INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2008; **75**:899–944
Published online 21 January 2008 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/nme.2275

Mesh adaptive computation of upper and lower bounds in limit analysis

H. Ciria[1], J. Peraire[1] and J. Bonet[2,*,†]

[1]*Aerospace Computational Design Laboratory, Department of Aeronautics, MIT, Cambridge, MA 02139, U.S.A.*
[2]*Civil & Computational Engineering Centre, School of Engineering, Swansea University, Swansea SA2 8PP, U.K.*

SUMMARY

An efficient procedure to compute strict upper and lower bounds for the exact collapse multiplier in limit analysis is presented, with a formulation that explicitly considers the exact convex yield condition. The approach consists of two main steps. First, the continuous problem, under the form of the static principle of limit analysis, is discretized twice (one per bound) using particularly chosen finite element spaces for the stresses and velocities that guarantee the attainment of an upper or a lower bound. The second step consists of solving the resulting discrete non-linear optimization problems. These are reformulated as second-order cone programs, which allows for the use of primal–dual interior point methods that optimally exploit the convexity and duality properties of the limit analysis model. To benefit from the fact that collapse mechanisms are typically highly localized, a novel method for adaptive meshing is introduced. The method first decomposes the total bound gap as the sum of positive contributions from each element in the mesh and then refines those elements with higher contributions. The efficiency of the methodology is illustrated with applications in plane stress and plane strain problems. Copyright © 2008 John Wiley & Sons, Ltd.

Received 5 June 2006; Revised 9 November 2007; Accepted 14 November 2007

KEY WORDS: limit analysis; bounds; conic programming; mesh adaptivity; finite elements

1. INTRODUCTION

Limit analysis is relevant in many practical engineering areas such as the design of mechanical structures and the analysis of soil mechanics [1–3]. Assuming a rigid, perfectly plastic solid subject to a static load distribution, the problem of limit analysis consists of finding the minimum multiple

*Correspondence to: J. Bonet, Civil & Computational Engineering Centre, School of Engineering, Swansea University, Swansea SA2 8PP, U.K.
†E-mail: j.bonet@swansea.ac.uk

Contract/grant sponsor: Singapore-MIT Alliance and DARPA-AFOSR; contract/grant number: F49620-03-1-0439

of this load distribution that will cause the body to collapse. This *collapse multiplier* results from solving an infinite dimensional saddle point problem, where the internal work rate is maximized over an admissible convex set of stresses—defined by a yield condition—in equilibrium with the applied loads and minimized over the linear space of kinematically admissible velocities for which the external work rate equals unity. This saddle point problem embeds the well-known convex (and equivalent) static and kinematic principles of limit analysis [4]. The presence of the yield condition introduces non-linearity in the problem, which represents an important difficulty.

Traditionally, the way to overcome this difficulty has been to linearize the convex yield condition, thereby obtaining a polyhedral approximation to the yield surface. With this linearization, first introduced in [5], the resulting problem reduces to a classical linear program (LP). Initially, the LP was solved using the simplex method [4, 6, 7] and, more recently, using interior point methods (IPM) [8, 9]. The first successful attempts to solve for the exact convex yield condition on fine grids were reported in [10], where the kinematic principle was discretized and then formulated as a minimization of sum of norms (MSN) subject to a linear constraint. Finally, the discrete problem was solved extending the ideas of IPMs for LP to the MSN. The approach, however, required the use of very cumbersome divergence-free elements when dealing with incompressible problems. This was overcome in [11], by simultaneously approximating the static and kinematic principles with a discrete duality problem that was solved using the method reported in [12]. This work was further improved in [13], by introducing automatic mesh refinement and using the primal–dual IPM developed in [14]. Unfortunately, the refinement strategy did not rely on rigorous local error measures but in heuristic estimates, thereby limiting its performance. A common feature of the above-mentioned works [10, 11, 13] is that they only provide approximations to the collapse multiplier, but do not yield strict bounds. In [15, 16], on the other hand, lower and upper bounds of the collapse multiplier are computed for soil mechanics problems on uniform meshes, using linear finite elements and a non-linear two-stage, quasi-Newton optimization algorithm. The method does not require the linearization of the yield condition, but can only handle smooth yield surfaces. Duality is not exploited and the solution process is not unified, since the upper and lower bounds are obtained by different versions of the quasi-Newton method. A new approach to obtain lower bounds is presented in [17]. It uses an IPM that exploits convexity and duality. This basic approach has been recently extended to the computation of upper bounds in [18]. More recently, an approach based on second-order cone programming (SOCP) has been followed in [19–21] using simplex elements for displacements and strains. Adaptive finite element procedures based on solution derivatives have been proposed in [22].

The main objective of the present work is to devise a unified and efficient method to compute upper and lower bounds, for the exact convex yield condition, using an approach that incorporates the methods presented in [17–20]. The convex nature of the limit analysis problem is exploited by solving the resulting optimization problems using standard conic programming (primal–dual) interior point algorithms. This guarantees efficiency in the solution process and global convergence to the optimal solutions. The domain is discretized using linear triangular finite element mesh on which two sets of specifically chosen interpolation spaces for the stresses and velocities are employed in turn in order to obtain an upper bound and a lower bound of the collapse load. In this way, the evaluation of both upper and lower bounds in this work makes it possible to derive rigorous mesh adaptive procedures, based on local error measures. These measures are based on a decomposition of the difference between upper and lower bounds, the bound gap, into positive local contributions. Areas that contribute significantly to the bound gap can then be identified and refined. The work presented here expands on the earlier results presented in [23, 24].

DOI: 10.1002/nme

2. THE LIMIT ANALYSIS PROBLEM: DUALITY AND EXACT BOUNDS

In this section, the main concepts of limit analysis as well as its mathematical description and main results are introduced. For its simplicity and ease of interpretation, the notation adopted corresponds with that used in [4, 25].

2.1. Notation and basic concepts

Let Ω denote the domain of study and $\partial\Omega$ its boundary, which consists of a Neumann portion Γ^{N}, where traction boundary conditions will apply, and a Dirichlet part Γ^{D}, where velocity boundary conditions will be enforced, so that $\partial\Omega=\Gamma^{\mathrm{N}}\cup\Gamma^{\mathrm{D}}$. Recall that limit analysis assumes a rigid, perfectly plastic material subject to a fixed load distribution (consisting of both surface and volume forces) and aims at computing the minimum multiple (collapse multiplier) of this load distribution that causes the collapse of the body.

The work rate of the external loads (*external work rate*) associated with a velocity or plastic flow $\mathbf{u}=\mathbf{u}(\mathbf{x})$ is given by the following linear functional:

$$F(\mathbf{u})=\int_{\Omega}\mathbf{f}\cdot\mathbf{u}\,\mathrm{d}V+\int_{\Gamma^{\mathrm{N}}}\mathbf{g}\cdot\mathbf{u}\,\mathrm{d}S \tag{1}$$

where $\mathbf{f}$ denotes the volume forces and $\mathbf{g}$ the surface forces acting on Γ^{N}. The internal work rate associated with the stress field $\boldsymbol{\sigma}=\boldsymbol{\sigma}(\mathbf{x})$ and the velocity $\mathbf{u}$ is given by the bilinear form

$$a(\boldsymbol{\sigma},\mathbf{u})=\int_{\Omega}\sum_{i,j}\sigma_{ij}\varepsilon_{ij}(\mathbf{u})\,\mathrm{d}V=\int_{\Omega}\sum_{i,j}\sigma_{ij}\frac{\partial u_i}{\partial x_j}\,\mathrm{d}V \tag{2}$$

where σ_{ij} and ε_{ij} denote the components of the stress and strain tensors in the x_i, $i=1:3$, cartesian coordinates.

Note that the plastic flow $\mathbf{u}$ must belong to an appropriate space $Y\equiv Y(\Omega)$ of kinematically admissible velocity fields. Similarly, the stress field $\boldsymbol{\sigma}$ is restricted to an appropriate space of symmetric stress tensors $X\equiv X(\Omega)$. The mathematical requirements on Y and X can be found in [25].

The equilibrium equation can now be expressed as the principle of virtual work:

$$a(\boldsymbol{\sigma},\mathbf{u})=F(\mathbf{u})\quad\forall\mathbf{u}\in Y \tag{3}$$

Moreover, the yield condition imposes the stress tensor $\boldsymbol{\sigma}$ to belong to a convex set $B(\mathbf{x})$ of admissible stresses for the material:

$$\boldsymbol{\sigma}(\mathbf{x})\in B(\mathbf{x})\quad\forall\mathbf{x}\in\Omega \tag{4}$$

To simplify the notation, the material will be assumed to be homogeneous, so that $B(\mathbf{x})$ is independent of $\mathbf{x}$ and can be denoted by B. The computational treatment of the limit analysis problem presented here exploits the convexity properties of B and requires, also, that B can be expressed in the generic form

$$B=\left\{\boldsymbol{\sigma}\in X\,\middle|\,\sum_{k}f_k^2(\sigma_{ij})\leqslant f_0^2(\sigma_{ij},q)\right\} \tag{5}$$

DOI: 10.1002/nme

where f_k and $f_0 \geqslant 0$ are affine functions of their arguments and q is a constant depending on the material properties. For instance, this structure is embedded in the von Mises (three-dimensional) yield condition, which is given by

$$B_{\mathrm{VM}} = \{\boldsymbol{\sigma} \in X \,|\, (\sigma_{11}-\sigma_{22})^2 + (\sigma_{22}-\sigma_{33})^2 + (\sigma_{33}-\sigma_{11})^2 + 6\sigma_{12}^2 + 6\sigma_{23}^2 + 6\sigma_{13}^2 \leqslant 2\sigma_{\mathrm{y}}^2\} \tag{6}$$

where σ_{y} denotes the yield stress in simple tension and can be interpreted as the generic constant q. Note that the yield set B might be unbounded, as is the case in the above example (any hydrostatic stress tensor satisfies the above inequality). The restriction of the von Mises yield condition to two-dimensional cases (either plane stress or plane strain) also satisfies structure (5). In plane strain, the same is valid for the Tresca (T), Mohr–Coulomb (MC) or Drucker–Prager (DP) models, which is given by

$$B_{\mathrm{T}} = \{\boldsymbol{\sigma} \in X \,|\, (\sigma_{11}-\sigma_{22})^2 + 4\sigma_{12}^2 \leqslant \sigma_y^2\} \tag{7}$$

$$B_{\mathrm{MC}} = \{\boldsymbol{\sigma} \in X \,|\, (\sigma_{11}-\sigma_{22})^2 + 4\sigma_{12}^2 \leqslant (2c\cos\phi - (\sigma_{11}+\sigma_{22})\sin\phi)^2,\ \sigma_{11}+\sigma_{22} \leqslant 2c\cot(\phi)\} \tag{8}$$

$$B_{\mathrm{DP}} = \{\boldsymbol{\sigma} \in X \,|\, (\sigma_{11}-\sigma_{22})^2 + 4\sigma_{12}^2 \leqslant (2\beta - 3\alpha(\sigma_{11}+\sigma_{22}))^2\} \tag{9}$$

where c is the material cohesion, ϕ denotes the friction angle and α and β refer to material properties. Clearly, (7)–(9) are convex sets in their variables σ_{ij} and can be expressed in the generic form (5).

2.2. The limit analysis problem, duality and exact bounds in continuous form

Let C denote the affine hyperplane $C = \{\mathbf{u} \in Y \,|\, F(\mathbf{u}) = 1\}$, consisting of those kinematically admissible velocity fields for which the external work rate equals 1. Now, the exact collapse multiplier, λ^*, which is the value to be bounded, results from solving any of the following equivalent problems:

$$\lambda^* = \sup \lambda \quad \text{s.t.} \begin{cases} \exists \boldsymbol{\sigma} \in B \\ a(\boldsymbol{\sigma}, \mathbf{u}) = \lambda F(\mathbf{u}) \ \ \forall \mathbf{u} \in Y \end{cases} \tag{10}$$

$$= \sup_{\boldsymbol{\sigma} \in B} \inf_{\mathbf{u} \in C} a(\boldsymbol{\sigma}, \mathbf{u}) \tag{11}$$

$$= \inf_{\mathbf{u} \in C} \sup_{\boldsymbol{\sigma} \in B} a(\boldsymbol{\sigma}, \mathbf{u}) \tag{12}$$

$$= \inf_{\mathbf{u} \in C} D(\mathbf{u}) \tag{13}$$

Problem (10) is the so-called *static principle of limit analysis*, whereas problem (13) is known as the *kinematic principle of limit analysis*. The equivalence between (10) and (11) follows from the linearity in $\mathbf{u}$ of both functionals $a(\boldsymbol{\sigma}, \mathbf{u})$ and $F(\mathbf{u})$. Additionally, the equality between (11) and (12) holds, thanks to strong duality, which is proved in detail in [25]. Finally, $D(\mathbf{u})$ is defined as the inner supremum in (12) and represents the total energy dissipation rate associated with $\mathbf{u}$. Moreover, Christiansen [25] also shows that collapse fields $\mathbf{u}^*$ and $\boldsymbol{\sigma}^*$ exist and are a saddle point of $a(\boldsymbol{\sigma}, \mathbf{u})$.

DOI: 10.1002/nme

2.3. The limit analysis problem, duality and exact bounds in discrete form

Let Ω_h denote a discretization of the domain of study, Ω, and let X_h and Y_h be finite element function spaces that approximate the continuous stress and velocity fields: $\boldsymbol{\sigma} \approx \boldsymbol{\sigma}_h \in X_h$ and $\mathbf{u} \approx \mathbf{u}_h \in Y_h$. Moreover, the discrete convex set of admissible stresses, B_h, must be such that $B_h \subseteq B \cap X_h$, and the affine hyperplane to which $\mathbf{u}_h$ is restricted now becomes $C_h = \{\mathbf{u}_h \in Y_h | F(\mathbf{u}_h) = 1\}$. Then, the discretized version of the variational continuous limit analysis problem (10)–(13) reads as follows:

$$\lambda_h^* = \max \lambda$$

$$\text{s.t.} \begin{cases} \exists \boldsymbol{\sigma}_h \in B_h \\ a(\boldsymbol{\sigma}_h, \mathbf{u}_h) = \lambda F(\mathbf{u}_h) \quad \forall \mathbf{u}_h \in Y_h \end{cases} \tag{14}$$

$$= \max_{\boldsymbol{\sigma}_h \in B_h} \min_{\mathbf{u}_h \in C_h} a(\boldsymbol{\sigma}_h, \mathbf{u}_h) \tag{15}$$

$$= \min_{\mathbf{u}_h \in C_h} \max_{\boldsymbol{\sigma}_h \in B_h} a(\boldsymbol{\sigma}_h, \mathbf{u}_h) \tag{16}$$

$$= \min_{\mathbf{u}_h \in C_h} D_h(\mathbf{u}_h) \tag{17}$$

The above discrete duality holds for all practical discretizations (see proof in [25]). Note that, thanks to discrete duality, the approximated value λ_h^* to the exact collapse multiplier, λ^*, can be obtained in several different ways. Indeed, one can compute λ_h^* by solving the discrete static principle (14) or the discrete kinematic principle (17). Also, both problems can be solved at the same time using a primal–dual optimization method to approximate simultaneously the collapse fields for the stresses and velocities. This is the approach that will be followed here.

In general, for a given choice of $X_h \times Y_h$, λ_h^* is only an estimate or approximation to λ^*, but not a bound. However, there exist particular combinations of appropriately chosen interpolation spaces $X_h \times Y_h$ for both stresses and velocities that, when used to discretize the continuous problem (10)–(13), lead to discretizations (14)–(17) whose solution, λ_h^*, is guaranteed to be either a lower bound ($\lambda_h^{*\mathrm{LB}} \leqslant \lambda^*$) or an upper bound ($\lambda^* \leqslant \lambda_h^{*\mathrm{UB}}$) of the true collapse multiplier, λ^* (and not only an estimate). In particular, the combination of spaces that guarantee the attainment of lower bounds will be named *purely static spaces*, whereas those that yield upper bounds will be named *purely kinematic spaces*. The requirements on these particular spaces are given as follows.

Purely static spaces: $X_h \times Y_h$ is said to be a combination of purely static spaces and is denoted by $X_h^{\mathrm{LB}} \times Y_h^{\mathrm{LB}}$, if the following two conditions hold:

1. Satisfaction of the discrete equilibrium equation on X_h^{LB} implies the continuous equilibrium equation. This is equivalent to the following implication for any $\boldsymbol{\sigma}_h \in X_h^{\mathrm{LB}}$:

$$a(\boldsymbol{\sigma}_h, \mathbf{u}_h) = \lambda F(\mathbf{u}_h) \quad \forall \mathbf{u}_h \in Y_h^{\mathrm{LB}} \implies a(\boldsymbol{\sigma}_h, \mathbf{u}) = \lambda F(\mathbf{u}) \quad \forall \mathbf{u} \in Y \tag{18}$$

2. Satisfaction of the membership condition $\boldsymbol{\sigma}_h \in B_h^{\mathrm{LB}} \subseteq B \cap X_h^{\mathrm{LB}}$ in some discrete or test points directly implies that $\boldsymbol{\sigma}_h \in B$ over the continuum.

Thus, the advantage of using purely static spaces is that, by only forcing discrete satisfaction of equilibrium and membership, one can guarantee equilibrium and membership over the whole domain. Therefore, the constraints in the maximization problem (10) are satisfied exactly, which

DOI: 10.1002/nme

means that the inner infimum in (11) is also computed exactly for all the stresses belonging to an appropriate set. Note that the above conditions imply that

$$\max_{\boldsymbol{\sigma}\in B_h}\inf_{\mathbf{u}\in C} a(\boldsymbol{\sigma},\mathbf{u}) = \max_{\boldsymbol{\sigma}\in B_h}\min_{\mathbf{u}\in C_h} a(\boldsymbol{\sigma},\mathbf{u}) \tag{19}$$

Therefore, the use of purely static spaces provides a computable strict lower bound as

$$\lambda^* = \sup_{\boldsymbol{\sigma}\in B}\inf_{\mathbf{u}\in C} a(\boldsymbol{\sigma},\mathbf{u}) \geqslant \max_{\boldsymbol{\sigma}\in B_h}\inf_{\mathbf{u}\in C} a(\boldsymbol{\sigma},\mathbf{u}) = \max_{\boldsymbol{\sigma}\in B_h}\min_{\mathbf{u}\in C_h} a(\boldsymbol{\sigma},\mathbf{u}) \equiv \lambda_h^{*\mathrm{LB}} \tag{20}$$

Purely kinematic spaces: $X_h \times Y_h$ is said to be a combination of purely kinematic spaces and is denoted by $X_h^{\mathrm{UB}} \times Y_h^{\mathrm{UB}}$, if the discrete energy dissipation rate, $D_h(\mathbf{u}_h)$, is exact on Y_h^{UB}:

$$\max_{\boldsymbol{\sigma}_h\in B_h} a(\boldsymbol{\sigma}_h,\mathbf{u}_h) = \sup_{\boldsymbol{\sigma}\in B} a(\boldsymbol{\sigma},\mathbf{u}_h) \quad \forall \mathbf{u}_h \in Y_h \tag{21}$$

Therefore, a computable strict upper bound is obtained as

$$\lambda^* = \inf_{\mathbf{u}\in C}\sup_{\boldsymbol{\sigma}\in B} a(\boldsymbol{\sigma},\mathbf{u}) \leqslant \min_{\mathbf{u}\in C_h}\sup_{\boldsymbol{\sigma}\in B} a(\boldsymbol{\sigma},\mathbf{u}) = \min_{\mathbf{u}\in C_h}\max_{\boldsymbol{\sigma}\in B_h} a(\boldsymbol{\sigma},\mathbf{u}) \equiv \lambda_h^{*\mathrm{UB}} \tag{22}$$

Hence, the use of purely kinematic spaces results in an upper bound method. Figure 1 summarizes graphically the above explanations.

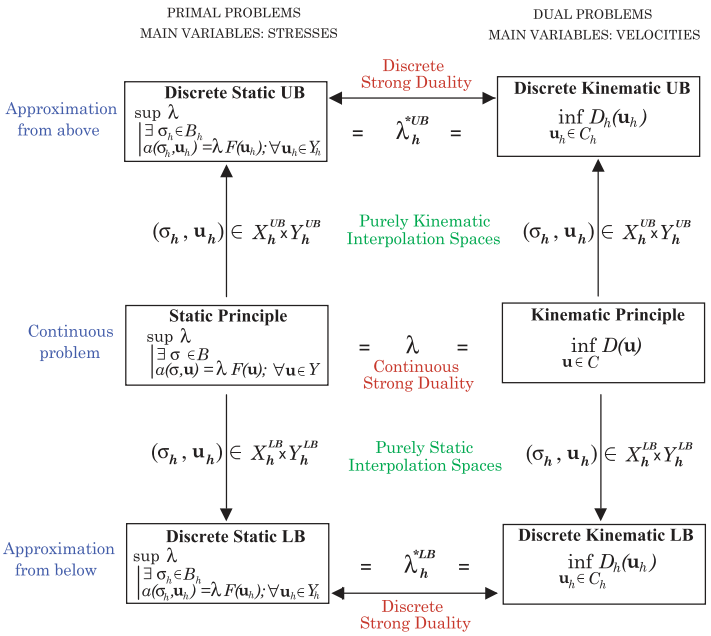


Figure 1. The use of purely static and purely kinematic spaces to obtain lower and upper bounds.

DOI: 10.1002/nme

3. SOCP WITHIN THE GENERAL FRAMEWORK OF CONIC PROGRAMMING

IPMs for the efficient solution of convex optimization problems have been developed in recent years. Therefore, recognizing the convex structure of the problem is important since it guarantees an efficient solution process. A particular class of convex problems are the so-called conic programs (CP) [26]. Any CP problem can be expressed in the following canonical primal (P) or dual (D) form:

$$\text{(P)}\ \min\{c^{\mathrm{T}}x \mid Ax=b,\ x\in\mathscr{K}\} \tag{23}$$

$$\text{(D)}\ \max\{b^{\mathrm{T}}y \mid A^{\mathrm{T}}y+s=c,\ s\in\mathscr{K}_*\} \tag{24}$$

where $x\in\Re^n$ is the vector of decision variables, $c\in\Re^n$, $b\in\Re^m$, $A\in\Re^{m\times n}$ are given data, $s\in\Re^n$ is a vector of slack variables, $\mathscr{K}\subset\Re^n$ is a pointed and closed convex cone with a non-empty interior and $\mathscr{K}_*=\{s\in\Re^n \mid s^{\mathrm{T}}x\geqslant 0, \forall x\in\mathscr{K}\}$ is the dual cone to $\mathscr{K}$ and is also a closed convex pointed cone with a non-empty interior. The most relevant cones satisfying these properties are the so-called canonical cones: the positive orthant, $\Re^n_+$, the Lorentz (or second-order or ice cream) cone, $\mathscr{L}^n$, and the positive-semidefinite cone, $\mathscr{S}^n_+$. In particular, a Lorentz cone is defined as

$$\mathscr{K}\equiv\mathscr{L}^n=\left\{x\in\Re^n \mid x_1\geqslant\sqrt{\sum_{i=2}^{n}x_i^2}\right\} \tag{25}$$

When the cone $\mathscr{K}$ is one of the canonical cones, then the following important subclasses of conic programming arise: linear programming (LP) when $\mathscr{K}\equiv\Re^n_+$, SOCP (or conic quadratic) when $\mathscr{K}\equiv\mathscr{L}^n$ and semidefinite programming (SDP) when $\mathscr{K}\equiv\mathscr{S}^n_+$. Note that LP is a particular case of SOCP which, in turn, can always be cast as an SDP program.

The most fundamental result of conic programming is the conic duality theorem, over which the interior point algorithms used in the solution process are built. Provided that either (P) or (D) is strictly feasible and bounded, the main practical implication of the theorem is that an optimal primal–dual feasible pair (x^*, y^*) exists and the optimal values of (P) and (D) are equal to each other, that is, $b^{\mathrm{T}}y^*=c^{\mathrm{T}}x^*$ (zero duality gap).

CP problems are generally solved using IPMs (see [27] for an extensive and deep treatment). Currently, the primal–dual path-following IPM (with its variants) is the method of choice in commercial implementations, thanks to its excellent performance in large-scale applications and, especially, in the presence of sparsity. The method is based on duality, and search directions for optimality are computed in both primal and dual feasible spaces. Typically, the search directions used in CP are AHO [28], HKM [29] and NT [30]. In particular, for SOCP, the NT option is the one that generally offers the best computational performance. Note finally that SOCP is slower to solve than LP problems, but notably faster than SDP problems, especially if sparsity is exploited.

4. BOUND PROBLEMS: METHODOLOGY, IMPLEMENTATION AND SOLUTION

This section presents the methodology that has been used to compute bounds to the exact collapse multiplier, λ^*. The bound problems solved here correspond to the two-dimensional version of the

DOI: 10.1002/nme

limit analysis problem (10)–(13), and both plane stress and plane strain models are addressed. Only the von Mises model has been used, but the method could be extended to other criteria satisfying criterion (5). Its restriction to plane stress, B_1, and plane strain, B_2, expressed in the format of (5) reads as follows:

$$B_1 = \{\boldsymbol{\sigma} \in X | (\sigma_{11} - \sigma_{22})^2 + \sigma_{11}^2 + \sigma_{22}^2 + 6\sigma_{12}^2 \leqslant 2\sigma_y^2\} \tag{26}$$

$$B_2 = \{\boldsymbol{\sigma} \in X | (\sigma_{11} - \sigma_{22})^2 + 4\sigma_{12}^2 \leqslant \tfrac{4}{3}\sigma_y^2\} \tag{27}$$

Let $\mathscr{T}_h$ denote a triangulation consisting of E triangular elements, Ω^e, that form a partition of the body. The boundary of the element Ω^e is denoted by $\partial\Omega^e$. Let $\mathscr{E}$ be the set of all the edges in the mesh, which is decomposed into the following three disjoint sets: $\mathscr{E} = \mathscr{E}^{\mathscr{O}} \cup \mathscr{E}^{\mathscr{D}} \cup \mathscr{E}^{\mathscr{N}}$, where $\mathscr{E}^{\mathscr{O}} = \{\xi_e^{e'} | \xi_e^{e'} = \partial\Omega^e \cap \partial\Omega^{e'}; \forall e, e' \in \mathscr{T}_h\}$ (set of interior edges), $\mathscr{E}^{\mathscr{D}} = \{\xi_e^{\mathrm{D}} | \xi_e^{\mathrm{D}} = \partial\Omega^e \cap \Gamma^{\mathrm{D}}; \forall e \in \mathscr{T}_h\}$ (set of edges associated with Dirichlet boundaries) and $\mathscr{E}^{\mathscr{N}} = \{\xi_e^{\mathrm{N}} | \xi_e^{\mathrm{N}} = \partial\Omega^e \cap \Gamma^{\mathrm{N}}; \forall e \in \mathscr{T}_h\}$ (set of edges associated with Neumann boundaries). For simplicity of presentation, it will be assumed that a given edge is of either a Dirichlet type or a Neumann type. The extension to mixed boundary conditions is straightforward.

4.1. The lower bound problem

4.1.1. Purely static spaces. For both plane stress and plane strain cases, one can show that a possible purely static formulation results from approximating the stress field with piecewise discontinuous stresses that are linearly interpolated within the elements, and the velocity field with constants on each element together with additional linear interpolations along the inter-element boundaries. These stresses can be expressed as

$$\sigma_{hi}^e(x_1, x_2) = \sum_{a=1}^{3} \sigma_i^{a,e} N_a^e(x_1, x_2), \quad i = 1:3 \tag{28}$$

where $a = 1:3$ refers to a local numbering of each of the three nodes of the element, $\sigma_i^{a,e}$ ($\sigma_1 \equiv \sigma_{11}$, $\sigma_2 \equiv \sigma_{22}$, $\sigma_3 \equiv \sigma_{12}$) are the nodal stress values and $N_a^e(\mathbf{x})$ is a linear shape function valuing 1 at the node a and 0 at the other two nodes. Clearly, the interpolation space introduces nine stress variables or unknowns per triangle ($\sigma_i^a; i = 1:3, a = 1:3$), thereby enabling different stress tensors to arise in each node of the mesh, as many as elements sharing that node.

The approximation of the velocity field, $\mathbf{u}_h \in Y_h^{\mathrm{LB}} \subset Y$, Y_h^{LB} introduces two unknowns per triangle (constant approximation for each element) and four additional degrees of freedom per edge (linear approximation). This is shown in the following (global and local) interpolations:

$$u_{hi}(x_1, x_2) = \sum_{e=1}^{E} u_i^e \psi_e(x_1, x_2), \quad i = 1:2 \tag{29}$$

$$u_{hi}^{\xi}(s) = \sum_{\alpha=1}^{2} u_i^{\alpha,\xi} N_\alpha^\xi(s), \quad i = 1:2 \tag{30}$$

where $\psi_e(\mathbf{x}) = 1$, $\forall \mathbf{x} \in \Omega^e$ and $\psi_e(\mathbf{x}) = 0$ otherwise, $N_\alpha^\xi(s)$ is a linear shape function over the edge ξ, s denotes a local coordinate and $\alpha = 1:2$ are the two nodes of the edge. Assuming that

DOI: 10.1002/nme

the body forces, **f**, are constant and the tractions, **g**, vary at most linearly, then the interpolation space Y_h^{LB} is rich enough, when compared with X_h^{LB}, to force equilibrium point by point. The formal proof is detailed in Reference [24] but a physical explanation follows from the fact that the imposition of the piecewise constant velocities (29) translates into an internal equilibrium equation in strong form for each element, which can be satisfied exactly because the body force is constant over the element and the stresses vary linearly; hence, their divergence is also constant. Additionally, the imposition of the linear velocities (30) over the edges of the elements impose continuity of tractions between adjacent elements and compatibility with the Neumann boundary conditions at the external edges of the boundary elements. These can be satisfied exactly because both the external tractions and the stress tensors vary linearly along the edges of the elements.

The stress field $\boldsymbol{\sigma}_h$ is required to belong to the set of admissible stresses, $B_{\delta,h}$, in every point of the domain Ω, that is, $\boldsymbol{\sigma}_h(\mathbf{x}) \in B_{\delta,h}, \forall \mathbf{x} \in \Omega$; where $B_{\delta,h} = B_\delta \cap X_h^{\mathrm{LB}}$ and $\delta = 1:2$ (plane stress or plane strain). Fortunately, this general pointwise condition can be satisfied by just forcing the membership constraint to hold over the nodes of each triangle, which translates into $3 \times E$ nodal inequalities. This nice property is due to the choice of a piecewise linear interpolation of the stresses and the use of triangular elements. The proof is a simple consequence of convexity and is given in Reference [24]. Note that, for other element types such as quadratic or bi-quadratic elements, the yield condition might be violated between nodes even if it is satisfied at all nodes.

4.1.2. Discretization process. Recall that the attainment of a lower bound or an upper bound depends only on the interpolation spaces used and not on the particular version of the limit analysis problem that is discretized (see Figure 1). In particular, the *static principle of limit analysis*, (10), has been chosen to be the problem to discretize, using the (previously described) purely static spaces, $X_h^{\mathrm{LB}} \times Y_h^{\mathrm{LB}}$. This choice corresponds to the lower left branch of Figure 1, leading to the formulation of the '*discrete static LB*' problem.

The static principle (14) can be exclusively expressed in terms of stresses belonging to the space X_h^{LB} as follows:

$$\begin{aligned} \max \quad & \lambda \\ \text{s.t.} \quad & \begin{cases} \nabla \cdot \boldsymbol{\sigma}_h^e + \lambda \mathbf{f}^e = \mathbf{0} & \text{in } \Omega^e \ \ \forall e \in \mathscr{T}_h \\ (\boldsymbol{\sigma}_h^e - \boldsymbol{\sigma}_h^{e'}) \cdot \mathbf{n}^{\xi e'}_{\xi e} = \mathbf{0} & \forall \xi_e^{e'} \in \mathscr{E}^{\mathscr{O}} \\ \boldsymbol{\sigma}_h^e \cdot \mathbf{n}^{\xi \mathrm{N}}_{\xi e} = \lambda \mathbf{g}^{\xi \mathrm{N}}_{\xi e} & \forall \xi_e^{\mathrm{N}} \in \mathscr{E}^{\mathscr{N}} \\ \boldsymbol{\sigma}_h^e \in B_{\delta,h} & \text{in } \Omega^e \ \ \forall e \in \mathscr{T}_h \end{cases} \end{aligned} \tag{31}$$

Next, the discretization of each of the above constraints is addressed.

Elemental equilibrium constraints. For each triangle Ω^e, the local interpolation (28) is inserted into the vectorial equation $\nabla \cdot \boldsymbol{\sigma}^e + \lambda \mathbf{f}^e = \mathbf{0}$. Grouping all the nodal stress components into a single vector and using the following notation for the (constant) derivatives of the (linear) shape functions

DOI: 10.1002/nme

$N^e_{a,i} = \partial N^e_a(\mathbf{x})/\partial x_i$, where $i = 1:2$, the following matrix equation emerges:

$$\begin{pmatrix} N^e_{1,1} & 0 & N^e_{1,2} & N^e_{2,1} & 0 & N^e_{2,2} & N^e_{3,1} & 0 & N^e_{3,2} \\ 0 & N^e_{1,2} & N^e_{1,1} & 0 & N^e_{2,2} & N^e_{2,1} & 0 & N^e_{3,2} & N^e_{3,1} \end{pmatrix} \begin{pmatrix} \sigma^{1,e}_1 \\ \sigma^{1,e}_2 \\ \sigma^{1,e}_3 \\ \sigma^{2,e}_1 \\ \sigma^{2,e}_2 \\ \sigma^{2,e}_3 \\ \sigma^{3,e}_1 \\ \sigma^{3,e}_2 \\ \sigma^{3,e}_3 \end{pmatrix} + \lambda \begin{pmatrix} f^e_1 \\ f^e_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{32}$$

or, equivalently in compact form,

$$\underline{\underline{B}}^e \underline{\sigma}^e_h + \lambda \underline{F}^{\mathrm{eq1},e}_h = \underline{0} \tag{33}$$

Now, let $\underline{\sigma}_h$ denote a global vector collecting the $9 \times E$ nodal stress components for all the elements in the mesh, and $\underline{F}^{\mathrm{eq1}}_h$ be a $2 \times E$ global volume force vector. Similarly, a $(2 \times E, 9 \times E)$ global matrix $\underline{\underline{A}}^{\mathrm{eq1}}$, consisting of the elemental matrices $\underline{\underline{B}}^e$, is built. The assembly process is straightforward since the equations for the elements are uncoupled. Consequently, $\underline{\underline{A}}^{\mathrm{eq1}}$ results in a very sparse block-diagonal matrix:

$$\underline{\underline{A}}^{\mathrm{eq1}} = \begin{pmatrix} \underline{\underline{B}}^1 & \underline{\underline{0}} & \cdots & \cdots & \underline{\underline{0}} \\ \underline{\underline{0}} & \underline{\underline{B}}^2 & \underline{\underline{0}} & \cdots & \underline{\underline{0}} \\ \vdots & & \ddots & & \vdots \\ \vdots & & & \ddots & \vdots \\ \underline{\underline{0}} & \cdots & \cdots & \cdots & \underline{\underline{B}}^E \end{pmatrix} \tag{34}$$

where $\underline{\underline{0}}$ is a zero matrix of dimensions $(2, 9)$. Finally, the global discrete system of equations corresponding to the elemental equilibrium constraint is given by

$$\underline{\underline{A}}^{\mathrm{eq1}} \underline{\sigma}_h + \underline{F}^{\mathrm{eq1}}_h \lambda = \underline{0} \tag{35}$$

Inter-element and boundary equilibrium constraints: The restriction of the linear interpolation (28) to an edge, ξ, can be expressed as follows:

$$\sigma_i(s) = \sum_{\alpha=1}^{2} \sigma^\alpha_i N^\xi_\alpha(s), \quad i = 1:3 \tag{36}$$

DOI: 10.1002/nme

Now, inserting interpolation (36) into the equations $(\boldsymbol{\sigma}_h^e - \boldsymbol{\sigma}_h^{e'}) \cdot \mathbf{n}^{\xi_e^{e'}} = \mathbf{0}, \forall \xi_e^{e'} \in \mathscr{E}^{\mathscr{O}}$ and $\boldsymbol{\sigma}_h^e \cdot \mathbf{n}^{\xi_e^{\mathrm{N}}} = \lambda \mathbf{g}^{\xi_e^{\mathrm{N}}}, \forall \xi_e^{\mathrm{N}} \in \mathscr{E}^{\mathscr{N}}$, one obtains, after some simple algebra, a system of four scalar equations for each vector equation. The four equations imposed in each interior edge, $\xi_e^{e'} \in \mathscr{E}^{\mathscr{O}}$, are

$$(\sigma_1^{1,e} - \sigma_1^{1,e'}) n_1^{\xi} + (\sigma_3^{1,e} - \sigma_3^{1,e'}) n_2^{\xi} = 0 \tag{37}$$

$$(\sigma_1^{2,e} - \sigma_1^{2,e'}) n_1^{\xi} + (\sigma_3^{2,e} - \sigma_3^{2,e'}) n_2^{\xi} = 0 \tag{38}$$

$$(\sigma_3^{1,e} - \sigma_3^{1,e'}) n_1^{\xi} + (\sigma_2^{1,e} - \sigma_2^{1,e'}) n_2^{\xi} = 0 \tag{39}$$

$$(\sigma_3^{2,e} - \sigma_3^{2,e'}) n_1^{\xi} + (\sigma_2^{2,e} - \sigma_2^{2,e'}) n_2^{\xi} = 0 \tag{40}$$

Similarly, the following four equations are imposed on each boundary edge, $\xi_e^{\mathrm{N}} \in \mathscr{E}^{\mathscr{N}}$:

$$\sigma_1^{1,e} n_1^{\xi} + \sigma_3^{1,e} n_2^{\xi} = \lambda g_1^{\xi} \tag{41}$$

$$\sigma_1^{2,e} n_1^{\xi} + \sigma_3^{2,e} n_2^{\xi} = \lambda g_1^{\xi} \tag{42}$$

$$\sigma_3^{1,e} n_1^{\xi} + \sigma_2^{1,e} n_2^{\xi} = \lambda g_2^{\xi} \tag{43}$$

$$\sigma_3^{2,e} n_1^{\xi} + \sigma_2^{2,e} n_2^{\xi} = \lambda g_2^{\xi} \tag{44}$$

Clearly, the total number of edge equations is $4 \times (|\mathscr{E}^{\mathscr{O}}| + |\mathscr{E}^{\mathscr{N}}|)$. The systems of equations (37)–(40) and (41)–(44) can be combined into a global matrix linear equation as follows:

$$\underline{\underline{A}}^{\mathrm{eq2}} \underline{\sigma}_h + \underline{F}_h^{\mathrm{eq2}} \lambda = \underline{0} \tag{45}$$

where $\underline{\sigma}_h$ is the $9 \times E$ vector of unknown nodal stresses previously used. Moreover, $\underline{\underline{A}}^{\mathrm{eq2}}$ is a $(4 \times (|\mathscr{E}^{\mathscr{O}}| + |\mathscr{E}^{\mathscr{N}}|), 9 \times E)$ matrix, whose entries are either zero or the components n_1^{ξ}, n_2^{ξ} of the unit vectors normal to the edges with the appropriate sign. Finally, $\underline{F}_h^{\mathrm{eq2}}$ is a vector consisting of zeros, for the equations associated with the interior edges, or minus the components g_1^{ξ}, g_2^{ξ} of the external surface loads, for the Neumann boundary edges.

Membership constraints. Formulation as multiple second-order cones: The yield condition is not violated at any point of the domain if the inequality

$$(\sigma_1^{a,e} - \sigma_2^{a,e})^2 + (\sigma_1^{a,e})^2 + (\sigma_2^{a,e})^2 + 6(\sigma_3^{a,e})^2 \leqslant 2\sigma_y^2 \quad (\delta = 1, \text{plane stress}) \tag{46}$$

or

$$(\sigma_1^{a,e} - \sigma_2^{a,e})^2 + 4(\sigma_3^{a,e})^2 \leqslant \tfrac{4}{3}\sigma_y^2 \quad (\delta = 2, \text{plane strain}) \tag{47}$$

is satisfied at every node, a, of each element, e, of the mesh ($3 \times E$ inequalities). A convenient way to impose inequality (46) or (47) is to force the vector $(\sqrt{2}\sigma_y, \sigma_1^{a,e}, \sigma_2^{a,e}, \sqrt{6}\sigma_3^{a,e}, \sigma_1^{a,e} - \sigma_2^{a,e})$ (plane stress) or $(2/\sqrt{3}\sigma_y, 2\sigma_3^{a,e}, \sigma_1^{a,e} - \sigma_2^{a,e})$ (plane strain) to belong to the Lorentz cone $\mathscr{L}^n$, where $n = 5$ for plane stress and $n = 3$ for plane strain. Since second-order cone constraints are

DOI: 10.1002/nme

directly imposed through the decision variables (see Section 3), it is convenient to introduce a vector of additional variables, $\underline{x}^{a,e}$, as follows:

$$\text{plane stress} \begin{cases} x_1^{a,e} = \sqrt{2}\sigma_y \\ -\sigma_1^{a,e} + x_2^{a,e} = 0 \\ -\sigma_2^{a,e} + x_3^{a,e} = 0 \\ -\sqrt{6}\sigma_3^{a,e} + x_4^{a,e} = 0 \\ -\sigma_1^{a,e} + \sigma_2^{a,e} + x_5^{a,e} = 0 \end{cases} \quad , \quad \text{plane strain} \begin{cases} x_1^{a,e} = \dfrac{2}{\sqrt{3}}\sigma_y \\ -2\sigma_3^{a,e} + x_2^{a,e} = 0 \\ -\sigma_1^{a,e} + \sigma_2^{a,e} + x_3^{a,e} = 0 \end{cases} \tag{48}$$

Now, each vector $\underline{x}^{a,e}$ is forced to belong to $\mathscr{L}^n$. The imposition of (48) over all the mesh requires $3n \times E$ equations (n equations per node and three nodes per element). In matrix notation, the global system can be expressed as follows:

$$\underline{\underline{A}}_{\delta}^{\text{soc}} \underline{\sigma}_h + \underline{\underline{I}}_{\delta} \underline{x}_{\delta}^{\text{soc}} = \underline{b}_{\delta}^{\text{soc}} \tag{49}$$

where $\underline{\sigma}_h$ is the usual vector of unknown nodal stresses, $\underline{\underline{I}}_{\delta}$ is a $(3n \times E, 3n \times E)$ identity matrix, $\underline{x}_{\delta}^{\text{soc}}$ is a vector of $3n \times E$ additional variables ordered in the same way as $\underline{\sigma}_h$, and $\underline{b}_{\delta}^{\text{soc}}$ and $\underline{\underline{A}}_{\delta}^{\text{soc}}$ are, respectively, a $3n \times E$ vector and a $(3n \times E, 9 \times E)$ block-diagonal matrix of the following forms:

$$\underline{b}_{\delta}^{\text{soc}} = \begin{pmatrix} \underline{b}_{\delta}^{a,e} \\ \underline{b}_{\delta}^{a,e} \\ \vdots \\ \vdots \\ \underline{b}_{\delta}^{a,e} \end{pmatrix}, \quad \underline{\underline{A}}_{\delta}^{\text{soc}} = \begin{pmatrix} \underline{\underline{M}}_{\delta} & \underline{\underline{0}} & \cdots & \cdots & \underline{\underline{0}} \\ \underline{\underline{0}} & \underline{\underline{M}}_{\delta} & \underline{\underline{0}} & \cdots & \underline{\underline{0}} \\ \vdots & & \ddots & & \vdots \\ \vdots & & & \ddots & \vdots \\ \underline{\underline{0}} & \cdots & \cdots & \cdots & \underline{\underline{M}}_{\delta} \end{pmatrix}, \quad \delta = 1:2 \tag{50}$$

where

$$\underline{b}_1^{a,e} = \begin{pmatrix} \sqrt{2}\sigma_y \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad \underline{\underline{M}}_1 = \begin{pmatrix} 0 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -\sqrt{6} \\ -1 & 1 & 0 \end{pmatrix}, \quad \underline{b}_2^{a,e} = \begin{pmatrix} \dfrac{2}{\sqrt{3}}\sigma_y \\ 0 \\ 0 \end{pmatrix}, \quad \underline{\underline{M}}_2 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -2 \\ -1 & 1 & 0 \end{pmatrix} \tag{51}$$

4.1.3. The lower bound problem as an SOCP. Expressing the previous matrix constraints (35), (45) and (49) in a single matrix equation, the discretization of the lower bound problem (31)

DOI: 10.1002/nme

is given by

$$\lambda_h^{*\mathrm{LB}} \equiv \max \quad \lambda$$

$$\text{s.t.} \quad \left\{ \begin{array}{l} \overbrace{\begin{pmatrix} \underline{\underline{A}}^{\mathrm{eq1}} & \vdots & \underline{\underline{F}}_h^{\mathrm{eq1}} & \vdots & \underline{\underline{0}} \\ \underline{\underline{A}}^{\mathrm{eq2}} & \vdots & \underline{\underline{F}}_h^{\mathrm{eq2}} & \vdots & \underline{\underline{0}} \\ \underline{\underline{A}}_\delta^{\mathrm{soc}} & \vdots & \underline{\underline{0}} & \vdots & \underline{\underline{I}}_\delta \end{pmatrix}}^{9\times E \quad + \quad 1 \quad + \quad 3n\times E} \begin{pmatrix} \underline{\sigma}_h \\ \lambda \\ \underline{x}_\delta^{\mathrm{soc}} \end{pmatrix} = \begin{pmatrix} \underline{0} \\ \underline{0} \\ \underline{b}_\delta^{\mathrm{soc}} \end{pmatrix} \left\} \begin{array}{l} m_1 = 2\times E \\ m_2 = 4\times(|\mathscr{E}^{\mathscr{O}}|+|\mathscr{E}^{\mathscr{N}}|) \\ m_3 = 3n\times E \end{array} \right. \\ \underline{\sigma}_h \text{ free},\ \lambda \text{ free},\ \underline{x}_\delta^{\mathrm{soc}} \in \mathscr{K} \end{array} \right. \tag{52}$$

where $\mathscr{K} = \overbrace{\mathscr{L}^n \times \cdots \times \mathscr{L}^n}^{3\times E}$.

Note that this problem is a CP and has the standard form required by most optimization packages. However, in order to optimize the computational cost of the solution, it is possible to introduce a change of variables that reduces considerably the number of equations involved. Algorithmic details of this process are given in Reference [24]. In addition, from Section 3, and given that the lower bound problem has a strictly feasible interior, it is clear that strong duality holds. Therefore, $\lambda_h^{*\mathrm{LB}}$ can also be obtained by solving the dual problem to (52), which takes the following form:

$$\lambda_h^{*\mathrm{LB}} \equiv \min \quad \underline{b}_\delta^{\mathrm{soc}^{\mathrm{T}}} \underline{y}$$

$$\text{s.t.} \quad \left\{ \begin{array}{l} \begin{pmatrix} \underline{\underline{A}}^{\mathrm{eq1}^{\mathrm{T}}} & \underline{\underline{A}}^{\mathrm{eq2}^{\mathrm{T}}} & \underline{\underline{A}}_\delta^{\mathrm{sos}^{\mathrm{T}}} \\ \underline{\underline{F}}_h^{\mathrm{eq1}^{\mathrm{T}}} & \underline{\underline{F}}_h^{\mathrm{eq2}^{\mathrm{T}}} & \underline{0}^{\mathrm{T}} \end{pmatrix} \begin{pmatrix} \underline{u}^1 \\ \underline{u}^2 \\ \underline{y} \end{pmatrix} = \begin{pmatrix} \underline{0} \\ 1 \end{pmatrix} \left\} \begin{array}{c} 9\times E \\ 1 \end{array} \right. \\ \underline{u}^1, \underline{u}^2 \text{ free},\ \underline{y} \in \mathscr{K} \end{array} \right. \tag{53}$$

where $\underline{u}^1 \in \Re^{m_1}$ and $\underline{u}^2 \in \Re^{m_2}$ and $\underline{y} \in \Re^{m_3}$. The interpretation of these equations and variables will become more transparent when the upper bound problem is described below.

4.2. The upper bound problem

4.2.1. Purely kinematic spaces. To obtain an upper bound, $\lambda_h^{*\mathrm{UB}}$, the use of purely kinematic spaces is required. In this case, and unlike the lower bound problem, different finite element spaces are used to solve the plane stress and the plane strain models.

In *plane stress*, a continuous, piecewise linear, function element space, $Y_{h,1}^{\mathrm{UB}} \subset Y$, is chosen to approximate the velocity field: $\mathbf{u} \approx \mathbf{u}_h \in Y_{h,1}^{\mathrm{UB}}$ as

$$u_i(x_1, x_2) = \sum_{A=1}^{N} u_i^A \phi_A(x_1, x_2), \quad i = 1:2 \tag{54}$$

$$u_i^e(x_1, x_2) = \sum_{a=1}^{3} u_i^{a,e} N_a^e(x_1, x_2), \quad i = 1:2 \tag{55}$$

DOI: 10.1002/nme

where $\phi_A(\mathbf{x})$ are the linear, hat shape functions and $N_a^e(\mathbf{x})$ their restriction to the element e. The index $A=1:N$ represents a global numbering of the N nodes of the mesh, whereas a is its local counterpart. From (54), it should be clear that $Y_{h,1}^{\mathrm{UB}}$ introduces $2\times N$ degrees of freedom in the displacement rates. Note that there exists a mapping that relates the superscript a,e with the global numbering A. Consequently, for a given global node, A, shared by two adjacent elements, e and e', the following holds: $u_i^A=u_i^{a,e}=u_i^{b,e'}$ for a certain value of a and b from 1 to 3. For the stresses, a piecewise constant function element space, $X_{h,1}^{\mathrm{UB}}\subset X$, is chosen. The global interpolation is given by

$$\sigma_i(x_1,x_2)=\sum_{e=1}^{E}\sigma_i^e\psi_e(x_1,x_2),\quad i=1:3 \tag{56}$$

where $\psi_e(\mathbf{x})=1,\ \forall\mathbf{x}\in\Omega^e$ and 0 otherwise.

In *plane strain*, the velocity field is approximated by a discontinuous, piecewise linear, finite element space, $Y_{h,2}^{\mathrm{UB}}\subset Y$, so that $\mathbf{u}\approx\mathbf{u}_h\in Y_{h,2}^{\mathrm{UB}}$. Therefore, the displacements $\mathbf{u}_h$ are linear and continuous within an element, but discontinuities are allowed between elements. Clearly, $Y_{h,2}^{\mathrm{UB}}$ introduces $6\times E$ velocity unknowns. The choice of this interpolation is due to the necessity to introduce enough degrees of freedom so that incompressibility can be satisfied [31]. The incompressibility condition, $\nabla\cdot\mathbf{u}_h=0$, is a requirement in plane strain because the von Mises yield set, B_2, is unbounded (hydrostatic pressure has no influence on yield), which means that only incompressible flow fields have finite energy dissipation rate. In this case, it is convenient to work with a local interpolation, which is given by

$$u_i^e(x_1,x_2)=\sum_{a=1}^{3}u_i^{a,e}N_a^e(x_1,x_2),\quad i=1:2 \tag{57}$$

Note that (57) looks exactly like (55), but here each global node, A, does not have a single velocity vector associated with it. Instead, it has as many velocity vectors as elements converging at the node.

For the stresses, a piecewise constant interpolation is used, as in the plane stress case:

$$\sigma_i(x_1,x_2)=\sum_{e=1}^{E}\sigma_i^e\psi_e(x_1,x_2),\quad i=1:3 \tag{58}$$

However, an additional internal traction field, denoted by $\mathbf{t}_h$, also needs to be introduced. This field represents the tractions acting on the internal edges of the mesh and is independently parameterized from the stress field, $\boldsymbol{\sigma}_h$. These tractions are allowed to vary linearly along the internal edges as

$$t_{i'}^{\xi_e^{e'}}(s)=\sum_{\alpha=1}^{2}t_{i'}^{\alpha,\xi_e^{e'}}N_\alpha^{\xi_e^{e'}}(s),\quad i'=1':2' \tag{59}$$

For convenience, the tractions will be expressed in a local coordinate system, $x_{1'},x_{2'}$, different for each edge. To be consistent, let e denote the element on the left of the edge $\xi_e^{e'}$ and e' the element on the right, when one moves from the node $\alpha=1$ of the edge to the node $\alpha=2$. Then, $x_{1'}$ is normal to the edge and points towards e', whereas $x_{2'}$ follows the edge pointing towards the node $\alpha=2$. For simplicity, $X_{h,2}^{\mathrm{UB}}$ will denote the interpolation spaces chosen for both stresses and tractions.

DOI: 10.1002/nme

Finally, to prove the purely kinematic nature of the interpolation spaces just introduced, one needs to show that when the continuous problem (10)–(13) is discretized using $X_{h,1}^{\mathrm{UB}} \times Y_{h,1}^{\mathrm{UB}}$ (plane stress) or $X_{h,2}^{\mathrm{UB}} \times Y_{h,2}^{\mathrm{UB}}$ (plane strain), condition (22), or the less general condition (21), holds. This is shown in Reference [24].

4.2.2. Discretization process. Upper bounds are typically computed using the *kinematic principle of limit analysis* (13): $\lambda^* = \inf_{\mathbf{u}\in C} D(\mathbf{u})$. This principle is formulated in terms of velocities that are implicitly required to satisfy the flow rule. The resulting problem is highly non-linear and the approach exploits neither convexity nor duality in an explicit manner. Going back to Figure 1, the conventional approach corresponds to discretizing the kinematic principle with a flow interpolation and, then, solving exclusively the problem shown in the figure as '*discrete kinematic UB*'.

Here, a different approach has been chosen whereby the *static principle of limit analysis* (14) is discretized. Recall that, as shown in (14)–(17), the static and kinematic principles are dual each other and completely equivalent. Moreover, purely kinematic interpolation spaces are used for both stresses and velocities. Thus, the approach formulates the '*discrete static UB*', which corresponds to the upper left branch of Figure 1. Additionally, the solution process is based on a primal–dual IPM (see Section 3), thereby explicitly considering the dual form. Note that following this approach there is no need to impose any restriction on the velocity field (such as the flow rule or the normalization of the external work rate to unity), since the dual problem directly takes care of this. In summary, the approach computes an upper bound by discretizing the static principle of limit analysis by means of a purely kinematic discretization. This may not be the natural way of thinking, but mathematically it is simpler and more efficient.

Equilibrium constraint in plane stress: To discretize the weak form of equilibrium, $a(\boldsymbol{\sigma}, \mathbf{u}) = \lambda F(\mathbf{u}), \forall \mathbf{u} \in Y$, one replaces the continuum fields $\boldsymbol{\sigma}$ and $\mathbf{u}$ by the global interpolations (54) and (56), respectively. The result of the discretization is given by the following global matrix equation (see details in Appendix A.1):

$$\underline{\underline{A}}^{\mathrm{eq}} \underline{\tilde{\sigma}}_h = \lambda \underline{F}_h^{\mathrm{eq}} \tag{60}$$

where $\underline{\underline{A}}^{\mathrm{eq}}$ is a matrix of dimensions $(2 \times N, 3 \times E)$, $\underline{F}_h^{\mathrm{eq}}$ is a $2 \times N$ vector of nodal forces and $\underline{\tilde{\sigma}}_h$ is a $3 \times E$ vector of elemental stresses. Note that the rows of $\underline{\underline{A}}^{\mathrm{eq}}$ associated with the Dirichlet boundary nodes have been removed.

Equilibrium constraint in plane strain: In plane strain, discontinuities in the velocities between elements are allowed. Consequently, the definition of the internal work rate needs to be extended to consider the work occurring in the inter-element boundaries. Thus, the weak form of equilibrium reads as follows:

$$\begin{gathered} a(\boldsymbol{\sigma}, \mathbf{t}, \mathbf{u}) = \lambda F(\mathbf{u}) \quad \forall \mathbf{u} \in Y \\ \sum_{i,j=1}^{2} \int_{\Omega} \sigma_{ij} \frac{\partial u_i}{\partial x_j} \, \mathrm{d}V + \sum_{\xi_e^{e'} \in \mathscr{E}^{\mathrm{U}}} \sum_{i'=1}^{2} \int_{\xi_e^{e'}} t_{i'}^{\xi_e^{e'}} (u_{i'}^{e'} - u_{i'}^{e}) \, \mathrm{d}S \\ = \lambda \sum_{i=1}^{2} \left(\int_{\Omega} f_i u_i \, \mathrm{d}V + \int_{\Gamma^{\mathrm{N}}} g_i u_i \, \mathrm{d}S \right) \quad \forall \mathbf{u} \in Y \end{gathered} \tag{61}$$

To discretize (61), the continuum spaces $X \times Y$ are replaced by the interpolation spaces $X_{h,2}^{\mathrm{UB}} \times Y_{h,2}^{\mathrm{UB}}$ defined by (57)–(59). The details of the discretization process are given in Appendix A.2. Finally,

DOI: 10.1002/nme

the discrete version of the weak equilibrium equation (61) corresponds to the following global matrix equation:

$$\underline{\underline{\tilde{A}}}^{\mathrm{eq1}}\underline{\tilde{\sigma}}_h+\underline{\underline{\tilde{A}}}^{\mathrm{eq2}}\underline{t}_h=\lambda\underline{\tilde{F}}_h^{\mathrm{eq}} \tag{62}$$

where $\underline{\underline{\tilde{A}}}^{\mathrm{eq1}}$ is a matrix of dimensions $(6\times E, 3\times E)$, $\underline{\underline{\tilde{A}}}^{\mathrm{eq2}}$ has dimensions $(6\times E, 4\times|\mathscr{E}^{\partial}|)$, $\underline{\tilde{\sigma}}_h$ is the usual $3\times E$ vector of elemental stresses, $\underline{t}_h$ is a $4\times|\mathscr{E}^{\partial}|$ vector collecting the nodal tractions for each interior edge and $\underline{\tilde{F}}_h^{\mathrm{eq}}$ is a vector of $6\times E$ discontinuous nodal forces. Here also, the rows of $\underline{\underline{\tilde{A}}}^{\mathrm{eq1}}$ and $\underline{\underline{\tilde{A}}}^{\mathrm{eq2}}$ associated with the Dirichlet boundary nodes need to be removed.

Membership constraints. Formulation as multiple second-order cones: Discretizing the constraint $\boldsymbol{\sigma}\in B_\delta$ is straightforward, since X_h^{UB} consists of elementally constant stresses. It is therefore only necessary to impose, in each element, that the uniform stress tensor belongs to the admissible set:

$$(\sigma_1^e-\sigma_2^e)^2+(\sigma_1^e)^2+(\sigma_2^e)^2+6(\sigma_3^e)^2\leqslant 2\sigma_y^2 \quad (\delta=1,\text{ plane stress}) \tag{63}$$

$$(\sigma_1^e-\sigma_2^e)^2+4(\sigma_3^e)^2\leqslant \tfrac{4}{3}\sigma_y^2 \quad (\delta=2,\text{ plane strain}) \tag{64}$$

The above elemental constraints are exactly the same as inequalities (46) and (47) in the lower bound problem, which were imposed in each of the three nodes of the element. Here, however, only E inequalities need to be imposed. For each inequality, the vectors $(\sqrt{2}\sigma_y, \sigma_1^e, \sigma_2^e, \sqrt{6}\sigma_3^e, \sigma_1^e-\sigma_2^e)$ or $(2/\sqrt{3}\sigma_y, 2\sigma_3^e, \sigma_1^e-\sigma_2^e)$ are forced to belong to the Lorentz cone $\mathscr{L}^n$, where $n=5$ for plane stress and $n=3$ for plane strain. This can be imposed by introducing an n-tuple elemental vector $\underline{\tilde{x}}^e$ of additional variables that satisfies Equations (48). Globally, this translates into $n\times E$ equations that are given, in matrix form, by the following equation:

$$\underline{\underline{\tilde{A}}}_\delta^{\mathrm{soc}}\underline{\tilde{\sigma}}_h+\underline{\tilde{I}}_\delta\underline{\tilde{x}}_\delta^{\mathrm{soc}}=\underline{\tilde{b}}_\delta^{\mathrm{soc}} \tag{65}$$

where $\underline{\tilde{\sigma}}_h$ is the usual $3\times E$ vector of unknown elemental stresses, $\underline{\tilde{I}}_\delta$ is an $(n\times E, n\times E)$ identity matrix, $\underline{\tilde{x}}^{\mathrm{soc}}$ is a vector of $n\times E$ additional variables ordered in the same way as $\underline{\tilde{\sigma}}_h$, and $\underline{\tilde{b}}_\delta^{\mathrm{soc}}$ and $\underline{\underline{\tilde{A}}}_\delta^{\mathrm{soc}}$ are an $n\times E$ vector and a $(n\times E, 3\times E)$ block-diagonal matrix of forms (51), respectively.

For plane strain, the condition $\mathbf{t}_h\in\tilde{B}_{2,h}$ must also be imposed, where $\tilde{B}_{2,h}$ is a set that places no restriction on the components of $\mathbf{t}_h$ that are normal to the inter-element edges, but forces the parallel components to satisfy the von Mises plane strain condition, $B_{2,h}$. The idea is to let the tangential tractions be as big as possible (in absolute value) within $B_{2,h}$, which is equivalent to restrict them to a pure shear state. Note that, in the local coordinates $x_{1'}, x_{2'}$, a pure shear state makes the tangential tractions coincide with the shear stresses, σ_{12}. Clearly, $B_{2,h}\subset\tilde{B}_{2,h}$, as was required to obtain a purely kinematic discretization and, consequently, an upper bound. The motivation for this new set of admissible stresses is to guarantee that the resulting velocity field is kinematically admissible. Indeed, the fact that the normal tractions are not restricted will force the normal jump in the velocities to vanish in each internal edge.

To impose the constraint $\mathbf{t}_h\in\tilde{B}_{2,h}$, the components of $\mathbf{t}_h$ parallel to the edges must satisfy, for each edge $\xi_e^{e'}$ (denoted ξ in the next expressions), the following inequality: $(t_{2'}^{\alpha,\xi})^2\leqslant\frac{1}{3}\sigma_y^2$, for

DOI: 10.1002/nme

$\alpha=1:2$ or, equivalently, $-1/\sqrt{3}\sigma_y \leqslant t_{2'}^{\alpha,\xi} \leqslant 1/\sqrt{3}\sigma_y$. These inequalities are imposed through the following equations:

$$\begin{cases} t_{2'}^{1,\xi}+s_1=\dfrac{1}{\sqrt{3}}\sigma_y, \quad -t_{2'}^{1,\xi}+s_2=\dfrac{1}{\sqrt{3}}\sigma_y, \quad s_1\geqslant 0, s_2\geqslant 0 \ \ (\alpha=1) \\ t_{2'}^{2,\xi}+s_3=\dfrac{1}{\sqrt{3}}\sigma_y, \quad -t_{2'}^{2,\xi}+s_4=\dfrac{1}{\sqrt{3}}\sigma_y, \quad s_3\geqslant 0, s_4\geqslant 0 \ \ (\alpha=2) \end{cases} \tag{66}$$

In matrix form, (66) reads as follows:

$$\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -1 \end{pmatrix} \begin{pmatrix} t_{1'}^{1,\xi} \\ t_{2'}^{1,\xi} \\ t_{1'}^{2,\xi} \\ t_{2'}^{2,\xi} \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \end{pmatrix} = \frac{1}{\sqrt{3}}\sigma_y \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \tag{67}$$

or

$$\underline{\underline{A}}_{\xi}^{t}\underline{t}_h^{\xi}+\underline{\underline{I}}^{\xi}\underline{s}^{\xi}=\underline{b}_{\xi}^{t}, \quad \underline{s}^{\xi}\geqslant\underline{0}^{\xi} \tag{68}$$

Finally, after the assembly process, (68) results in the following global equation:

$$\underline{\underline{A}}^{t}\underline{t}_h+\underline{\underline{I}}^{t}\underline{s}=\underline{b}^{t}, \quad \underline{s}\geqslant\underline{0} \tag{69}$$

where $\underline{\underline{A}}^t$ is a block-diagonal matrix of dimensions $(4\times|\mathscr{E}^{\mathscr{O}}|, 4\times|\mathscr{E}^{\mathscr{O}}|)$, $\underline{\underline{I}}^t$ is an identity matrix of the same dimensions, $\underline{t}_h$ is the previously introduced $4\times|\mathscr{E}^{\mathscr{O}}|$ traction vector and s is a $4\times|\mathscr{E}^{\mathscr{O}}|$ vector of slack variables.

4.2.3. The upper bound problem as an SOCP. Considering Equations (60) and (65) for plane stress, and (62), (65) and (69) for plane strain the discretization of the upper bound problem (14) is obtained. For *plane stress* this gives

$$\lambda_h^{*\mathrm{UB}} \equiv \max \quad \lambda$$

$$\text{s.t.} \quad \begin{cases} \overbrace{\begin{pmatrix} \underline{\underline{A}}^{\mathrm{eq}} & \vdots & -\underline{F}_h^{\mathrm{eq}} & \vdots & \underline{0} \\ \underline{\underline{\tilde{A}}}_1^{\mathrm{soc}} & \vdots & \underline{0} & \vdots & \underline{\underline{\tilde{I}}}_1 \end{pmatrix}}^{3\times E \quad + \quad 1 \quad + \quad 5\times E} \begin{pmatrix} \underline{\tilde{\sigma}}_h \\ \lambda \\ \underline{\tilde{x}}_1^{\mathrm{soc}} \end{pmatrix} = \begin{pmatrix} \underline{0} \\ \underline{\tilde{b}}_1^{\mathrm{soc}} \end{pmatrix} \begin{matrix} \} \ m_1=2\times(N-N^{\mathscr{D}}) \\ \} \ m_2=5\times E \end{matrix} \\ \underline{\tilde{\sigma}}_h \text{ free}, \ \lambda \text{ free}, \ \underline{\tilde{x}}_1^{\mathrm{soc}} \in \tilde{\mathscr{K}} \end{cases} \tag{70}$$

DOI: 10.1002/nme

whereas for *plane strain*

$$\lambda_h^{*\mathrm{UB}} \equiv \max \quad \lambda$$

$$\text{s.t.} \quad \begin{cases} \overbrace{\begin{pmatrix} \underline{\underline{\tilde{A}}}^{\mathrm{eq1}} & \vdots & \underline{\underline{\tilde{A}}}^{\mathrm{eq2}} & \vdots & -\underline{\tilde{F}}_h^{\mathrm{eq}} & \vdots & \underline{\underline{0}} & \vdots & \underline{\underline{0}} \\ \underline{\underline{\tilde{A}}}_2^{\mathrm{soc}} & \vdots & \underline{\underline{0}} & \vdots & \underline{0} & \vdots & \underline{\underline{0}} & \vdots & \underline{\underline{\tilde{I}}}_2 \\ \underline{\underline{0}} & \vdots & \underline{\underline{A}}^t & \vdots & \underline{0} & \vdots & \underline{\underline{I}}^t & \vdots & \underline{\underline{0}} \end{pmatrix}}^{3\times E \;+\; 4\times|\mathscr{E}^{\mathscr{O}}| \;+\; 1 \;+\; 4\times|\mathscr{E}^{\mathscr{O}}| \;+\; 3\times E} \begin{pmatrix} \underline{\tilde{\sigma}}_h \\ \underline{t}_h \\ \lambda \\ \underline{s} \\ \underline{\tilde{x}}_2^{\mathrm{soc}} \end{pmatrix} \\ = \begin{pmatrix} \underline{0} \\ \underline{\tilde{b}}_2^{\mathrm{soc}} \\ \underline{b}^t \end{pmatrix} \end{cases} \begin{matrix} r_1 = 6\times E - 4\times|\mathscr{E}^{\mathscr{D}}| \\ r_2 = 3\times E \\ r_3 = 4\times|\mathscr{E}^{\mathscr{O}}| \end{matrix} \quad \underline{\tilde{\sigma}}_h \text{ free}, \underline{t}_h \text{ free}, \lambda \text{ free}, \; \underline{s} \geqslant \underline{0}, \; \underline{\tilde{x}}_2^{\mathrm{soc}} \in \tilde{\mathscr{K}} \tag{71}$$

where $\tilde{\mathscr{K}} = \overbrace{\mathscr{L}^n \times \cdots \times \mathscr{L}^n}^{E}$ ($n=5$ for plane stress and $n=3$ for plane strain) and $N^{\mathscr{D}}$ is the total number of Dirichlet nodes. Note that the above problems present the canonical form of a CP. Thanks to strong duality, $\lambda_h^{*\mathrm{UB}}$ is also the solution to the problems dual to (70) and (71). For *plane stress*, the dual to (70) is

$$\lambda_h^{*\mathrm{UB}} \equiv \min \quad \underset{\sim}{\tilde{b}}_1^{\mathrm{soc}^{\mathrm{T}}} \underset{\sim}{\tilde{y}}$$

$$\text{s.t.} \quad \begin{cases} \begin{pmatrix} \underline{\underline{A}}^{\mathrm{eq}^{\mathrm{T}}} & \underline{\underline{\tilde{A}}}_1^{\mathrm{soc}^{\mathrm{T}}} \\ -\underline{F}_h^{\mathrm{eq}^{\mathrm{T}}} & \underline{0}^{\mathrm{T}} \end{pmatrix} \begin{pmatrix} \underline{u} \\ \underline{\tilde{y}} \end{pmatrix} = \begin{pmatrix} \underline{0} \\ 1 \end{pmatrix} \end{cases} \begin{matrix} 3\times E \\ 1 \end{matrix} \\ \underline{u} \text{ free}, \; \underline{\tilde{y}} \in \tilde{\mathscr{K}} \end{cases} \tag{72}$$

where $\underline{u} \in \Re^{m_1}$, $\underline{\tilde{y}} \in \Re^{m_2}$ and for *plane strain*, the problem dual to (71) is

$$\lambda_h^{*\mathrm{UB}} \equiv \min \quad \underset{\sim}{\tilde{b}}_2^{\mathrm{soc}^{\mathrm{T}}} \underline{y}_1 + \underline{b}^{t^{\mathrm{T}}} \underline{y}_2$$

$$\text{s.t.} \quad \begin{cases} \begin{pmatrix} \underline{\underline{\tilde{A}}}^{\mathrm{eq1}^{\mathrm{T}}} & \underline{\underline{\tilde{A}}}_2^{\mathrm{soc}^{\mathrm{T}}} & \underline{\underline{0}} \\ \underline{\underline{\tilde{A}}}^{\mathrm{eq2}^{\mathrm{T}}} & \underline{\underline{0}} & \underline{\underline{A}}^{t^{\mathrm{T}}} \\ -\underline{F}_h^{\mathrm{eq}^{\mathrm{T}}} & \underline{0}^{\mathrm{T}} & \underline{0}^{\mathrm{T}} \end{pmatrix} \begin{pmatrix} \underline{\tilde{u}} \\ \underline{y}_1 \\ \underline{y}_2 \end{pmatrix} = \begin{pmatrix} \underline{0} \\ \underline{0} \\ 1 \end{pmatrix} \end{cases} \begin{matrix} 3\times E \\ 4\times|\mathscr{E}^{\mathscr{O}}| \\ 1 \end{matrix} \\ \underline{\tilde{u}} \text{ free}, \; \underline{y}_2 \geqslant \underline{0}, \; \underline{y}_1 \in \tilde{\mathscr{K}} \end{cases} \tag{73}$$

where $\underline{\tilde{u}} \in \Re^{r_1}$, $\underline{y}_1 \in \Re^{r_2}$ and $\underline{y}_2 \in \Re^{r_3}$.

DOI: 10.1002/nme

The equations and variables in the above dual problems have useful interpretations. For instance, in problem (72), the vector $\underline{u}$ are the nodal velocities (this vector is equivalent to the variables $\underline{u}^1$ and $\underline{u}^2$ in the lower bound dual problem (53), which represented elemental and inter-element velocities, respectively), whereas the vector $\underline{y}$ collects the plastic multipliers associated with the flow rule that govern the deformation of each element. Taking this into account and the fact that the vector $\underset{\sim}{\underline{b}}_1^{\text{soc}}$ incorporates the yield stress, σ_y, the cost function can be seen as an approximation to the total energy dissipation rate, $D(\mathbf{u})$. Finally, the first matrix equation approximately imposes the flow rule, whereas the second equation equates the approximated external work rate to unity.

An analogous interpretation can be performed for problem (73). The vector $\underline{\tilde{u}}$ collects the nodal velocities associated with each element, which results in a global discontinuous velocity field. The vector $\underline{y}_1$ gives the non-negative plastic multipliers involved in the flow rule of each element. Similarly, $\underline{y}_2$ collects non-negative scalars that can be seen as plastic multipliers of an alternative flow rule for the tractions $\underline{t}_h$. Now, as previously, the cost function is an approximation to the energy dissipation rate $D(\mathbf{u})$. Note that, in this case, there is an additional term, $\underline{b}'^{\mathrm{T}}\underline{y}_2$ that corresponds to the rate of energy dissipated at the inter-element boundaries. Clearly, this is due to the discontinuities in the flow. The first equation is the classical flow rule relating the deformation with the gradient of the yield function in each element. The second equation can be seen as another flow rule, but in this case for the jump in the velocities in each internal edge $\xi_e^{e'}$. More specifically, it forces two conditions: (1) it gives the magnitude of the jump in the velocities in the tangential direction, i.e. in the direction of the edge; (2) it imposes the jump in the velocities to be zero in the direction normal to the edge. Thus, if two adjacent elements have different velocities, this last condition forces their relative displacement to be tangential, therefore resulting in a velocity field that is *kinematically admissible*. Finally, the third equation makes the external work rate equal to 1.

Similar to the lower bound evaluation problem, it is possible to reduce the number of unknowns by appropriate changes in problem variables. Details are given in Reference [24].

4.3. Solution to bound problems

The above problems present the canonical form of a CP and, in particular, their level of complexity is that of a second-order cone program (SOCP). This is important mainly for two reasons. First, it allows for the use of state-of-the-art primal–dual interior point algorithms that have been particularly developed for SOCP and that guarantee global convergence and efficiency in the solution process. Moreover, with these algorithms, not only the above discrete static bound problems but also their duals are solved, which are discrete versions of the kinematic principle. In this way, collapse fields for the stresses and velocities are simultaneously obtained. Second, the bound problems can be solved using any conic programming optimization package. In particular, the public domain conic solvers SeDuMi [32] and SDPT3 [33] are the ones used here.

5. MESH ADAPTIVITY

The objective of mesh adaptivity is to efficiently refine the computational mesh, $\mathscr{T}_h$, by only dividing the elements that make the largest contribution to the overall numerical error. In the problem considered here, the numerical error is measured by the *bound gap*, $\Delta_h = \lambda_h^{*\mathrm{UB}} - \lambda_h^{*\mathrm{LB}}$, for which it is possible to identify the contribution of each element in the mesh. Indeed, for both plane stress and plane strain cases, this elemental contribution, named *elemental bound gap*, is

DOI: 10.1002/nme

given by

$$\Delta_h^e = \underbrace{\int_{\Omega^e} \sigma_y \varepsilon_{\mathrm{eq}}(\mathbf{u}_{\mathrm{UB}}^e)\,\mathrm{d}V}_{D^e(\mathbf{u}_{\mathrm{UB}}^e)} - \underbrace{\left(\int_{\Omega^e} (-\nabla\cdot\boldsymbol{\sigma}_{\mathrm{LB}}^e)\cdot\mathbf{u}_{\mathrm{UB}}^e\,\mathrm{d}V + \int_{\partial\Omega^e} (\mathbf{n}^{\xi_e}\cdot\boldsymbol{\sigma}_{\mathrm{LB}}^e)\cdot\mathbf{u}_{\mathrm{UB}}^e\,\mathrm{d}S\right)}_{F^e(\mathbf{u}_{\mathrm{UB}}^e)} \tag{74}$$

In (74), $\boldsymbol{\sigma}_{\mathrm{LB}}^e$ is the linear elemental stress tensor computed in the lower bound problem (52) and $\mathbf{u}_{\mathrm{UB}}^e$ are the linear velocities, restricted to the element Ω^e, obtained when the upper bound problem (70) or (71) is solved. These elemental stress and velocity fields are illustrated in Figure 2. Furthermore, $\mathbf{n}^{\xi_e}$ is the unit outward normal vector acting on the edge ξ_e of a particular element and $\varepsilon_{\mathrm{eq}}$ is a scalar known as the *effective strain rate* given by

$$\varepsilon_{\mathrm{eq}}(\mathbf{u}) = \sqrt{\tfrac{2}{3}\boldsymbol{\varepsilon}'(\mathbf{u}):\boldsymbol{\varepsilon}'(\mathbf{u})} \tag{75}$$

where $\boldsymbol{\varepsilon}'$ is the deviatoric component of the strain rate tensor. Note that Δ_h^e is obtained as the difference between the total elemental energy dissipation rate, $D^e(\mathbf{u}_{\mathrm{UB}}^e)$, and the elemental external work rate, $F^e(\mathbf{u}_{\mathrm{UB}}^e)$, in both cases for the upper bound elemental velocity, $\mathbf{u}_{\mathrm{UB}}^e$.

The elemental gap, Δ_h^e, satisfies the following two important properties:

1. It is always positive, i.e. $\Delta_h^e \geqslant 0, \forall e \in \mathscr{T}_h$.
2. Its sum over all the elements equals the total bound gap, at least as the mesh size tends to zero. In particular, for *plane stress*, the property holds strictly regardless of the mesh size, i.e. $\sum_{e\in\mathscr{T}_h} \Delta_h^e = \Delta_h$. On the other hand, for *plane strain*, the property holds only in the limit as $h \to 0$. In this case, the total bound gap can be decomposed into two positive terms, namely, $\Delta_h = \Delta_h^{\mathscr{O}} + \bar{\Delta}_h$, where $\Delta_h^{\mathscr{O}}$ comes from the continuum (interior of the elements) and $\bar{\Delta}_h$ comes from the inter-element boundaries. With the previous definition of Δ_h^e, one can show that the sum of all the elemental bound gaps adds up to the total contribution from the continuum, i.e. $\sum_{e\in\mathscr{T}_h} \Delta_h^e = \Delta_h^{\mathscr{O}}$. Finally, $\bar{\Delta}_h$ is found, in practice, to converge to zero as the mesh is refined, i.e. $\bar{\Delta}_h \to 0$ when $h \to 0$.

The above properties make Δ_h^e an effective indicator of the elemental contribution to the numerical error. Consequently, the strategy used here for mesh adaptivity consists of refining only the elements with higher Δ_h^e, that is, those elements for which $\Delta_h^e > \alpha \Delta_{h\max}^e$. Through numerical trial and error, a value of $\alpha = 0.005$ has emerged as a reasonable choice. *Regular refinement* is used whereby the element to be refined is divided into four triangles by joining the midpoints of the sides. The resulting non-conformity is overcome by bisection of the adjacent triangles.

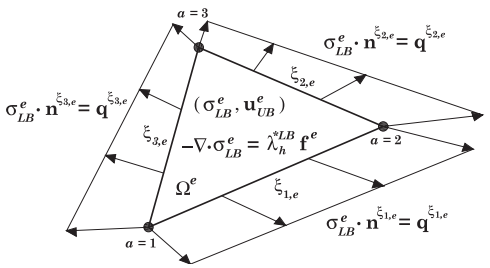


Figure 2. Notation for the elemental bound gap Δ_h^e.

DOI: 10.1002/nme

Proof of the above properties is given in the following sections.

5.1. Positiveness of Δ_h^e

Consider the individual element and the loading shown in Figure 2 as an isolated limit analysis problem. Recall that, in the lower bound problem, the stresses $\boldsymbol{\sigma}_{\mathrm{LB}}$ are in equilibrium over the continuum and belong to the yield set $B_\delta, \delta = 1{:}2$, at any point of the domain. Therefore, the local problem shown in Figure 2 is also in equilibrium and satisfies the von Mises yield criterion. This is enough to guarantee, in accordance with the static principle, that the lower bound for this problem, $\lambda_h^{e,\mathrm{LB}}$, is greater than or equal to 1:

$$\lambda_h^{e,\mathrm{LB}} \geqslant 1 \tag{76}$$

On the other hand, if the kinematic principle (13) is applied to the same elemental problem in Figure 2, using kinematically admissible velocity fields, $\mathbf{u}^e$, its upper bound, $\lambda_h^{e,\mathrm{UB}}$, is equal to

$$\lambda_h^{e,\mathrm{UB}} = \inf_{\mathbf{u}^e} \frac{D^e(\mathbf{u}^e)}{F^e(\mathbf{u}^e)} \tag{77}$$

In particular, choosing the continuous velocity field $\mathbf{u}_{\mathrm{UB}}^e$ that results from the global upper bound problem, it is apparent that

$$\frac{D^e(\mathbf{u}_{\mathrm{UB}}^e)}{F^e(\mathbf{u}_{\mathrm{UB}}^e)} \geqslant \lambda_h^{e,\mathrm{UB}} \tag{78}$$

Finally, combining (78) with (76) one obtains

$$\frac{D^e(\mathbf{u}_{\mathrm{UB}}^e)}{F^e(\mathbf{u}_{\mathrm{UB}}^e)} \geqslant \lambda_h^{e,\mathrm{UB}} \geqslant \lambda_h^{e,\mathrm{LB}} \geqslant 1 \implies \Delta_h^e = D^e(\mathbf{u}_{\mathrm{UB}}^e) - F^e(\mathbf{u}_{\mathrm{UB}}^e) \geqslant 0 \tag{79}$$

which proves the statement.

5.2. Sum of Δ_h^e

To make the next proof easier to follow, it is convenient to recall first the following identities for the global upper bound, $\lambda_h^{*\mathrm{UB}}$. In plane stress, for $\mathbf{u}_h \in X_{h,1}^{\mathrm{UB}}$, one had

$$\lambda_h^{*\mathrm{UB}} = \min_{F(\mathbf{u}_h)=1} D(\mathbf{u}_h) = D(\mathbf{u}_{\mathrm{UB}}) \tag{80}$$

$\mathbf{u}_{\mathrm{UB}} \in X_{h,1}^{\mathrm{UB}}$ being the optimal solution. In plane strain, the inter-element discontinuities had also to be considered in the internal work rate. Therefore, for $\mathbf{u}_h \in X_{h,2}^{\mathrm{UB}}$ and $(\boldsymbol{\sigma}_h, \mathbf{t}_h) \in Y_{h,2}^{\mathrm{UB}}$, one had

$$\begin{aligned}
\lambda_h^{*\mathrm{UB}} &= \min_{F(\mathbf{u}_h)=1} \max_{\boldsymbol{\sigma}_h \in B_{h,2}, \mathbf{t}_h \in \tilde{B}_{h,2}} a(\boldsymbol{\sigma}_h, \mathbf{t}_h, \mathbf{u}_h) \\
&= \min_{F(\mathbf{u}_h)=1} \begin{cases} D(\mathbf{u}_h) + \displaystyle\sum_{\xi_e^{e'} \in \mathscr{E}^{\mathscr{U}}} \int_{\xi_e^{e'}} \tau_{\max} |(\mathbf{u}_h^{e'} - \mathbf{u}_h^e) \cdot \boldsymbol{\tau}^{\xi_e^{e'}}| \, \mathrm{d}S & \text{if } (\mathbf{u}_h^{e'} - \mathbf{u}_h^e) \cdot \mathbf{n}^{\xi_e^{e'}} = 0 \ \ \forall \xi_e^{e'} \in \mathscr{E}^{\mathscr{U}} \\ \infty & \text{otherwise} \end{cases} \\
&= D(\mathbf{u}_{\mathrm{UB}}) + \tau_{\max} \sum_{\xi_e^{e'} \in \mathscr{E}^{\mathscr{U}}} \int_{\xi_e^{e'}} |(\mathbf{u}_{\mathrm{UB}}^{e'} - \mathbf{u}_{\mathrm{UB}}^e) \cdot \boldsymbol{\tau}^{\xi_e^{e'}}| \, \mathrm{d}S = D(\mathbf{u}_{\mathrm{UB}}) + \bar{D}(\mathbf{u}_{\mathrm{UB}})
\end{aligned} \tag{81}$$

DOI: 10.1002/nme

where $\mathbf{n}^{\xi_e^{e'}}$ and $\boldsymbol{\tau}^{\xi_e^{e'}}$ represent orthonormal vectors, following the directions $x_{1'}$ and $x_{2'}$ (normal and tangential to the edge), $\tau_{\max}=\sigma_y/\sqrt{3}$ given that the constraint $\mathbf{t}_h \in \tilde{B}_{h,2}$ implies $-1/\sqrt{3}\sigma_y \leqslant t_{2'}^{\alpha,\xi_e^{e'}} \leqslant 1/\sqrt{3}\sigma_y$, for $\alpha=1:2$ (see (66)), and $\bar{D}(\mathbf{u}_{\mathrm{UB}})$ represents an inter-element energy dissipation rate.

In order to construct the proof, plane strain will be assumed. The proof can be easily extended to plane stress by setting to zero the inter-element terms. Adding the elemental contributions, one obtains

$$\sum_{e\in\mathscr{T}_h} \Delta_h^e = \sum_{e\in\mathscr{T}_h} D^e(\mathbf{u}_{\mathrm{UB}}^e) - \sum_{e\in\mathscr{T}_h} F^e(\mathbf{u}_{\mathrm{UB}}^e) \tag{82}$$

The first term satisfies

$$\sum_{e\in\mathscr{T}_h} D^e(\mathbf{u}_{\mathrm{UB}}^e) = \sum_{e\in\mathscr{T}_h} \int_{\Omega^e} \sigma_y \varepsilon_{\mathrm{eq}}(\mathbf{u}_{\mathrm{UB}}^e) = \int_{\Omega} \sigma_y \varepsilon_{\mathrm{eq}}(\mathbf{u}_{\mathrm{UB}}) = D(\mathbf{u}_{\mathrm{UB}}) = \lambda_h^{*\mathrm{UB}} - \bar{D}(\mathbf{u}_{\mathrm{UB}}) \tag{83}$$

where the last equality follows from (81). For the second term, the following equalities hold:

$$\begin{aligned}
\sum_{e\in\mathscr{T}_h} F^e(\mathbf{u}_{\mathrm{UB}}^e) &\overset{(1)}{=} \sum_{e\in\mathscr{T}_h} \left(\int_{\Omega^e} (-\nabla\cdot\boldsymbol{\sigma}_{\mathrm{LB}}^e)\cdot\mathbf{u}_{\mathrm{UB}}^e \,\mathrm{d}V + \int_{\partial\Omega^e} (\mathbf{n}^{\xi_e}\cdot\boldsymbol{\sigma}_{\mathrm{LB}}^e)\cdot\mathbf{u}_{\mathrm{UB}}^e \,\mathrm{d}S \right) \\
&\overset{(2)}{=} \sum_{e\in\mathscr{T}_h} \int_{\Omega^e} \lambda_h^{*\mathrm{LB}} \mathbf{f}^e \cdot \mathbf{u}_{\mathrm{UB}}^e \,\mathrm{d}V + \sum_{e\in\mathscr{T}_h} \sum_{\xi_e\in\partial\Omega^e} \int_{\xi_e} \mathbf{q}_{\mathrm{LB}}^{\xi_e}\cdot\mathbf{u}_{\mathrm{UB}}^e \,\mathrm{d}S \\
&\overset{(3)}{=} \lambda_h^{*\mathrm{LB}} \sum_{e\in\mathscr{T}_h} \int_{\Omega^e} \mathbf{f}^e\cdot\mathbf{u}_{\mathrm{UB}}^e \,\mathrm{d}V + \lambda_h^{*\mathrm{LB}} \sum_{\xi_e^{\mathrm{N}}\in\mathscr{E}^{\mathscr{N}}} \int_{\xi_e^{\mathrm{N}}} \mathbf{g}^{\xi_e^{\mathrm{N}}}\cdot\mathbf{u}_{\mathrm{UB}}^e \,\mathrm{d}S \\
&\quad + \sum_{\xi_e^{e'}\in\mathscr{E}^{\mathscr{O}}} \int_{\xi_e^{e'}} \mathbf{q}_{\mathrm{LB}}^{\xi_e^{e'}}\cdot(\mathbf{u}_{\mathrm{UB}}^e - \mathbf{u}_{\mathrm{UB}}^{e'}) \,\mathrm{d}S \\
&\overset{(4)}{=} \lambda_h^{*\mathrm{LB}} \left(\int_{\Omega} \mathbf{f}\cdot\mathbf{u}_{\mathrm{UB}} \,\mathrm{d}V + \int_{\Gamma^{\mathrm{D}}} \mathbf{g}\cdot\mathbf{u}_{\mathrm{UB}} \,\mathrm{d}S \right) + \sum_{\xi_e^{e'}\in\mathscr{E}^{\mathscr{O}}} \int_{\xi_e^{e'}} \mathbf{q}_{\mathrm{LB}}^{\xi_e^{e'}}\cdot(\mathbf{u}_{\mathrm{UB}}^e - \mathbf{u}_{\mathrm{UB}}^{e'}) \,\mathrm{d}S \\
&\overset{(5)}{=} \lambda_h^{*\mathrm{LB}} + \sum_{\xi_e^{e'}\in\mathscr{E}^{\mathscr{O}}} \int_{\xi_e^{e'}} q_{2',\mathrm{LB}}^{\xi_e^{e'}} (\mathbf{u}_{\mathrm{UB}}^e - \mathbf{u}_{\mathrm{UB}}^{e'})\cdot\boldsymbol{\tau}^{\xi_e^{e'}} \,\mathrm{d}S
\end{aligned} \tag{84}$$

Equalities (1) and (4) are trivial. Equality (2) follows from (31). In equality (3) the integration is performed over all the edges ξ_e of all the elements. These edges can be either interior edges, $\xi_e^{e'}$, or boundary edges, ξ_e^{D} or ξ_e^{N}. For each interior edge, two integrals are computed, one for element e and another one for element e'. The inter-element stresses, $\mathbf{q}_{\mathrm{LB}}^{\xi_e^{e'}}$, are of equal magnitude and opposite sign (see second constraint in problem (31)), which explains the third term in the equality (recall that in plane strain the displacements are discontinuous). Moreover, the integrals over the Dirichlet edges, ξ_e^{D}, vanish because $\mathbf{u}_{\mathrm{UB}}$ is equal to zero. For the Neumann boundaries ξ_e^{N}, note that $\mathbf{q}_{\mathrm{LB}}^{\xi_e^{\mathrm{N}}} = \lambda_h^{*\mathrm{LB}} \mathbf{g}^{\xi_e^{\mathrm{N}}}$ (see third constraint in problem (31)). Finally, equality (5) holds because $F(\mathbf{u}_{\mathrm{UB}})=1$ and $(\mathbf{u}_{\mathrm{UB}}^e - \mathbf{u}_{\mathrm{UB}}^{e'})\cdot\mathbf{n}^{\xi_e^{e'}} = 0$ ($\mathbf{u}_{\mathrm{UB}}$ is forced to be kinematically admissible).

DOI: 10.1002/nme

From (83) and (84), it is possible to conclude that

$$\sum_{e\in\mathscr{T}_h}\Delta_h^e=(\lambda_h^{*\mathrm{UB}}-\lambda_h^{*\mathrm{LB}})-\underbrace{\left(\bar{D}(\mathbf{u}_{\mathrm{UB}})+\sum_{\xi_e^{e'}\in\mathscr{E}^{\mathscr{O}}}\int_{\xi_e^{e'}}q_{2',\mathrm{LB}}^{\xi_e^{e'}}(\mathbf{u}_{\mathrm{UB}}^e-\mathbf{u}_{\mathrm{UB}}^{e'})\cdot\boldsymbol{\tau}^{\xi_e^{e'}}\,\mathrm{d}S\right)}_{\bar{\Delta}_h}$$
$$\sum_{e\in\mathscr{T}_h}\Delta_h^e=\Delta_h-\bar{\Delta}_h \tag{85}$$

Observe that $\bar{\Delta}_h\geqslant 0$, since $q_{2',\mathrm{LB}}^{\xi_e^{e'}}\leqslant\tau_{\max}$. Now, if $\Delta_h^{\mathscr{O}}$ denotes the sum of the elemental bound gaps, it follows from (85) that $\Delta_h=\Delta_h^{\mathscr{O}}+\bar{\Delta}_h$, where $\Delta_h^{\mathscr{O}}=\sum_{e\in\mathscr{T}_h}\Delta_h^e\geqslant 0$. Additionally, all the computations performed show that, in practice, $\bar{\Delta}_h\to 0$ as $h\to 0$. This concludes the proof for the plane strain case.

Finally, in plane stress the velocity field is continuous, which implies that $\mathbf{u}_{\mathrm{UB}}^e-\mathbf{u}_{\mathrm{UB}}^{e'}|_{\xi_e^{e'}}=0,\ \forall\xi_e^{e'}\in\mathscr{E}^{\mathscr{O}}$. Consequently, $\bar{\Delta}_h=0$ and $\sum_{e\in\mathscr{T}_h}\Delta_h^e=\Delta_h$.

6. NUMERICAL EXAMPLES

To assess the efficiency and accuracy of the method presented in the previous sections, three two-dimensional examples are solved in plane stress and plane strain. For simplicity, and to make it possible to compare with known results, the volume forces are set to zero and the yield stress is chosen to be $\sigma_y=\sqrt{3}$. In the last section, the size of the problems and the computational time involved in the solution process are indicated.

6.1. Asymmetrical cantilever in plane stress

In this example, an end-loaded wide tapered cantilever, for which an analytical solution does not seem to be known, is studied for the plane stress model. The geometry and load distribution are illustrated in Figure 3. The following sections show, first, the results obtained when the computational meshes are refined uniformly and, second, the results when mesh adaptivity is considered. Finally, the improvements in the mesh adaptive strategy are evaluated.

6.1.1. Uniform meshing. Table I summarizes the main results for five different uniform meshes that are obtained by refining the initial mesh. In each refinement, a triangle is divided into four smaller triangles. The lower and upper bound relative errors in the table are computed as follows: $(\lambda_h^{*\mathrm{UB}}-\lambda^*)/\lambda^*$ and $(\lambda^*-\lambda_h^{*\mathrm{LB}})/\lambda^*$, where λ^* is the unknown exact solution. In this case, λ^* has been assumed to be $\lambda^*=0.68504$, which corresponds to the limit of convergence when very fine meshes are used.

Figure 4 illustrates the deformed geometry for different meshes. Note that only the right-hand side of the cantilever flows. Figure 5 shows graphically the bounds obtained for each refinement and, also, the rate of convergence for the bound errors and the bound gap. After four refinements,

DOI: 10.1002/nme

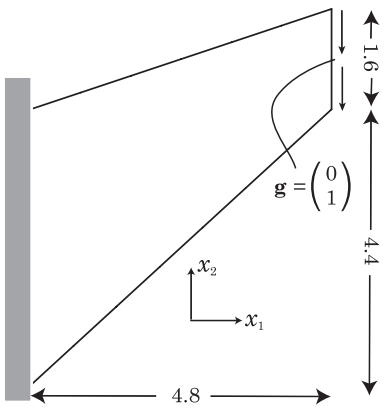


Figure 3. Geometry and loads for the cantilever problem in plane stress.

Table I. Results for the cantilever problem in plane stress using uniform meshing.

Number of refinements	Number of elements	Lower bound $\lambda_h^{*\mathrm{LB}}$	Upper bound $\lambda_h^{*\mathrm{UB}}$	Bound gap Δ_h	Lower bound error (%)	Upper bound error (%)
Uniform meshing						
0	34	0.52186	0.75759	0.23573	23.821	10.591
1	136	0.65432	0.71936	0.06503	4.484	5.010
2	544	0.68079	0.69704	0.01624	0.620	1.752
3	2176	0.68349	0.68983	0.00634	0.226	0.699
4	8704	0.68440	0.68662	0.00223	0.093	0.231

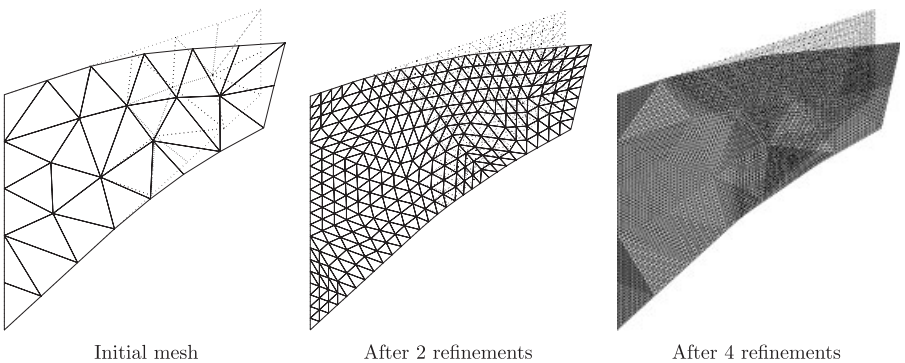
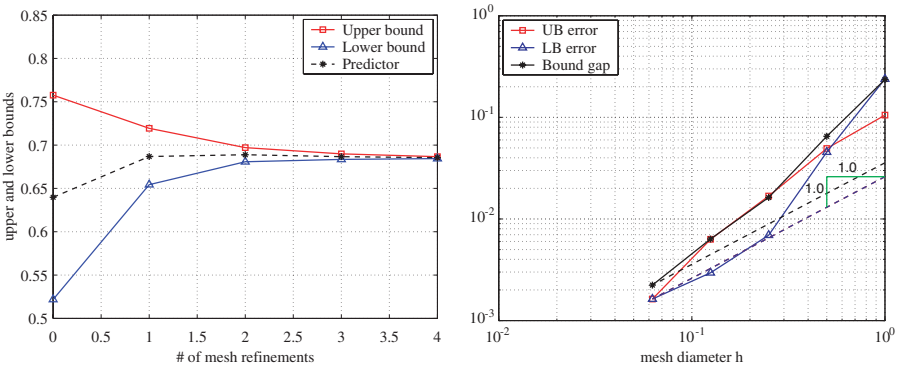


Figure 4. Cantilever problem—deformed geometry.

with 8704 elements, the maximum relative error is only 0.231%. Moreover, the upper bound error presents a rate of convergence clearly higher than linear. On the other hand, the lower bound error converges linearly in the asymptotic range, despite the initial superlinear convergence.

DOI: 10.1002/nme



Upper and lower bounds Bound gap rate of convergence

Figure 5. Cantilever problem—convergence.

Table II. Results for the cantilever problem in plane stress using adaptive meshing.

Number of refinements	Number of elements	Lower bound $\lambda_h^{*\mathrm{LB}}$	Upper bound $\lambda_h^{*\mathrm{UB}}$	Bound gap Δ_h	Lower bound error (%)	Upper bound error (%)
Adaptive meshing						
0	34	0.52186	0.75759	0.23573	23.821	10.591
1	90	0.65782	0.71951	0.06169	3.973	5.032
2	300	0.68079	0.69704	0.01625	0.620	1.752
3	882	0.68349	0.68989	0.00640	0.226	0.708
4	2450	0.68440	0.68667	0.00227	0.093	0.238
5	5506	0.68459	0.68549	0.00090	0.066	0.066

6.1.2. Adaptive meshing. Table II summarizes the results obtained when the adaptive mesh procedure introduced in Section 5 is used. In this case, five refinements of the original mesh have been performed. Note that, with only 2450 elements, the maximum relative error is 0.238%, practically the same as the one obtained previously with 8704 elements. Moreover, with 5506 elements, the error reduces to 0.066%, which can be considered negligible in practice.

The deformed geometry is shown in Figure 6 for three different adaptive meshes. Moreover, Figure 7 shows the contribution of each element to the total bound gap. In Figure 8, the elements with higher contribution, which are the ones selected to be refined, are identified and filled in blue. As can be observed in Figure 6, the adaptive meshing strategy seems to capture very well the collapse mechanism, identifying four 'slip-lines' that converge in a central plastic hinge, thus dividing the cantilever into four regions (this mechanism is predicted by the 'slip-line' theory in the symmetric cantilevers analysed in [34]).

Figure 9 plots the upper and lower bounds computed for each mesh refinement. Note the accuracy obtained. Finally, Figure 10 compares the performance of the adaptive meshing strategy with the uniform meshing. Clearly, the adaptive meshing outperforms the uniform refinement.

6.1.3. Comparison of the elemental bound gap adaptive meshing strategy with an alternative, deformation-based strategy. In [13], two alternative strategies for adaptive meshing are presented.

DOI: 10.1002/nme

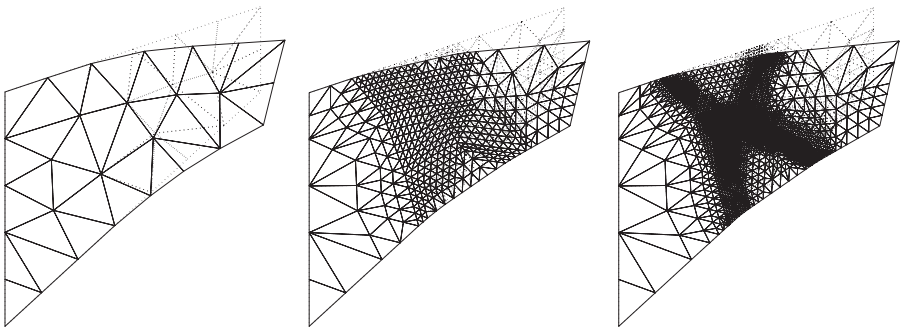


Figure 6. Cantilever problem—deformed geometry using adaptive meshing.

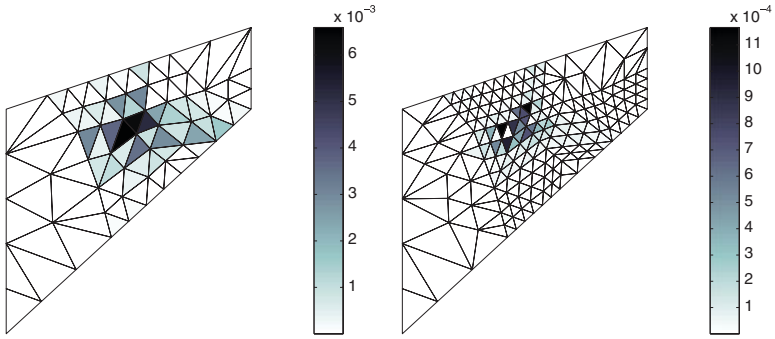


Figure 7. Cantilever problem—elemental contribution to the bound gap Δ_h^e.

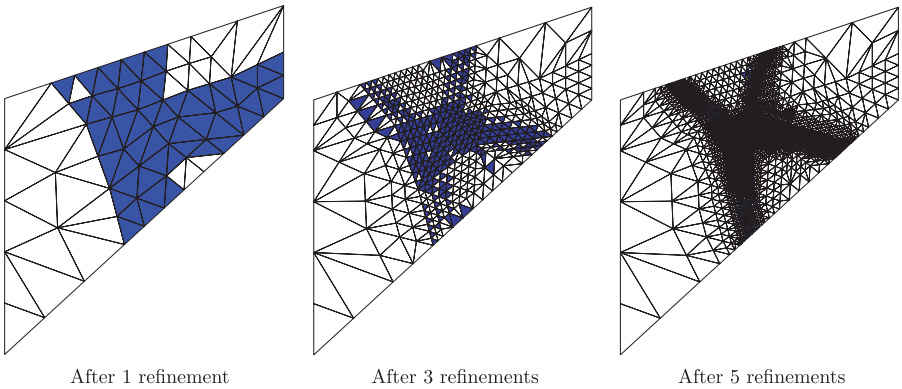


Figure 8. Cantilever problem—adaptive meshing strategy: elements to be refined.

DOI: 10.1002/nme

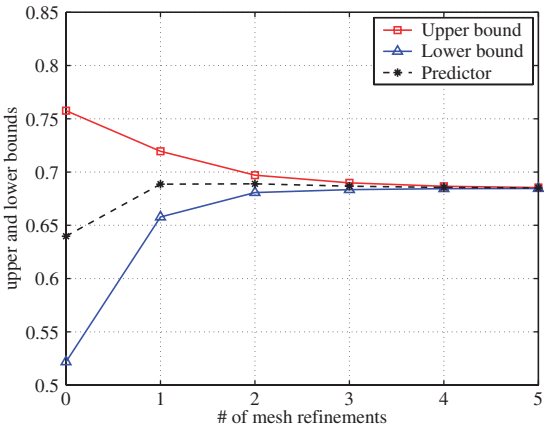


Figure 9. Cantilever problem—bounds using adaptive meshing.

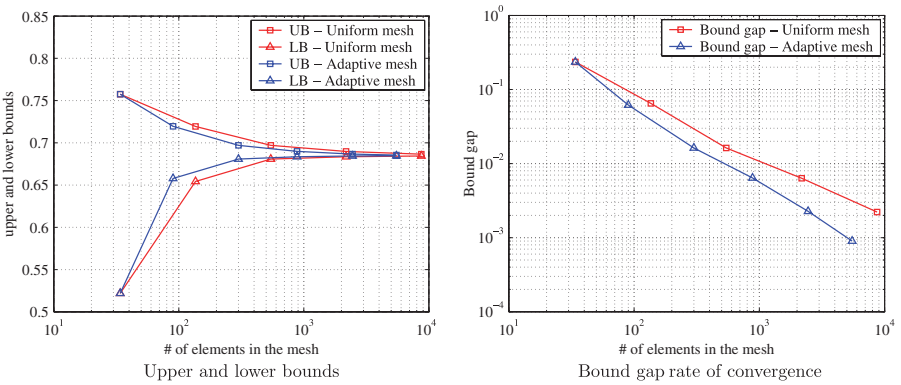


Figure 10. Cantilever problem—comparison of adaptive meshing with uniform meshing.

The first one consists of refining only the elements whose deformation is greater than a certain threshold. In the second strategy, a triangle is refined only if a certain number of its vertices are 'sufficiently' close to the yield surface.

As mentioned in [13], the non-uniqueness of the collapse fields hampers the applicability of the above two conditions. Furthermore, the lack of a local error measure as a refinement criterion also undermines the optimality of the above strategies. On top of these two distinct strategies, two different refinement procedures are tried: the *regular refinement* (the refinement technique used in this article) and the *refinement by bisection of longest edge* [35]. The conclusion in [13] is that the strain strategy together with the regular refinement is, in practice, the best combination. Consequently, this is the adaptive meshing combination that is going to be compared with the elemental bound gap strategy presented in Section 5.

DOI: 10.1002/nme

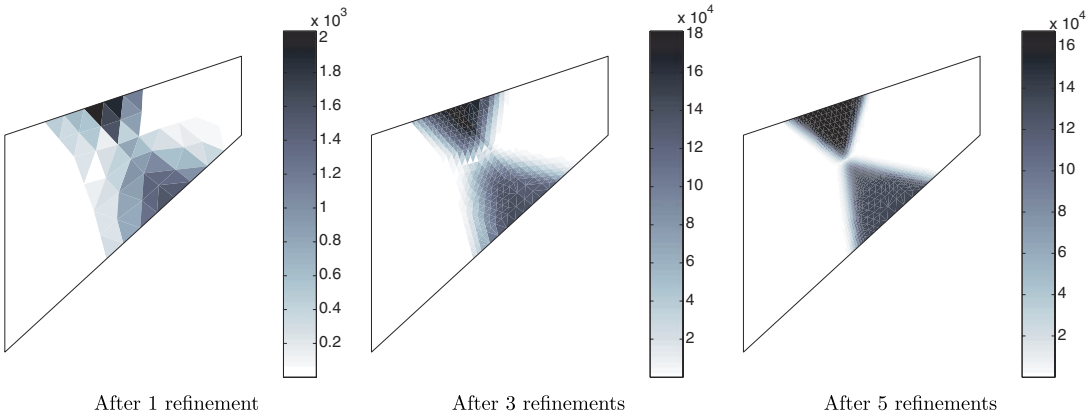


Figure 11. Cantilever problem—elemental measure of deformation $\|\boldsymbol{\varepsilon}_h^e\|$.

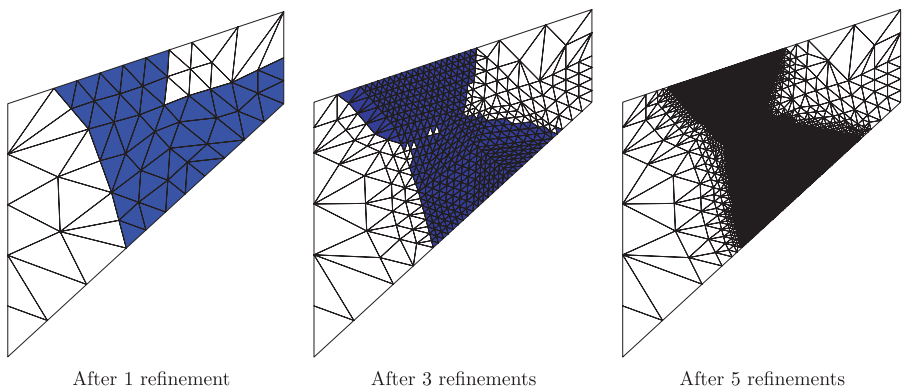


Figure 12. Cantilever problem—alternative adaptive meshing strategy: elements to be refined.

More specifically, the strain strategy in [13] refines an element if the following inequality is satisfied:

$$\|\boldsymbol{\varepsilon}_h^e\| = \sqrt{(\varepsilon_{11}^e)^2 + (\varepsilon_{22}^e)^2 + (\varepsilon_{12}^e)^2} > \delta \tag{86}$$

where $\boldsymbol{\varepsilon}_h^e$ is the strain tensor for element Ω^e and δ is a given threshold that, in this case, has been chosen to be $\delta = 0.005\|\boldsymbol{\varepsilon}_h^e\|_{\max}$.[‡] Figure 11 shows, for different adaptive meshes, the elemental deformation measure $\|\boldsymbol{\varepsilon}_h^e\|$, upon which one decides the elements to be refined. Note the difference between this figure and the equivalent one for the elemental bound gap strategy (Figure 7). As a result, the refined meshes are clearly different, as one can observe by comparing Figure 12 with Figure 8. It seems clear that for this example and refinement threshold the deformation-based

[‡]Similarly, in the adaptive meshing proposed in this article, the threshold chosen to refine an element is: $\Delta_h^e > 0.005\Delta_{h\max}^e$.

DOI: 10.1002/nme

Table III. Results for the cantilever problem in plane stress using an alternative, deformation based, adaptive meshing strategy.

Number of refinements	Number of elements	Lower bound $\lambda_h^{*\mathrm{LB}}$	Upper bound $\lambda_h^{*\mathrm{UB}}$	Bound gap Δ_h	Lower bound error (%)	Upper bound error (%)
Alternative adaptive meshing						
0	34	0.52186	0.75759	0.23573	23.820	10.591
1	90	0.65782	0.71951	0.06170	3.973	5.032
2	304	0.68079	0.69704	0.01625	0.621	1.751
3	950	0.68349	0.68982	0.00634	0.227	0.698
4	3265	0.68440	0.68662	0.00223	0.094	0.231
5	11 222	0.68457	0.68540	0.00083	0.069	0.053

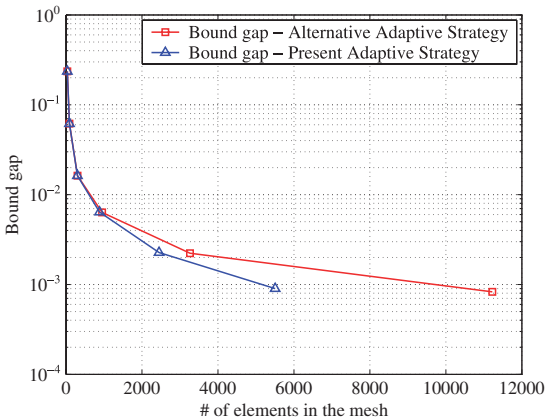


Figure 13. Cantilever problem—bound gap rate of convergence for different adaptive meshing strategies.

meshing strategy is not able to capture the slip-lines in the collapse mechanism (as opposed to the elemental bound gap strategy), this way incurring in an unnecessary overrefining.

The numerical results are summarized in Table III. Additionally, Figure 13 illustrates the improved efficiency of the elemental bound gap strategy since, in all cases, fewer elements are required in the mesh to achieve a given bound gap.

6.2. Slotted square block in plane stress

This example shows a square block with external thin symmetrical cuts, subject to a uniform tension in plane stress. The depth of the cut has been chosen to be $\frac{1}{6}$ of the total height of the block. Figure 14 illustrates the problem. As in the previous example, the results corresponding to the uniform meshing approach are presented before those obtained with the proposed adaptive meshing strategy.

6.2.1. Uniform meshing. Table IV shows the bounds computed for six different uniform meshes. Note that, in the last refinement, a very fine mesh consisting of 18 432 elements is used. The magnitude of these problems proves the capacity of the method to solve large, complex limit

DOI: 10.1002/nme

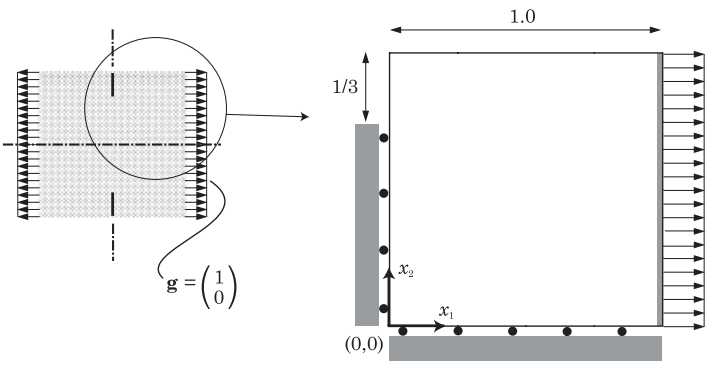


Figure 14. Geometry and loads for the slotted block problem in plane stress.

Table IV. Results for the slotted block problem in plane stress using uniform meshing.

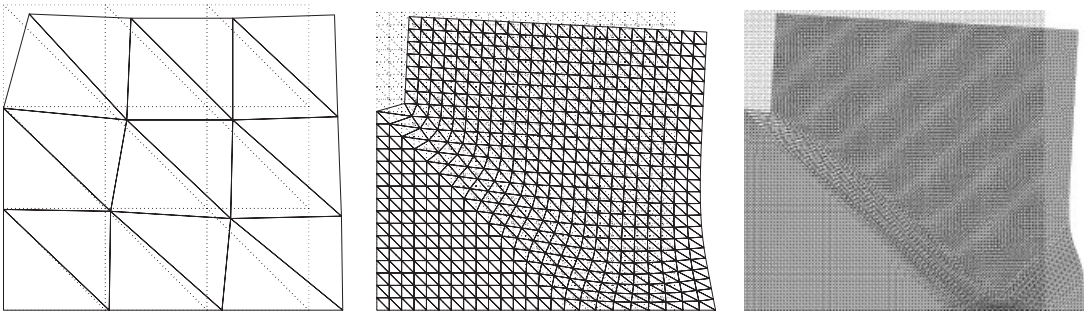
Number of refinements	Number of elements	Lower bound $\lambda_h^{*\mathrm{LB}}$	Upper bound $\lambda_h^{*\mathrm{UB}}$	Bound gap Δ_h	Maximum relative error, e_h (%)
Uniform meshing					
0	18	1.0414	1.5690	0.52765	20.213
1	72	1.1830	1.4408	0.25774	9.823
2	288	1.2352	1.3619	0.12662	4.876
3	1152	1.2553	1.3183	0.06303	2.449
4	4608	1.2639	1.2960	0.03206	1.252
5	18432	1.2679	1.2844	0.01643	0.644

analysis problems. Since the exact collapse multiplier solution is not known for this example, the last column in the table corresponds to the maximum possible relative error, e_h, associated with the *predictor*, λ_h^{pr} (the average of the upper and lower bounds). Clearly, for each refinement, the exact solution belongs to the interval $\lambda_h^{\mathrm{pr}} \pm e_h$.

Figure 15 illustrates for different computational meshes. Note that the upper part of the domain moves as a rigid body, sliding over the lower left region, which remains static. Moreover, the lower right part is pushed horizontally by the upper region of the body. Finally, Figure 16 shows a linear rate of convergence for the bound gap, Δ_h.

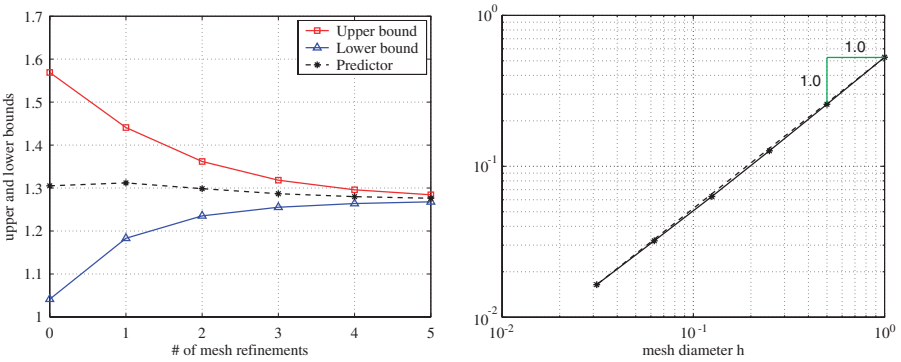
6.2.2. Adaptive meshing. Nine refinements of the initial mesh have been considered, for which the results are given, numerically, in Table V. These results are also shown, graphically, in Figure 19. The adaptive meshing is found to capture the collapse mechanism very accurately, as can be observed in Figures 17 and 18. Finally, Figure 20 shows the efficiency of the adaptive meshing when compared with the uniform meshing. Indeed, the rate of convergence for the bound gap is much higher in the adaptive case. Note also that in the sixth adaptive refinement, with 1550 elements, a bound gap of only 0.01515 is obtained. In the uniform meshing, the finer mesh yielded a bound gap of 0.01643 and used 18435 elements. In the last adaptive refinement (5568 elements), the bound gap reduces to 0.00490, which translates into a maximum error of 0.192%.

DOI: 10.1002/nme



Initial mesh | After 3 refinements | After 5 refinements

Figure 15. Slotted block problem 1—deformed geometry using uniform meshing.



Upper and lower bounds | Bound gap rate of convergence

Figure 16. Slotted block problem 1—convergence using uniform meshing.

Table V. Results for the slotted block problem in plane stress using adaptive meshing.

Number of refinements	Number of elements	Lower bound $\lambda_h^{*\mathrm{LB}}$	Upper bound $\lambda_h^{*\mathrm{UB}}$	Bound gap Δ_h	Maximum error (%)
Adaptive meshing					
0	18	1.0414	1.5690	0.52765	20.213
1	70	1.2181	1.4402	0.22206	8.353
2	254	1.2496	1.3615	0.11190	4.285
3	483	1.2593	1.3202	0.06095	2.363
4	714	1.2663	1.3028	0.03654	1.422
5	1082	1.2690	1.2907	0.02173	0.849
6	1550	1.2703	1.2855	0.01515	0.593
7	2538	1.2710	1.2808	0.00979	0.384
8	3564	1.2714	1.2785	0.00705	0.276
9	5568	1.2719	1.2768	0.00490	0.192

DOI: 10.1002/nme

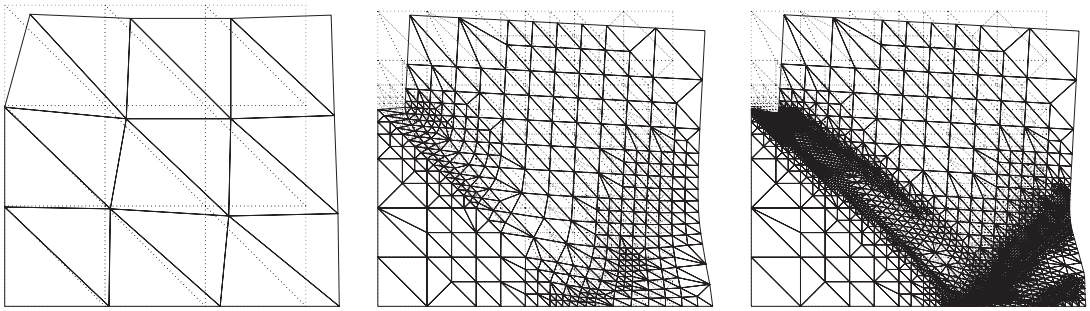


Figure 17. Slotted block problem 1—deformed geometry using adaptive meshing.

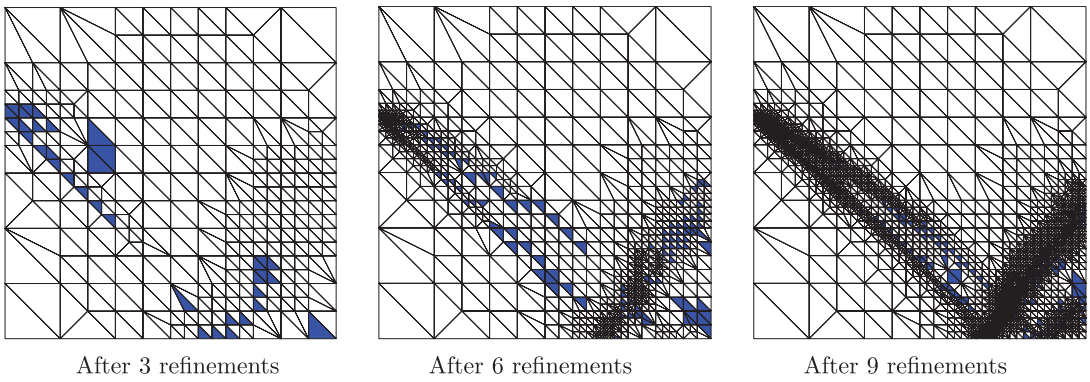


Figure 18. Slotted block problem 1—adaptive meshing strategy: elements to be refined.

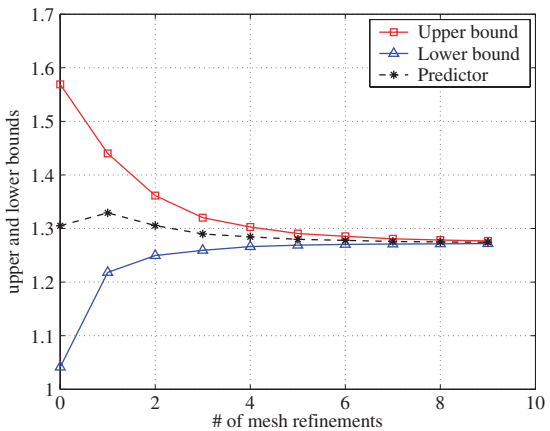


Figure 19. Slotted block problem 1—bounds using adaptive meshing.

DOI: 10.1002/nme

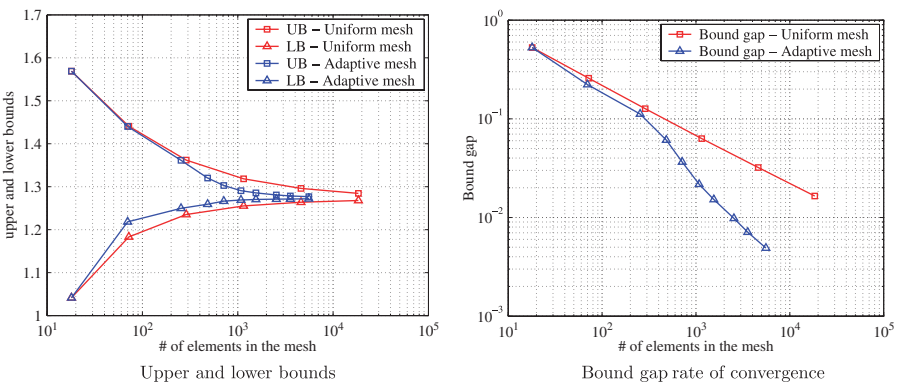


Figure 20. Slotted block problem 1—comparison of adaptive meshing with uniform meshing.

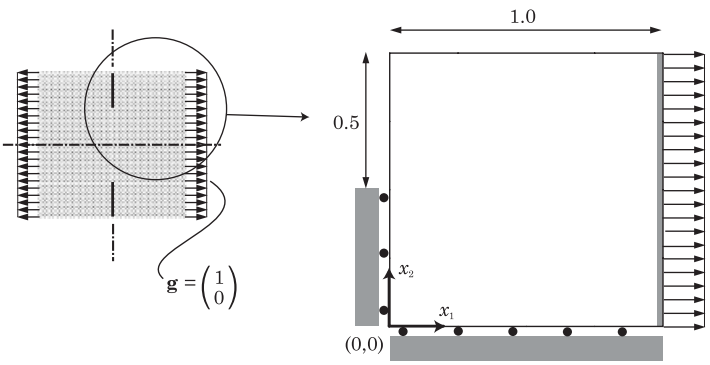


Figure 21. Geometry and loads for the slotted block problem in plane strain.

6.3. Slotted square block in plane strain

Very similar to the second example, the slotted square block considered here has a cut of $\frac{1}{4}$ of the total height of the block, as shown in Figure 21, and is solved assuming plane strain. This example is a classical problem and is well documented in the literature. In particular, it has been studied in [11] (main reference), [4, 8, 10, 13, 25] or, more recently, in [17]. Although a rigorous exact solution is not available, the extrapolated value $\lambda^* = 1.13156$, derived in [11], is considered here as a sufficiently accurate solution. This problem is very appropriate to achieve different purposes: (i) verify the correctness of the method in plane strain; (ii) address convergence studies for various cases; (iii) compare the results with those of other sources; and (iv) show the relevance that the choice of the computational mesh might have on the accuracy of the results. In particular, to illustrate the fourth point, the same problem is solved starting from three different initial meshes, which are shown in Figure 22 and denoted by $\mathscr{T}_{h,1}$, $\mathscr{T}_{h,2}$ and $\mathscr{T}_{h,3}$. Clearly, the use of different interpolation spaces will lead to different solutions. In particular, note that, in all the meshes, the edges of the elements define preferential slip directions in the domain.

DOI: 10.1002/nme

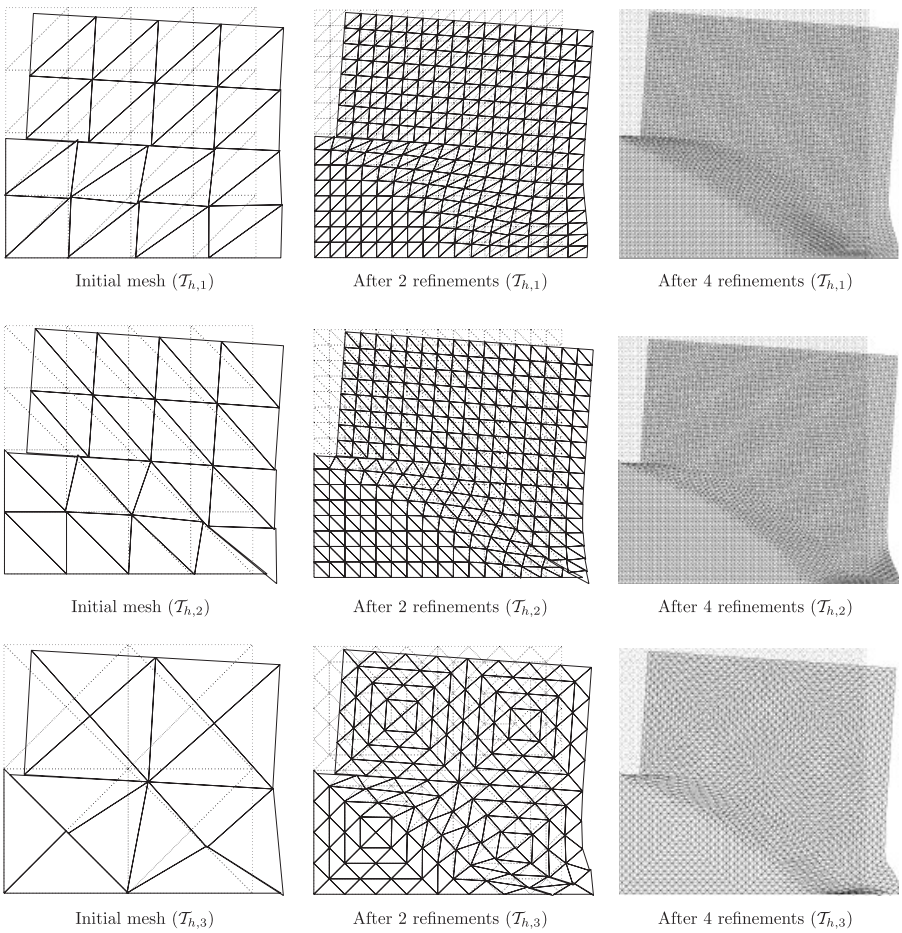


Figure 22. Slotted block problem 2—deformed geometry using uniform meshing.

6.3.1. Uniform meshing. For the three meshes under consideration, the numerical results are summarized in Table VI, and the collapse geometries are given in Figure 22. Moreover, Figures 23 and 24 show, respectively, the upper and lower bounds obtained and, also, the rate of convergence for the bound gap as well as for the upper and lower bound relative errors. Note that the accuracies obtained for the different meshes differ substantially. Indeed, in terms of bound gap and with only half of the elements, $\mathscr{T}_{h,3}$ outperforms $\mathscr{T}_{h,2}$ which, in turn, works better than $\mathscr{T}_{h,1}$. These results are not surprising, as explained next. Since the collapse mechanisms are typically highly localized, one can expect better solutions if the preferential directions of the meshes match, or are similar, to the slip-lines of the real mechanism. In this problem, two slip-lines exist. The main one has its origin at the end of the cut and goes down to a point near the right boundary, describing a logarithmic spiral. From this point, a second discontinuity develops following a direction perpendicular to the previous main slip-line. Then, one expects $\mathscr{T}_{h,2}$ to work better than $\mathscr{T}_{h,1}$, since its preferential direction is similar to the main slip-line of the mechanism, whereas

DOI: 10.1002/nme

Table VI. Results for the slotted block problem in plane strain using uniform meshing.

Number of refinements	Number of elements	Lower bound $\lambda_h^{*\mathrm{LB}}$	Upper bound $\lambda_h^{*\mathrm{UB}}$	Bound gap Δ_h	Lower bound error (%)	Upper bound error (%)
Uniform meshing with initial mesh $\mathscr{T}_{h,1}$						
0	32	0.9952	1.3688	0.37355	12.051	20.966
1	128	1.0760	1.2840	0.20797	4.910	13.472
2	512	1.1069	1.2240	0.11716	2.179	8.169
3	2048	1.1198	1.1862	0.06647	1.039	4.829
4	8192	1.1258	1.1641	0.03833	0.509	2.876
Uniform meshing with initial mesh $\mathscr{T}_{h,2}$						
0	32	0.9109	1.2444	0.33344	19.496	9.972
1	128	1.0341	1.2061	0.17193	8.613	6.587
2	512	1.0871	1.1784	0.09128	3.929	4.139
3	2048	1.1103	1.1601	0.04976	1.879	2.522
4	8192	1.1212	1.1489	0.02770	0.916	1.532
Uniform meshing with initial mesh $\mathscr{T}_{h,3}$						
0	16	1.11405	1.22888	0.114830	1.5478	8.6001
1	64	1.12690	1.20063	0.073735	0.4119	6.1044
2	256	1.13046	1.17785	0.047395	0.0973	4.0911
3	1024	1.13126	1.16057	0.029304	0.0262	2.5636
4	4096	1.13147	1.14956	0.018082	0.0076	1.5904

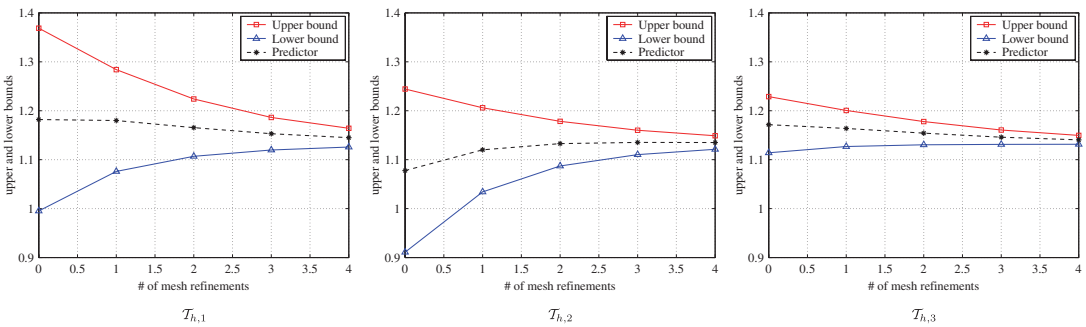


Figure 23. Slotted block problem 2—upper and lower bounds using uniform meshing.

in $\mathscr{T}_{h,1}$ the preferential direction only matches the second, less important, slip-line. Finally, $\mathscr{T}_{h,3}$ has to be the best mesh, since more flexibility is introduced and the elements can slide in both directions.

Besides the mesh phenomenon, it is also remarkable that the errors in the lower bound are clearly inferior to the upper bound ones. This agrees with the observation made in the cantilever example. In the same line, Figure 24 shows that the rate of convergence for the bound gap is, for all cases, sub-linear, as was also observed in the cantilever example. This is due to the poor convergence rate of the upper bound error, since the lower bound error converges, at least, linearly. Note the extraordinary accuracy and rate of convergence obtained in the lower bound problem

DOI: 10.1002/nme

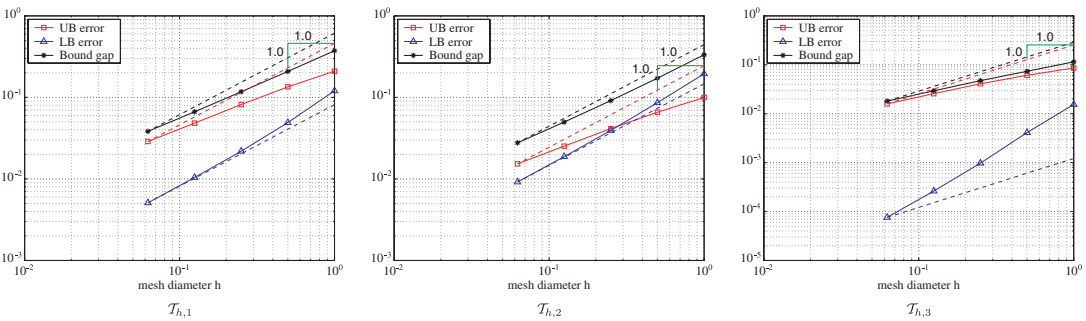


Figure 24. Slotted block problem 2—rates of convergence using uniform meshing.

Table VII. Results for the slotted block problem in plane strain using adaptive meshing.

Number of refinements	Number of elements	Lower bound $\lambda_h^{*\mathrm{LB}}$	Upper bound $\lambda_h^{*\mathrm{UB}}$	Bound gap Δ_h	Lower bound error (%)	Upper bound error (%)
Adaptive meshing with initial mesh $\mathcal{T}_{h,1}$						
0	32	0.9952	1.3688	0.37355	12.051	20.966
1	87	1.0708	1.2840	0.21321	5.370	13.472
2	291	1.1107	1.2240	0.11336	1.843	8.169
3	962	1.1213	1.1862	0.06484	0.907	4.829
4	2821	1.1265	1.1645	0.03802	0.447	2.911
5	6901	1.1290	1.1522	0.02320	0.226	1.824
Adaptive meshing with initial mesh $\mathcal{T}_{h,2}$						
0	32	0.9109	1.2444	0.33344	19.496	9.972
1	74	1.0514	1.2063	0.15494	7.084	6.605
2	262	1.1055	1.1784	0.07295	2.303	4.139
3	807	1.1195	1.1601	0.04063	1.066	2.522
4	2065	1.1256	1.1491	0.02346	0.527	1.550
5	4621	1.1286	1.1428	0.01421	0.261	0.993
Adaptive meshing with initial mesh $\mathcal{T}_{h,3}$						
0	16	1.11405	1.22888	0.114830	1.5478	8.6001
1	38	1.12425	1.20063	0.076386	0.6462	6.1044
2	114	1.13035	1.17830	0.047951	0.1068	4.1309
3	397	1.13126	1.16087	0.029602	0.0261	2.5899
4	1089	1.13147	1.15011	0.018635	0.0076	1.6392
5	2716	1.13153	1.14347	0.011947	0.0029	1.0528
6	5913	1.13154	1.13902	0.007480	0.0021	0.6589

for the mesh $\mathcal{T}_{h,3}$: after four refinements, the lower bound error is only 0.0076% and its rate of convergence is found to be quadratic.

6.3.2. Adaptive meshing. When mesh adaptivity is used, previous observations are even more valid. In this aspect, $\mathcal{T}_{h,3}$ continues to behave clearly better than $\mathcal{T}_{h,2}$ and $\mathcal{T}_{h,1}$, as can be observed in Table VII and Figures 26 and 27. Also, the upper bound is consistently less accurate than the lower bound and has a worst convergence rate. Furthermore, $\mathcal{T}_{h,3}$ yields an outstanding accuracy

DOI: 10.1002/nme

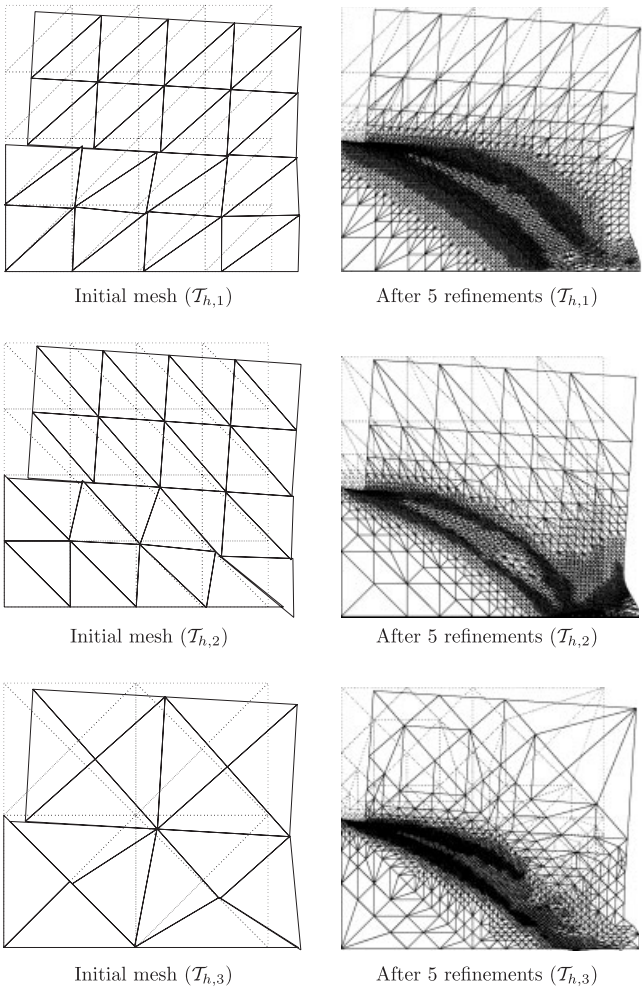


Figure 25. Slotted block problem 2—deformed geometry using adaptive meshing.

for the lower bound, with a final relative error of only 0.0021%. It is interesting to compare, in Figure 25, the meshes $\mathcal{T}_{h,1}$ and $\mathcal{T}_{h,2}$ after five refinements. Note that, in the first case, the refinement is principally concentrated in a wide region around the main slip-line (the one starting from the cut and going down to the right side), and less refinement is performed around the second slip-line (the one in the lower right corner). However, in $\mathcal{T}_{h,2}$, the refinement is very thin around the main slip-line, and a dense refinement is necessary to capture the second slip-line, whose direction is perpendicular to the diagonal edges of the mesh. This clearly agrees with the expected relevance of the preferential directions.

Figure 27 compares the rate of convergence for both the lower bound error and the bound gap, for the three meshes considered under uniform and adaptive meshing. Clearly, mesh adaptivity permits gains in both accuracy and convergence. Also, the impact of the initial mesh is proved.

DOI: 10.1002/nme

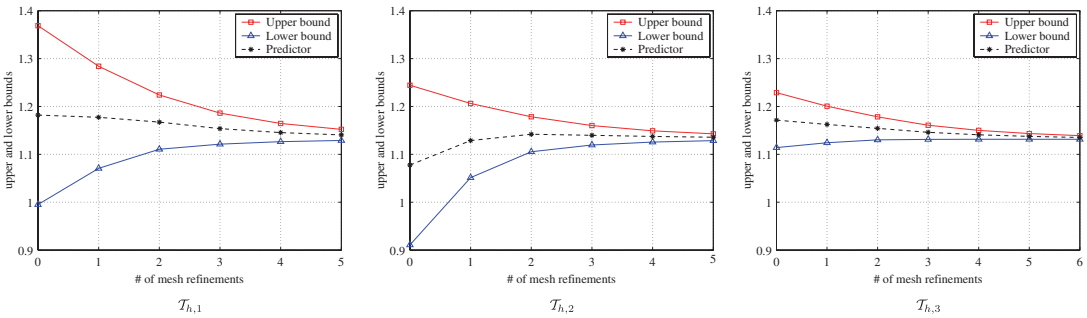


Figure 26. Slotted block problem 2—bounds using adaptive meshing.

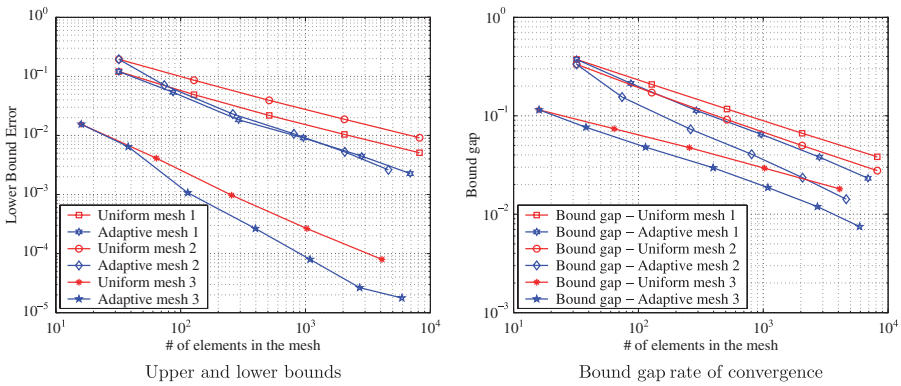


Figure 27. Slotted block problem 2—comparison of adaptive meshing with uniform meshing.

6.3.3. Comparison of the results with other approaches. The results obtained for this example are now compared with those given in two recent sources [11, 17].

In Reference [11], a mixed formulation stress velocities is used to approximate, but not bound, the collapse multiplier. Linear, continuous interpolations are chosen for both the stresses and the velocities on a triangular finite element mesh whose triangles are oriented as in the mesh $\mathcal{T}_{h,2}$. The best estimate obtained for λ^* is 1.13582, for a mesh of 115 200 elements. When using a coarser mesh of 7200 elements, the estimate is 1.1485. Note that this value is worse than the upper bound obtained when adaptive meshing is applied to the initial mesh $\mathcal{T}_{h,2}$. Indeed, with only 4621 elements, the value obtained is 1.1428. Moreover, the method proposed in this article yields a strict upper bound, as opposed to just an approximation. The rate of convergence in [11] is linear, which coincides with our results.

In [17], the problem is solved using a lower bound formulation based on a discontinuous, piecewise linear interpolation of the stresses (the same formulation used here for the lower bound problem). The initial mesh considered is of type $\mathcal{T}_{h,3}$, which explains the quadratic rate of convergence reported when refining the mesh uniformly. Recall (see Figure 24) that this result is only a consequence of the mesh. Indeed, for the lower bound case, the same formulation yields a

DOI: 10.1002/nme

Table VIII. Computational cost of solving the lower and upper bound problems for the finest uniform and adaptive meshes used in all previous examples.

			Lower bound problem				Upper bound problem		
Example	Meshing	No. of elements	No. of equations	No. of variables	SeDuMi	SDPT3	No. of equations	No. of variables	SeDuMi
Cantilever	Uniform	8704	122 176	130 561	1:57	4:18	52 384	69 633	0:21
	Adaptive	5506	77 252	82 591	1:07	1:27	33 120	44 049	0:14
Block1	Uniform	18 432	258 496	276 481	—	11:12	110 816	147 457	1:22
	Adaptive	5568	78 058	83 521	1:07	1:27	33 461	44 545	0:14
Beam	Uniform	6912	76 432	82 945	2:35	10:26	102 864	123 201	1:31
	Adaptive	4788	52 858	57 457	1:51	7:05	71 538	85 713	0:53
Block2-$\mathscr{T}_{h,1}$	Uniform	8192	90 432	98 305	—	25:09	122 176	146 433	1:38
	Adaptive	6901	75 979	82 813	—	21:43	103 243	123 879	1:10
Block2-$\mathscr{T}_{h,2}$	Uniform	8192	90 432	98 305	—	26:27	122 176	146 433	1:36
	Adaptive	4621	50 929	55 453	—	11:31	69 073	82 839	0:43
Block2-$\mathscr{T}_{h,3}$	Uniform	4906	45 216	49 153	—	8:16	61 088	73 217	0:38
	Adaptive	5913	65 079	70 957	—	14:30	88 559	106 263	0:56

Note: For the cases not solved, SeDuMi ran into numerical problems.

linear rate of convergence when using $\mathscr{T}_{h,1}$ or $\mathscr{T}_{h,2}$, whereas $\mathscr{T}_{h,3}$ gives quadratic convergence. Since the formulation in [17] coincides with the lower bound approach presented in this article, the numerical results are the same when uniform meshing is used. However, when adaptive meshing is considered, considerable gains are obtained. For instance, with 2716 elements, the present approach yields a lower bound of 1.13153 (error of 0.0029%), whereas in [17], 5776 elements are needed to obtain a worse lower bound of 1.13150 (0.0052% error).

6.4. Computational cost of previous examples

This section gives an indication of the computational cost required to solve both the lower and the upper bound problems for the finest uniform and adaptive meshes used in all previous examples. To solve the bound problems, the free software SeDuMi 1.05R5 and SDPT3-3.02 were used in Matlab 6.5.1 on a Pentium®4 2.53 GHz desktop PC. For both cases, the NT search direction was considered (see Section 3).

Table VIII shows, for each example, the number of elements in the meshes considered, the number of equations and variables as well as the time (in minutes and seconds) required to solve the optimization problems.

When solving the problems, SDPT3 was very robust and it always yielded optimal solutions. On the other hand, for some cases and always for the lower bound problem, SeDuMi ran into numerical difficulties that stopped the solution process. Possible reasons for this behavior has been studied in [19]. For the cases where SeDuMi could solve, it was systematically faster than SDPT3.

Note that examples of considerable size can be solved. For instance, the finest uniform mesh in the second example (Block1) consists of 18 432 elements resulting in matrices of dimensions (258 496, 276 481) (lower bound) and (110 816, 147 457) (upper bound), leading to the bound problems being solved in only 11 min 12 s and 1:22 s, respectively. For the adaptive case, matrices of dimensions (78 058, 83 521) and (33 461, 44 545) were addressed in 1:07 and 0:14 s.

DOI: 10.1002/nme

7. CONCLUSIONS

The paper has presented a methodology for the evaluation of strict upper and lower bounds in limit analysis for convex yield criteria using SOCP. Applications have been presented in plane stress and strain using linear finite element interpolation spaces. For the lower bound evaluation, static interpolation spaces have been used in which the stresses are piecewise linear but discontinuous across element edges and displacement fields are piecewise constant. Action and reaction equilibrium across element edges is enforced by means of linear displacement fields acting as Lagrange multipliers. In the case of the upper bound evaluation, the interpolation spaces are reversed and linear displacements fields are used together with constant stresses in each element. For plane strain applications, the incompressibility constraint requires the displacement fields to be discontinuous across element edges with the impenetrability constraint being enforced via linear traction Lagrange multipliers on element edges. Similar displacement fields could be used for plane stress, but the size of the problem is significantly reduced by using continuous displacement interpolation fields across elements.

The paper has shown that the gap between upper and lower bounds can be expressed as a sum of positive elemental contributions. The relative values of these contributions can then be used to derive a nested mesh adaptivity procedure to ensure that the gap is closed to within a prescribed tolerance. Examples demonstrating this process and the resulting improved rate of convergence have been given for both plane stress and plane strain applications.

APPENDIX A: DISCRETIZATION OF THE EQUILIBRIUM EQUATIONS FOR THE UPPER BOUND PROBLEM

A.1. Equilibrium constraint in plane stress

To discretize the equilibrium equation $a(\boldsymbol{\sigma},\mathbf{u})=\lambda F(\mathbf{u}), \forall \mathbf{u}\in Y$, where $a(\boldsymbol{\sigma},\mathbf{u})$ is given in (2), one replaces the stresses, $\boldsymbol{\sigma}$, and velocities, $\mathbf{u}$, by their global interpolations (54) and (56), respectively

$$a(\boldsymbol{\sigma},\mathbf{u})=\lambda F(\mathbf{u}) \implies \sum_{i,j=1}^{2}\int_{\Omega}\sigma_{ij}\frac{\partial u_i}{\partial x_j}\,\mathrm{d}V=\lambda\sum_{i=1}^{2}\left(\int_{\Omega}f_i u_i\,\mathrm{d}V+\int_{\Gamma^{\mathrm{N}}}g_i u_i\,\mathrm{d}S\right) \tag{A1}$$

$$\sum_{e=1}^{E}\sum_{A=1}^{N}\sum_{i,j=1}^{2}\int_{\Omega}\sigma_{ij}^{e}\psi_e(\mathbf{x})u_i^A\frac{\partial\phi_A(\mathbf{x})}{\partial x_j}\,\mathrm{d}V=\lambda\sum_{A=1}^{N}\sum_{i=1}^{2}u_i^A\left(\int_{\Omega}f_i\phi_A(\mathbf{x})\,\mathrm{d}V+\int_{\Gamma^{\mathrm{N}}}g_i\phi_A(s)\,\mathrm{d}S\right) \tag{A2}$$

where $\phi_A(s)$ is the restriction of $\phi_A(\mathbf{x})$ to the boundary Γ^{N}. Taking into account that the stress shape function $\psi_e(\mathbf{x})$ is equal to 1 inside the element Ω^e and vanishes outside, (A2) can be simplified as follows:

$$\sum_{e=1}^{E}\sum_{A=1}^{N}\sum_{i,j=1}^{2}\int_{\Omega^e}\sigma_{ij}^{e}u_i^A\frac{\partial N_A^e(\mathbf{x})}{\partial x_j}\,\mathrm{d}V=\lambda\sum_{A=1}^{N}\sum_{i=1}^{2}u_i^A\sum_{e=1}^{E}\left(\int_{\Omega^e}f_i N_A^e(\mathbf{x})\,\mathrm{d}V+\int_{\varsigma_e^{\mathrm{N}}}g_i N_A^e(s)\,\mathrm{d}S\right) \tag{A3}$$

Note that $\phi_A(\mathbf{x})$ has been substituted by $N_A^e(\mathbf{x})$ because the integrals are now restricted to the elements. To avoid a cumbersome notation, the global nodal index A has been maintained in the

DOI: 10.1002/nme

local shape functions. In matrix form, and using the notation $N^e_{a,i} = \partial N^e_a(\mathbf{x})/\partial x_i$, where $i = 1:2$, (A3) reads as follows:

$$\sum_{e=1}^{E}\sum_{A=1}^{N}(u_1^A, u_2^A)\begin{pmatrix}\displaystyle\int_{\Omega^e} N^e_{A,1}\,\mathrm{d}V & 0 & \displaystyle\int_{\Omega^e} N^e_{A,2}\,\mathrm{d}V\\ 0 & \displaystyle\int_{\Omega^e} N^e_{A,2}\,\mathrm{d}V & \displaystyle\int_{\Omega^e} N^e_{A,1}\,\mathrm{d}V\end{pmatrix}\begin{pmatrix}\sigma^e_{11}\equiv\sigma^e_1\\ \sigma^e_{22}\equiv\sigma^e_2\\ \sigma^e_{12}\equiv\sigma^e_3\end{pmatrix} \tag{A4}$$

$$=\lambda\sum_{A=1}^{N}(u_1^A, u_2^A)\begin{pmatrix}\displaystyle\sum_{e=1}^{E}\left(\int_{\Omega^e} f_1 N^e_A(\mathbf{x})\,\mathrm{d}V+\int_{\xi^{\mathrm{N}}_e} g_1 N^e_A(\mathbf{x})\,\mathrm{d}S\right)\\ \displaystyle\sum_{e=1}^{E}\left(\int_{\Omega^e} f_2 N^e_A(\mathbf{x})\,\mathrm{d}V+\int_{\xi^{\mathrm{N}}_e} g_2 N^e_A(\mathbf{x})\,\mathrm{d}S\right)\end{pmatrix} \tag{A5}$$

or, equivalently,

$$\sum_{e=1}^{E}\sum_{A=1}^{N}(\underline{u}_h^A)^{\mathrm{T}}\underline{\underline{B}}^e_A\tilde{\underline{\sigma}}^e_h=\lambda\sum_{A=1}^{N}(\underline{u}_h^A)^{\mathrm{T}}\underline{F}^A_h \tag{A6}$$

Finally, the above expression can be expressed in a more compact form by using global matrices and vectors:

$$(\underline{u}_h)^{\mathrm{T}}\underline{\underline{A}}^{\mathrm{eq}}\tilde{\underline{\sigma}}_h=\lambda(\underline{u}_h)^{\mathrm{T}}\underline{F}^{\mathrm{eq}}_h \tag{A7}$$

where

$$\underline{\underline{A}}^{\mathrm{eq}}=\begin{pmatrix}\underline{\underline{B}}^1_1 & \underline{\underline{B}}^2_1 & \cdots & \cdots & \underline{\underline{B}}^E_1\\ \underline{\underline{B}}^1_2 & \underline{\underline{B}}^2_2 & \cdots & \cdots & \underline{\underline{B}}^E_2\\ \vdots & & \ddots & & \vdots\\ \vdots & & & \ddots & \vdots\\ \underline{\underline{B}}^1_A & \cdots & \cdots & \cdots & \underline{\underline{B}}^E_A\end{pmatrix},\quad \underline{u}_h=\begin{pmatrix}\underline{u}^1_h\\ \vdots\\ \underline{u}^A_h\\ \vdots\\ \underline{u}^N_h\end{pmatrix},\quad \tilde{\underline{\sigma}}_h=\begin{pmatrix}\tilde{\underline{\sigma}}^1_h\\ \vdots\\ \tilde{\underline{\sigma}}^e_h\\ \vdots\\ \tilde{\underline{\sigma}}^E_h\end{pmatrix},\quad \underline{F}^{\mathrm{eq}}_h=\begin{pmatrix}\underline{F}^1_h\\ \vdots\\ \underline{F}^A_h\\ \vdots\\ \underline{F}^N_h\end{pmatrix} \tag{A8}$$

Clearly, the dimensions of the matrix $\underline{\underline{A}}^{\mathrm{eq}}$ are $(2\times A, 3\times E)$. Since Equation (A7) must be satisfied $\forall\mathbf{u}_h\in Y_h^{UB}$, one can eliminate the nodal displacement vector $\underline{u}_h$, thereby obtaining the following global equilibrium equation:

$$\underline{\underline{A}}^{\mathrm{eq}}\tilde{\underline{\sigma}}_h=\lambda\underline{F}^{\mathrm{eq}}_h \tag{A9}$$

DOI: 10.1002/nme

A.2. Equilibrium constraint in plane strain

In plane strain, the weak form of equilibrium is given by

$$
\begin{gathered}
a(\boldsymbol{\sigma},\mathbf{t},\mathbf{u}) = \lambda F(\mathbf{u}) \quad \forall \mathbf{u} \in Y \\
\underbrace{\sum_{i,j=1}^{2} \int_{\Omega} \sigma_{ij} \frac{\partial u_i}{\partial x_j} \,\mathrm{d}V}_{1} + \underbrace{\sum_{\xi_e^{e'} \in \mathscr{E}^{\mathscr{C}}} \sum_{i'=1}^{2} \int_{\xi_e^{e'}} t_{i'}^{\xi_e^{e'}} (u_{i'}^{e'} - u_{i'}^{e}) \,\mathrm{d}S}_{2} \\
= \underbrace{\lambda \sum_{i=1}^{2} \left(\int_{\Omega} f_i u_i \,\mathrm{d}V + \int_{\Gamma^{\mathrm{N}}} g_i u_i \,\mathrm{d}S \right)}_{3} \quad \forall \mathbf{u} \in Y
\end{gathered}
\tag{A10}
$$

To discretize (A10), one inserts interpolations (57) (for $\mathbf{u}_h$), (58) (for $\boldsymbol{\sigma}_h$) and (59) (for $\mathbf{t}_h$). Note that the terms (1) and (3) in the above equation are the same as in the plane stress case.

Using (57) and (58), term (1) results in the following expression:

$$
\begin{aligned}
&\sum_{e=1}^{E} \sum_{a=1}^{3} (u_1^{a,e}, u_2^{a,e}) \begin{pmatrix} \int_{\Omega^e} N_{a,1}^e \,\mathrm{d}V & 0 & \int_{\Omega^e} N_{a,2}^e \,\mathrm{d}V \\ 0 & \int_{\Omega^e} N_{a,2}^e \,\mathrm{d}V & \int_{\Omega^e} N_{a,1}^e \,\mathrm{d}V \end{pmatrix} \begin{pmatrix} \sigma_1^e \\ \sigma_2^e \\ \sigma_3^e \end{pmatrix} \\
&= \sum_{e=1}^{E} \left(\sum_{a=1}^{3} (\underline{u}_h^{a,e})^{\mathrm{T}} \underline{\underline{B}}_a^e \right) \tilde{\underline{\sigma}}_h^e = \sum_{e=1}^{E} (\underline{u}^{1,e}, \underline{u}^{2,e}, \underline{u}^{3,e}) \begin{pmatrix} \underline{\underline{B}}_1^e \\ \underline{\underline{B}}_2^e \\ \underline{\underline{B}}_3^e \end{pmatrix} \tilde{\underline{\sigma}}_h^e \\
&= \sum_{e=1}^{E} \underline{u}^e \underline{\underline{B}}^e \tilde{\underline{\sigma}}_h^e = (\tilde{\underline{u}}_h)^{\mathrm{T}} \tilde{\underline{\underline{A}}}^{\mathrm{eq1}} \tilde{\underline{\sigma}}_h
\end{aligned}
\tag{A11}
$$

where $\tilde{\underline{u}}_h$ is a vector collecting the $6 \times E$ nodal displacements, $\tilde{\underline{\underline{A}}}^{\mathrm{eq1}}$ is a matrix of dimensions $(6 \times E, 3 \times E)$ and $\tilde{\underline{\sigma}}_h$ is a $3 \times E$ vector of elemental stresses (the same as in (A8)).

Regarding term (3), using (57) one obtains

$$
\begin{aligned}
&\sum_{e=1}^{E} \sum_{a=1}^{3} (u_1^{a,e}, u_2^{a,e}) \begin{pmatrix} \int_{\Omega^e} f_1^e N_A^e(\mathbf{x}) \,\mathrm{d}V + \int_{\xi_e^{\mathrm{N}}} g_1^{\xi_e^{\mathrm{N}}} N_A^e(\mathbf{x}) \,\mathrm{d}S \\ \int_{\Omega^e} f_2^e N_A^e(\mathbf{x}) \,\mathrm{d}V + \int_{\xi_e^{\mathrm{N}}} g_2^{\xi_e^{\mathrm{N}}} N_A^e(\mathbf{x}) \,\mathrm{d}S \end{pmatrix} = \sum_{e=1}^{E} \left(\sum_{a=1}^{3} (\underline{u}_h^{a,e})^{\mathrm{T}} \underline{F}_{ha}^e \right) \\
&= \sum_{e=1}^{E} (\underline{u}^{1,e}, \underline{u}^{2,e}, \underline{u}^{3,e}) \begin{pmatrix} \underline{F}_{h1}^e \\ \underline{F}_{h2}^e \\ \underline{F}_{h3}^e \end{pmatrix} = \sum_{e=1}^{E} \underline{u}^e \underline{F}_h^e = (\tilde{\underline{u}}_h)^{\mathrm{T}} \tilde{\underline{F}}_h^{\mathrm{eq}}
\end{aligned}
\tag{A12}
$$

where $\tilde{\underline{F}}_h^{\mathrm{eq}}$ is a vector of $6 \times E$ discontinuous nodal forces.

DOI: 10.1002/nme

Let $\mathbf{n}^{\xi_e^{e'}}$ and $\boldsymbol{\tau}^{\xi_e^{e'}}$ be orthonormal vectors, following the directions $x_{1'}$ and $x_{2'}$, respectively, but expressed in the conventional rectangular coordinates $x_1 - x_2$. Thus, term (2) can be rewritten as

$$\sum_{\xi_e^{e'} \in \mathscr{E}^{\mathbb{O}}} \int_{\xi_e^{e'}} (t_{1'}^{\xi_e^{e'}}, t_{2'}^{\xi_e^{e'}}) \begin{pmatrix} (\mathbf{u}_h^{e'} - \mathbf{u}_h^{e}) \cdot \mathbf{n}^{\xi_e^{e'}} \\ (\mathbf{u}_h^{e'} - \mathbf{u}_h^{e}) \cdot \boldsymbol{\tau}^{\xi_e^{e'}} \end{pmatrix} \mathrm{d}S$$

$$= \sum_{\xi_e^{e'} \in \mathscr{E}^{\mathbb{O}}} \int_{\xi_e^{e'}} (t_{1'}^{\xi_e^{e'}}, t_{2'}^{\xi_e^{e'}}) \underbrace{\begin{pmatrix} n_1^{\xi_e^{e'}} & n_1^{\xi_e^{e'}} \\ \tau_1^{\xi_e^{e'}} & \tau_1^{\xi_e^{e'}} \end{pmatrix}}_{\underline{\underline{M}}^{\xi_e^{e'}}} \begin{pmatrix} u_1^{e'} - u_1^{e} \\ u_2^{e'} - u_2^{e} \end{pmatrix} \mathrm{d}S \tag{A13}$$

To simplify the notation in the following expressions, the edge $\xi_e^{e'}$ will be denoted as ξ. Moreover, the interpolation of the displacements on the edge considered will be expressed as follows:

$$u_i^e(s) = u_i^{\alpha_1} N_1^{\xi}(s) + u_i^{\alpha_2} N_1^{\xi}(s), \quad u_i^{e'}(s) = u_i^{\beta_1} N_1^{\xi}(s) + u_i^{\beta_2} N_1^{\xi}(s), \quad i = 1:2 \tag{A14}$$

Now, introducing in (A13) interpolations (59) for $\mathbf{t}_h$, and (A14) for $\mathbf{u}_h^e$ and $\mathbf{u}_h^{e'}$, one has

$$\int_{\xi} (t_{1'}^{1,\xi} N_1^{\xi} + t_{1'}^{2,\xi} N_2^{\xi}, t_{2'}^{1,\xi} N_1^{\xi} + t_{2'}^{2,\xi} N_2^{\xi}) \underline{\underline{M}}^{\xi} \begin{pmatrix} (u_1^{\beta_1} - u_1^{\alpha_1}) N_1^{\xi} + (u_1^{\beta_2} - u_1^{\alpha_2}) N_2^{\xi} \\ (u_2^{\beta_1} - u_2^{\alpha_1}) N_1^{\xi} + (u_2^{\beta_2} - u_2^{\alpha_2}) N_2^{\xi} \end{pmatrix} \mathrm{d}S \tag{A15}$$

Expanding (A15), one obtains

$$(t_{1'}^{1,\xi}, t_{2'}^{1,\xi}) \underline{\underline{B}}_{11}^{\xi} \begin{pmatrix} u_1^{\beta_1} - u_1^{\alpha_1} \\ u_2^{\beta_1} - u_2^{\alpha_1} \end{pmatrix} + (t_{1'}^{1,\xi}, t_{2'}^{1,\xi}) \underline{\underline{B}}_{12}^{\xi} \begin{pmatrix} u_1^{\beta_2} - u_1^{\alpha_2} \\ u_2^{\beta_2} - u_2^{\alpha_2} \end{pmatrix}$$

$$+ (t_{1'}^{2,\xi}, t_{2'}^{2,\xi}) \underline{\underline{B}}_{12}^{\xi} \begin{pmatrix} u_1^{\beta_1} - u_1^{\alpha_1} \\ u_2^{\beta_1} - u_2^{\alpha_1} \end{pmatrix} + (t_{1'}^{2,\xi}, t_{2'}^{2,\xi}) \underline{\underline{B}}_{22}^{\xi} \begin{pmatrix} u_1^{\beta_2} - u_1^{\alpha_2} \\ u_2^{\beta_2} - u_2^{\alpha_2} \end{pmatrix} \tag{A16}$$

where $\underline{\underline{B}}_{ij}^{\xi} = \underline{\underline{M}}^{\xi} \int_{\xi} N_i^{\xi} N_j^{\xi} \mathrm{d}S$, for $i, j = 1:2$. Now, the local displacements have to be related to the global vector of displacements $\underline{\tilde{u}}_h$. This is done by means of two matrices, $\underline{\underline{A}}_1^{\xi}$ and $\underline{\underline{A}}_2^{\xi}$, of dimensions $(2, 6 \times E)$:

$$\begin{pmatrix} u_1^{\beta_1} - u_1^{\alpha_1} \\ u_2^{\beta_1} - u_2^{\alpha_1} \end{pmatrix} = \underline{\underline{A}}_1^{\xi} \underline{\tilde{u}}_h, \quad \begin{pmatrix} u_1^{\beta_2} - u_1^{\alpha_2} \\ u_2^{\beta_2} - u_2^{\alpha_2} \end{pmatrix} = \underline{\underline{A}}_2^{\xi} \underline{\tilde{u}}_h \tag{A17}$$

DOI: 10.1002/nme

Finally, doing the necessary algebra, (A16) results in the following expression:

$$(\tilde{\underline{u}}_h)^{\mathrm{T}} \underbrace{((\underline{\underline{A}}_1^{\xi})^{\mathrm{T}}(\underline{\underline{A}}_2^{\xi})^{\mathrm{T}}) \begin{pmatrix} (\underline{\underline{B}}_{11}^{\xi})^{\mathrm{T}} & (\underline{\underline{B}}_{12}^{\xi})^{\mathrm{T}} \\ (\underline{\underline{B}}_{12}^{\xi})^{\mathrm{T}} & (\underline{\underline{B}}_{22}^{\xi})^{\mathrm{T}} \end{pmatrix}}_{\underline{\underline{A}}_{\xi}^{\mathrm{eq2}}} \begin{pmatrix} t_{1'}^{1,\xi} \\ t_{2'}^{1,\xi} \\ t_{1'}^{2,\xi} \\ t_{2'}^{2,\xi} \end{pmatrix} = (\tilde{\underline{u}}_h)^{\mathrm{T}} \underline{\underline{A}}_{\xi}^{\mathrm{eq2}} \underline{t}_h^{\xi} \tag{A18}$$

Expression (A18) gives the contribution of the edge $\xi_e^{e'}$ to term (2). The last step is to sum over all the internal edges as follows:

$$\sum_{\xi_e^{e'} \in \mathscr{E}^{\mathscr{O}}} (\tilde{\underline{u}}_h)^{\mathrm{T}} \underline{\underline{A}}_{\xi_e^{e'}}^{\mathrm{eq2}} \underline{t}_h^{\xi_e^{e'}} = (\tilde{\underline{u}}_h)^{\mathrm{T}} \underline{\underline{\tilde{A}}}^{\mathrm{eq2}} \underline{t}_h \tag{A19}$$

where $\underline{\underline{\tilde{A}}}^{\mathrm{eq2}}$ is a $(6 \times E, 4 \times |\mathscr{E}^{\mathscr{O}}|)$ global matrix and $\underline{t}_h$ is a $4 \times |\mathscr{E}^{\mathscr{O}}|$ vector collecting the nodal tractions for each interior edge.

Finally, considering (A11), (A12) and (A19), the discretization of the equilibrium constraint (A10) takes the following form:

$$(\tilde{\underline{u}}_h)^{\mathrm{T}} \underline{\underline{\tilde{A}}}^{\mathrm{eq1}} \tilde{\underline{\sigma}}_h + (\tilde{\underline{u}}_h)^{\mathrm{T}} \underline{\underline{\tilde{A}}}^{\mathrm{eq2}} \underline{t}_h = \lambda (\tilde{\underline{u}}_h)^{\mathrm{T}} \underline{\tilde{F}}_h^{\mathrm{eq}} \quad \forall \tilde{\underline{u}}_h \in Y_h^{\mathrm{UB}} \tag{A20}$$

Since (A20) must hold for all $\mathbf{u}_h$ in Y_h^{UB}, the displacement vector can be removed from the equation, thereby obtaining

$$\underline{\underline{\tilde{A}}}^{\mathrm{eq1}} \tilde{\underline{\sigma}}_h + \underline{\underline{\tilde{A}}}^{\mathrm{eq2}} \underline{t}_h = \lambda \underline{\tilde{F}}_h^{\mathrm{eq}} \tag{A21}$$

ACKNOWLEDGEMENTS

The authors acknowledge the financial support of the Singapore-MIT Alliance and DARPA-AFOSR, through grant F49620-03-1-0439, which made the work presented in this article possible.

REFERENCES

1. Belytschko T, Hodge PG. Plane stress limit analysis by finite elements. *Journal of the Engineering Mechanics Division* (ASCE) 1970; **96**:931–944.
2. Pastor J, Turgeman S. Mise en oeuvre numérique des méthodes de l'analyse limite pour les matériaux de von Mises et de Coulomb standards en déformation plane. *Mechanics Research Communications* 1976; **3**:469–476.
3. Lysmer J. Limit analysis of plane problems in soil mechanics. *Journal of the Soil Mechanics and Foundations Division* (ASCE) 1970; **96**:1311–1334.
4. Christiansen E. Computation of limit loads. *International Journal for Numerical Methods in Engineering* 1981; **17**:1547–1570.
5. Maier G. A matrix structural theory of piecewise-linear plasticity with interacting yield planes. *Meccanica* 1970; **5**:55–66.

DOI: 10.1002/nme

6. Anderheggen E, Knöpfel H. Finite element limit analysis using linear programming. *International Journal of Solids and Structures* 1972; **8**:1413–1431.
7. Capurso M. Limit analysis of continuous media with piecewise linear yield condition. *Meccanica—Journal of the Italian Association for Theoretical and Applied Mechanics* 1971; **6**:53–58.
8. Andersen KD, Christiansen E. Limit analysis with the dual affine scaling algorithm. *Journal of Computational and Applied Mathematics* 1995; **59**:233–243.
9. Christiansen E, Kortanek KO. Computation of the collapse state in limit analysis using the LP primal affine scaling algorithm. *Journal of Computational and Applied Mathematics* 1991; **34**:47–63.
10. Andersen KD, Christiansen E, Overton ML. Computing limit loads by minimizing a sum of norms. *SIAM Journal on Scientific Computing* 1998; **19**(3):1046–1062.
11. Christiansen E, Andersen KD. Computation of collapse states with von Mises type yield condition. *International Journal for Numerical Methods in Engineering* 1999; **46**:1185–1202.
12. Andersen KD, Christiansen E. Minimizing a sum of norms subject to linear equality constraints. *Computational Optimization and Applications* 1998; **11**:65–79.
13. Christiansen E, Pedersen OS. Automatic mesh refinement in limit analysis. *International Journal for Numerical Methods in Engineering* 2001; **50**(6):1331–1346.
14. Andersen KD, Christiansen E, Conn AR, Overton ML. An efficient primal–dual interior-point method for minimizing a sum of Euclidian norms. *SIAM Journal on Scientific Computing* 2000; **22**(1):243–263.
15. Lyamin AV, Sloan SW. Lower bound limit analysis using non-linear programming. *International Journal for Numerical Methods in Engineering* 2002; **55**:573–611.
16. Lyamin AV, Sloan SW. Upper bound limit analysis using linear finite elements and non-linear programming. *International Journal for Numerical and Analytical Methods in Geomechanics* 2002; **26**:181–216.
17. Krabbenhoft K, Damkilde L. A general non-linear optimization algorithm for lower bound limit analysis. *International Journal for Numerical Methods in Engineering* 2003; **56**:165–184.
18. Krabbenhoft K, Lyamin AV, Hjiad MhD, Sloan SW. A new discontinuous upper bound formulation. *International Journal for Numerical Methods in Engineering* 2005; **63**:1069–1088.
19. Makrodimopoulos A, Martin CM. Lower bound limit analysis of cohesive-frictional materials using second order programming. *International Journal for Numerical Methods in Engineering* 2006; **66**(4):604–634.
20. Makrodimopoulos A, Martin CM. Upper bound limit analysis of cohesive-frictional materials using simplex strain elements and second order programming. *International Journal of Numerical and Analytical Methods in Geomechanics* 2007; **31**:835–865.
21. Makrodimopoulos A. Computational formulation of shakedown analysis as a conic quadratic optimization problem. *Mechanics Research Communications* 2006; **33**:72–83.
22. Borges L, Zouain N, Costa C, Feijoo R. An adaptive approach to limit analysis. *International Journal of Solids and Structures* 2001; **38**(10–13):1707–1720.
23. Ciria H, Peraire J. Computation of upper and lower bounds in limit analysis using second-order cone programming and mesh adaptivity. *Ninth ASCE Special Conference on Probabilistic Mechanics and Structural Reliability*, Albuquerque, 2004.
24. Ciria H. Computation of upper and lower bounds in limit analysis using second-order cone programming and mesh adaptivity. *M.S. Thesis*, Massachusetts Institute of Technology, Cambridge, MA, 2004.
25. Christiansen E. Limit analysis of collapse states. In *Handbook of Numerical Analysis*, vol. 4, Ciarlet PG, Lions JL (eds). North-Holland: Amsterdam, 1996; 193–312.
26. Ben-Tal A, Nemirovski A. *Lectures on Modern Convex Optimization*: *Analysis*, *Algorithms and Engineering Applications*. SIAM: Philadelphia, PA, 2001.
27. Nesterov Y, Nemirovskii A. *Interior Point Polynomial Algorithms in Convex Programming*. SIAM: Philadelphia, PA, 1994.
28. Alizadeh F, Haeberly JA, Overton M. A new primal–dual interior point method for semidefinite programming. In *Proceedings of the Fifth SIAM Conference on Applied Linear Algebra*, Lewis JG (ed.). SIAM: Philadelphia, PA, 1994; 113–117.
29. Helmberg C, Rendl F, Vanderbei RJ, Wolkowicz H. An interior-point method for semidefinite programming. *SIAM Journal on Optimization* 1996; **6**:342–361.
30. Nesterov Y, Todd MJ. Self-scaled barriers and interior point methods for convex programming. *Mathematics of Operations Research* 1997; **22**(1):1–42.
31. Nagtegaal JC, Parks DM, Rice JR. On numerical accurate finite element solutions in the fully plastic range. *Computer Methods in Applied Mechanics and Engineering* 1977; **4**:153–177.

DOI: 10.1002/nme

32. Sturm JF. Using SeDuMi 1.02, a Matlab toolbox for optimization over symmetric cones (updated for version 1.05), 2001. Available from: http://fewcal.kub.nl/∼sturm (April 2005).
33. Tütüncü RH, Toh KC, Todd MJ. SDPT3—a MATLAB software package for semidefinite-quadratic-linear programming, version 3.0, 2001. Available from: http://www.math.cmu.edu/∼reha/sdpt3.html (April 2005).
34. Lubliner J. *Plasticity Theory*. Macmillan Publishing Company: New York, 1990.
35. Rivara MC. Algorithms for refining triangular grids suitable for adaptive and multigrid techniques. *International Journal for Numerical Methods in Engineering* 1984; **20**:745–756.

DOI: 10.1002/nme