modeling with GAMS

GAMS [16], an acronym for General Algebraic Modeling System, is a high-level modeling language specially designed to facilitate the construction, solution and maintenance of large and complex mathematical programming models. In many instances, the data manipulation requirements limit mathematical programming applications more than optimization requirements. The typical end-user is generally more concerned with model formulation, representation and solution than with the details of the mathematical programming techniques involved. The use of a modeling language makes the solution phase as simple as possible while at the same time allowing for easy construction of large and complex models.

We do not intend to elaborate too much on the well-known advantages provided by modeling languages, but wish to point out some specific ones in our particular optimization application. These include: simplicity and compactness of model construction (e.g. the same statements apply to very small as well as to very large models); ease of changing from one formulation to another (e.g. from the penalty approach to the parametric one, and even from design to analysis using the optimal solution found as data); possibility of trying out different solvers with a single statement change (especially important for the difficult class of MPECs); and ability to easily carry out sensitivity type analyses on the optimal design. GAMS, in addition, provides other important capabilities such as an internal sparse representation and automatic differentiation.

In order to illustrate explicitly how a GAMS model looks like, we will develop one for the simple two degree-of-freedom, three-bar truss shown in Fig. 1, under the specified loading. Our aim is to carry out its minimum volume design using Algorithm 1 (the penalty method). The following material parameters are adopted: Young's modulus of elasticity $E = 20000$ and yield stress $\sigma = 50$ (in tension and compression). Bar areas are constrained to be identical and all deflections limited to a magnitude of 4; there are no constraints on the plastic multipliers. We further assume that the hardening parameters in tension and compression $= 0.125Ea_i/\ell_i$, for all bars i. The various quantities (with the element number indicated by a subscript) required for setting up the model (12) are

$$F = \begin{bmatrix} 400 \\ 600 \end{bmatrix}, \quad C = \begin{bmatrix} 0.6 & 0.8 \\ 0 & 1 \\ -0.6 & 0.8 \end{bmatrix}, \quad N = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix},$$

$$S = diag\left(\frac{Ea_1}{\ell_1}, \frac{Ea_2}{\ell_2}, \frac{Ea_3}{\ell_3}\right), \quad r = (\sigma a_1, \sigma a_1, \sigma a_2, \sigma a_2, \sigma a_3, \sigma a_3),$$

$$H = diag\left(\frac{0.125Ea_1}{\ell_1}, \frac{0.125Ea_1}{\ell_1}, \frac{0.125Ea_2}{\ell_2}, \frac{0.125Ea_2}{\ell_2}, \frac{0.125Ea_3}{\ell_3}, \frac{0.125Ea_3}{\ell_3}\right).$$

The GAMS model for this example is given in the Appendix. It should be noted that matrix C for this simple case is specified explicitly element by element. For large models, it should either be generated within GAMS as in [26] or imported as data generated by some conventional finite element package. We do not need to describe the model further. Readers familiar with GAMS will immediately understand the "sets", "variables" and "equations" declarations, while those not familiar with GAMS will appreciate the concise yet descriptive style and should recognize the parallel to (12). CONOPT is the specified solver. On executing this GAMS model, a minimum volume $\ell^T a = 10167$ (with all cross-sectional areas $= 7.262$) will be obtained.

Remark 8 There is currently, within the mathematical programming community, quite an intense effort being directed towards the development and implementation of mathematical programming solvers in general, and of MPEC solvers in particular. To facilitate the development and testing process for algorithm developers and also their use by application experts, Dirkse and Ferris [27] have recently extended GAMS to allow MPECs to be formulated explicitly within the modeling language (not indirectly in the form of some nonlinear program as in our current work). The value of this extension is that they allow algorithm developers to directly obtain relevant function and derivative values provided by GAMS. The interface also enables researchers developing MPEC codes easy access to data from realistic applications implemented as GAMS models.

5 Examples

We present three examples in this section to illustrate application of both algorithms. Examples 1 [10] and 2 [8] are academic structures which have been used in the literature to illustrate various approaches for the solution of the optimum problem. Example 3 is a more realistic structure representing a double-layer space truss. In all cases, we imposed the stringent criterion $w^T z \leq 10^{-8}$ to indicate that complementarity has been satisfied. We report on some computational details for Examples 2 and 3 only; Example 1 is too small for any useful comparisons. All runs were carried out on a Pentium Pro 200. It should also be noted that we did not try to tune any of the runs by adjusting the various parameters involved.

5.1 Example 1

This example, originally considered by Cinquini and Contro [10] concerns the simple 3-bar truss shown in Fig. 2. For the indicated geometry and loading, it is required to find the optimal volume ω of the structure for the following parameters (cross-sectional areas a_i are the design variables): Young's modulus $E = 1$, yield limit (tension and compression) $\sigma = 0.0015E$, hardening matrix $H_i = diag\,(Ea_i/6\ell_i, Ea_i/6\ell_i)$, and the vertical deflection of the loaded node fixed at 1/400.

Algorithms 1 and 2 were used, via a GAMS model and MINOS (set for automatic scaling), to solve this problem; CONOPT had difficulties with this apparently poorly scaled problem. For Algorithm 1, we used an initial penalty parameter $\rho = 10$ and incremented it by a factor of 10 every major iteration. For Algorithm 2, we set the initial bound on the volume to $\mu = 1$ and increased it by a factor of 1.1 every major iteration; this was followed by the refinement scheme previously described in which μ was reduced to 99.99% of the current optimum volume for each major iteration. In both cases, we adopted the starting point: $a_i = 1$, $S_i = 1$, $r_i = (1, 1)$.

Algorithm 1 gave the optimum solution $\omega = 36.917$, $a = (35.621, 2.795, 1.250)$ and Algorithm 2 the solution $\omega = 36.884$, $a = (35.600, 2.793, 1.239)$. These can be compared with the results in [10], that is $\omega = 36.9$, $a = (35.4, 3.0, 1.3)$. As in that reference, two further cases were modeled using Algorithm 1, namely (a) assuming an indefinitely elastic behavior and (b) a purely elastic structure with the same constraints on stress levels as for the initial case. We modeled Case (a) by simply increasing the yield limit to a large (unattainable value), and Case (b) by imposing the constraint $z = 0$. For such cases, Algorithm 2 is not applicable since complementarity is always zero. In Case (a), we obtained $\omega = 29.223$, $a = (24.549, 5.097, 2.279)$ as compared with the reported solution $\omega = 29.2$, $a = (24.5, 5.1, 2.3)$. Case (b) results are $\omega = 37.079$, $a = (36.455, 2.272, 1.016)$ as compared with the reported solution $\omega = 37.1$, $a = (36.5, 2.3, 1.0)$.

5.2 Example 2

This second example was used by Kaneko and Maier [8]. It concerns a six-bar truss with the geometry shown in Fig. 3. Each bar was assumed to have a single perfectly-plastic (i.e. $H_i = 0$) yield mode in compression implying also an infinitely elastic behavior in tension. The optimization was formulated in terms of the compressive plastic resistances r_i of the members, and the objective function ω expressed as $\ell^T r$. With $r = (r_1, r_2, r_3, r_4, r_5, r_6)$, it was also assumed that the stiffnesses were numerically equal to the resistances so that $S = diag(r_1, r_2, r_3, r_4, r_5, r_6)$. Further, a limitation of 4 was imposed on all displacements. As in [8], three cases were considered as follows:

Case 1: $F = (9, 0, 0, 0)$; $r_1 = r_2 = r_3$, $r_4 = r_5 = r_6$.

Case 2: $F = (7, -3, 0, 0)$; $r_1 = r_2 = r_3$, $r_4 = r_5 = r_6$.

Case 3: $F = (9, 0, 0, 0)$; no constraints on r.

In addition to the above, we also imposed (although not explicitly stated in [8]) a small positive bound on $r \geq 10^{-6}$ to prevent members from being completely removed from the structure. We used GAMS/CONOPT for all runs with Algorithms 1 and 2. For Algorithm 1, the initial $\rho = 10^{-2}$ was increased by a factor of 10 every major iteration; the starting vector was $a_i = 1$, $S_i = 1$, $r_i = 1$ and (all components of) $w = 1$, $Q = 1$, $u = 1$. For Algorithm 2, the initial $\mu = 1$ was increased by a factor of 1.2 every major iteration of the initial stage followed by a 0.01% decrease in optimum ω for each major iteration of the refinement stage; the starting vector was $a_i = 0.1$, $S_i = 0.1$, $r_i = 0.1$ and (all components of) $w = 1$, $Q = 1$, $u = 1$.

Table 1 summarizes the results obtained and compares them with those in [8], wherever possible. We report only on essential details. For all three cases, the optimum values given by Algorithms 1 and 2 are better than those obtained by Kaneko and Maier [8]; the difference is especially noticeable for Case 3. Although this particular example is too small for definite comparisons, Algorithm 1 appears to outperform Algorithm 2 by a factor of about 3–5. Obviously, as previously noted, the execution times depend also to a large extent on the various parameters (e.g. ρ, μ, etc.) adopted.

5.3 Example 3

This example concerns the space truss shown in Fig. 4. This double-layer parallel grid is 16 m by 16 m in plan size and $2\sqrt{2}$ m high. It is restrained vertically at each top node along the perimeter and in all directions at the four corner supports to avoid rigid body motion. The truss consists of 128 members, 41 nodes and 99 degrees of freedom. It was loaded by nodal vertical loads applied to the top nodes to simulate a uniformly distributed loading of 0.1α T/m^2, that is by nodal point loads of 1.6α T at each of the 9 interior top nodes. Adopting cross-sectional areas a as design variables, we assumed the following for all cases considered (units are T and cm): $\alpha = 15$; $S_i = 5a_i$; $r_i = (2.5a_i, 1.25a_i)$; $H_i = diag(0.625a_i, h_i^-)$; and objective function $\omega = \Sigma a_i$ (since all bar lengths are equal). Further, all deflections were limited to a maximum absolute value of 10. Four cases were investigated:

Case 1: all a_i equal; $h_i^- = 0$.

Case 2: same top bars, same diagonal bars, same bottom bars; $h_i^- = 0$.
Case 3: $a_i \geq 10^{-3}$; $h_i^- = 0$.
Case 4: $a_i \geq 10^{-3}$; $h_i^- = 10^{-6}$.

As in Example 2, we used GAMS/CONOPT for all runs. For Algorithm 1, the initial $\rho = 1$ was increased by a factor of 10 every major iteration; the starting vector was $a_i = 1$, $S_i = 5$, $r_i = (2, 2)$, $H_i = diag(0.625, 0)$ and (all components of) $z = 0.01$, $w = 0.1$, $Q = 10$, $u = 10$. For Algorithm 2, μ, set initially at 1000 for Cases 1 and 2 and 600 for Cases 3 and 4, was increased by a factor of 1.2 every major iteration of the initial stage followed by a 1% decrease in optimum ω for each major iteration of the refinement stage; the starting vector was $a_i = 1$, $S_i = 5$, $r_i = (2, 2)$, $H_i = diag(0.625, 0)$ and (all components of) $z = 0.1$, $w = 10$, $Q = 10$, $u = 1$.

Table 2 summarizes the results obtained. Algorithm 1, for the parameters used, performed better than Algorithm 2, especially for Case 4 in which bar areas were allowed to vary freely (as in optimum topology design). It also appears that Algorithm 2 produced apparently better optimum designs for Cases 3 and 4 but it should be noted that the optimum values are actually on the bounds set on μ. The optima obtained in these cases seem to be in fact dependent on the termination criterion set for complementarity. We suspect therefore that the optimum values given by Algorithm 1 are more reliable.

Using the areas obtained by Algorithm 1, we have also carried out corresponding holonomic analyses, given by (8), using the powerful GAMS "restart" facility and the mixed complementarity solver PATH [28]. Corresponding curves of load factor α versus vertical central deflection of the top layer centre node (expected to be maximum at that location) are shown in Fig. 5. All designs clearly satisfy the target $\alpha = 15$, maximum deflection ≤ 10. However, surprisingly at first sight, the deflection responses for Cases 3 and 4 both show, for some load levels, a decrease in deflection for increasing loads. Further, for Case 4, the deflection limit of 10 is actually violated for smaller load levels than $\alpha = 15$. These behaviors are due to extensive unloading of the activated yield modes (from about $\alpha = 9.4$) causing a severe violation of nonholonomy. It should also be noted that the introduction of full hardening in Case 4 implies a unique deflection field, but still does not preclude the holonomic response from being a grossly inexact approximation of nonholonomic behavior. No unloading occurs for Case 1. For Case 2, a limited amount of unloading occurs from about $\alpha = 13.1$, albeit not enough to cause reversal of deflections.

6 Concluding Remarks

This paper is concerned with an important and difficult class of optimum design problems involving displacement constraints and complementarity conditions, the latter embodying a key behavior of holonomic plasticity. The minimum weight design problem, illustrated without undue loss of generality by truss-like structures, is cast as a mathematical programming problem with equilibrium constraints (MPECs).

Motivated by the need for simple, yet robust, approaches to solve this problem for practical, often large-scale structures, we attempt to take advantage of the increased availability of advanced and powerful software (and hardware) by proposing two simple algorithms apt to solve our problem via the GAMS modeling language and its associated nonlinear programming solvers CONOPT and MINOS.

Both algorithms, one based on a penalty approach (Algorithm 1) and the other on a parametric bounded weight approach (Algorithm 2), attempt to drive the complementarity term to zero. Algorithm 1 appears to perform slightly better than Algorithm 2 for practically-based problems. However, the former is superior on difficult problems such as those involving topology-type optimization. It is also our view that Algorithm 1 is more likely to furnish an optimum solution with minimum tuning. However, more extensive computational testing is required to verify these assertions.

Some interesting and useful extensions of the present work include: consideration of other structural types including necessarily nonlinear dependence on the design variables; procedures, such as some genetic-type search, likely to improve on the optimal solution; application to topology optimization taking special care of unloading so that nonholonomic behavior is simulated; and comparison of the algorithms presented in this paper with other (possibly more theoretically soundly based) MPEC algorithms.

Acknowledgments

This research is partly supported by the Australian Research Council.

Appendix: GAMS model for three-bar truss

```
sets   d    No. of structure dof              / d1*d2 /
       m    No. of members                    / m1*m3 /
       y    No. of yield functions per member / y1*y2 /;
alias  (y,y1);

scalar ro       Penalty factor  / 1e-2 /
       E        Young's modulus / 20e3 /
       sigma    Yield limit     / 50   /;

parameter L(m), F(d), C(m,d), N(m,y);
   L("m1") = 500; L("m2") = 400; L("m3") = 500;
   F("d1") = 400; F("d2") = 600;
   C("m1","d1") = 0.6; C("m1","d2") = 0.8;
   C("m2","d2") = 1;
   C("m3","d1") = -0.6; C("m3","d2") = 0.8;
   N(m,"y1") = 1; N(m,"y2") = -1;

variables           obj, S(m), r(m,y), H(m,y,y), Q(m), u(d);
positive variables  a(m), z(m,y), w(m,y);

H.fx(m,"y1","y2") = 0; H.fx(m,"y2","y1") = 0;
u.lo(d) = -4; u.up(d) = 4;

equations   cost, tech(m), stiff(m), limit(m,y),
            hard(m,y), compat(m), equil(d), yield(m,y);

cost        ..   obj =e= sum(m,L(m)*a(m))
                   + ro*sum((m,y),w(m,y)*z(m,y));
tech(m)     ..   a(m) =e= a("m1");
stiff(m)    ..   S(m) - E*a(m)/L(m) =e= 0;
limit(m,y)  ..   r(m,y) - sigma*a(m) =e= 0;
hard(m,y)   ..   H(m,y,y) - 0.125*E*a(m)/L(m) =e= 0;
compat(m)   ..   - Q(m) + S(m)*sum(d,C(m,d)*u(d))
                   - S(m)*sum(y,N(m,y)*z(m,y)) =e= 0;
equil(d)    ..   sum(m,C(m,d)*Q(m)) - F(d) =e= 0;
yield(m,y)  ..   - N(m,y)*Q(m) + sum(y1,H(m,y,y1)*z(m,y1))
                   + r(m,y) =e= w(m,y);

model truss / all /;
option nlp = conopt;

a.l(m) = 1; S.l(m) = 1; r.l(m,y) = 1; H.l(m,y,y) = 1;
```

```
scalar wz / 100 /;
set iter / iter1*iter50 /;
loop(iter$(wz gt 1e-8),
   solve truss using nlp minimizing obj;
   wz = sum((m,y),w.l(m,y)*z.l(m,y));
   ro = 10*ro;
);
```

References

[1] R. Levy and O. Lev, Recent developments in structural optimization, ASCE, Journal of Structural Engineering 113 (1987) 1939–1962.

[2] R.T. Haftka and Z. Gürdal, Elements of Structural Optimization, Kluwer, Dordrecht, Netherlands, 1992.

[3] M.P. Bendsøe and C.A. Mota Soares, eds., Topology Design of Structures, Kluwer, Dordrecht, Netherlands, 1993.

[4] P. Pedersen, ed., Optimal Design with Advanced Materials, Elsevier, Netherlands, 1993.

[5] G.I.N. Rozvany, ed., Optimization of Large Structural Systems, Vols. I and II, Kluwer, Dordrecht, Netherlands, 1993.

[6] N. Olhoff and G.I.N. Rozvany, eds., WCSMO-1 Structural and Multidisciplinary Optimization, Pergamon, Oxford, 1995.

[7] G.I.N. Rozvany, M.P. Bendsøe and U. Kirsch, Layout optimization of structures, Applied Mechanics Reviews 48 (1995) 41–119.

[8] I. Kaneko and G. Maier, Optimum design of plastic structures under displacement constraints, Computer Methods in Applied Mechanics and Engineering 27 (1981) 369–391.

[9] I. Kaneko, On some recent engineering applications of complementarity problems, Mathematical Programming Study 17 (1982) 111–125.

[10] C. Cinquini and R. Contro, Optimization of elastic-hardening structures in presence of displacement constraints, ASME, Journal of Mechanisms, Transmission, and Automation in Design 106 (1984) 179–182.

[11] Z.Q. Luo, J.S. Pang and D. Ralph, Mathematical Programs with Equilibrium Constraints, Cambridge University Press, 1996.

[12] S. Kaliszky and J. Lógó, Optimal plastic limit and shakedown design of bar structures with constraints on plastic deformation, Engineering Structures 19 (1997) 19–27.

[13] R. Muralidhar and J.R.J. Rao, New models for optimal truss topology in limit design based on unified elastic/plastic analysis, Computer Methods in Applied Mechanics and Engineering 140 (1997) 109–138.

[14] B.A. Murtagh and M.A. Saunders, MINOS 5.4 user's guide, Technical Report SOL 83-20R, Systems Optimization Laboratory, Stanford University, Stanford, California, 1995.

[15] A, Drud, CONOPT – a large-scale GRG code, ORSA Journal on Computing 6 (1994) 207–216.

[16] A. Brooke, D. Kendrick and A. Meeraus, GAMS: A User's Guide, Release 2.25, Boyd & Fraser Publishing Company, Massachusetts, 1992.

[17] G. Maier and J. Munro, Mathematical programming applications to engineering plastic analysis, Applied Mechanics Reviews 35 (1982) 1631–1643.

[18] G. Maier and D. Lloyd Smith, Update to "Mathematical programming applications to engineering plastic analysis", ASME, Applied Mechanics Update (1986) 377–383.

[19] L. Corradi, On compatible finite element models for elastic plastic analysis, Meccanica 13 (1978) 133–150.

[20] G. Maier, A matrix theory of piecewise linear elastoplasticity with interacting yield planes, Meccanica 5 (1970) 54–66.

[21] R.W. Cottle, J.S. Pang and R.E. Stone, The Linear Complementarity Problem, Academic Press, 1992.

[22] M.B. Fuchs, The explicit inverse of the stiffness matrix, International Journal of Solids and Structures 29 (1992) 2101–2113.

[23] G. Maier, S. Giacomini and F. Paterlini, Combined elastoplastic and limit analysis via restricted basis linear programming, Computer Methods in Applied Mechanics and Engineering 19 (1979) 21–48.

[24] Z.Q. Luo, J.S. Pang, D. Ralph and S.Q. Wu, Exact penalization and stationarity conditions of mathematical programs with equilibrium constraints, Mathematical Programming 75 (1996) 17–76.

[25] F. Tin-Loi and M.C. Ferris, A simple mathematical programming method to a structural identification problem, Proceedings, 7th International Conference on Computing in Civil and Building Engineering (ICCCBE-VII), Seoul, 19–21 Aug 97 (forthcoming).

[26] F. Tin-Loi, A GAMS model for the plastic analysis of plane frames, Applied Mathematical Modelling 17 (1993) 595–602.

[27] S.P. Dirkse and M.C. Ferris, Traffic modeling and variational inequalities using GAMS, Mathematical Programming Technical Report 97-06, Computer Sciences Department, University of Wisconsin, Madison, Wisconsin, 1997.

[28] S.P. Dirkse and M.C. Ferris, The PATH solver: a non-monotone stabilization scheme for mixed complementarity problems, Optimization Methods and Software 5 (1995) 123–156.

Table 1 – Results for Example 2

Case	Algorithm	ω	r	Time (secs)
1	1	19.860	2.468, 2.468, 2.468, 2.936, 2.936, 2.936	0.4
1	2	19.859	2.494, 2.494, 2.494, 2.918, 2.918, 2.918	1.4
1	Ref. [8]	19.887	2.684, 2.684, 2.684, 2.789, 2.789, 2.789	–
2	1	15.446	1.920, 1.920, 1.920, 2.283, 2.283, 2.283	0.2
2	2	15.446	1.921, 1.921, 1.921, 2.283, 2.283, 2.282	1.0
2	Ref. [8]	15.462	2.043, 2.043, 2.043, 2.200, 2.200, 2.200	–
3	1	10.933	0.000, 0.000, 0.000, 6.364, 0.000, 1.367	0.2
3	2	10.933	0.000, 0.000, 0.000, 6.364, 0.000, 1.367	0.9
3	Ref. [8]	13.091	1.410, 1.410, 1.410, 5.326, 0.470, 0.470	–

Table 2 – Results for Example 3

Case	Algorithm	ω	a (top, diagonal, bottom)	Time (secs)	Minor iter	Major iter
1	1	1922.23	15.017	18	954	2
1	2	1922.23	15.017	21	994	6
2	1	1812.46	18.538, 12.219, 12.039	23	1247	1
2	2	1812.46	18.538, 12.219, 12.039	32	1353	6
3	1	864.12	–	324	4311	2
3	2	864.00	–	596	10266	4
4	1	868.90	–	211	3264	2
4	2	846.81	–	974	12392	6

LIST OF FIGURES