



Hugo Bastos de Sá Bruno

**Return-Mapping Algorithms for Associative
Plasticity Using Conic Optimization**

Tese de Doutorado

Thesis presented to the Programa de Pós-graduação em Engenharia Civil of PUC-Rio in partial fulfillment of the requirements for the degree of Doutor em Engenharia Civil.

Advisor : Prof. Luiz Fernando Campos Ramos Martha
Co-advisor: Prof. Ivan Fábio Mota de Menezes

Rio de Janeiro
February 2020



Hugo Bastos de Sá Bruno

**Return-Mapping Algorithms for Associative
Plasticity Using Conic Optimization**

Thesis presented to the Programa de Pós-graduação em Engenharia Civil of PUC-Rio in partial fulfillment of the requirements for the degree of Doutor em Engenharia Civil. Approved by the undersigned Examination Committee.

Prof. Luiz Fernando Campos Ramos Martha

Advisor

Departamento de Engenharia Civil e Ambiental – PUC-Rio

Prof. Ivan Fábio Mota de Menezes

Co-advisor

Departamento de Engenharia Mecânica – PUC-Rio

Prof. Eurípedes do Amaral Vargas

Departamento de Engenharia Civil e Ambiental – PUC-Rio

Prof. Anderson Pereira

Departamento de Engenharia Mecânica – PUC-Rio

Prof. Rodrigo Bird Burgos

Universidade Estadual do Rio de Janeiro – UERJ

Prof. André Maués Brabo Pereira

Universidade Federal Fluminense – UFF

Rio de Janeiro, February the 14th, 2020

All rights reserved.

Hugo Bastos de Sá Bruno

Graduated in Civil Engineering from Universidade Federal Fluminense - UFF in 2014 and obtained his master degree from Pontifícia Universidade Católica - PUC-Rio in 2017.

Bibliographic data

Bruno, Hugo Bastos de Sá

Return-Mapping Algorithms for Associative Plasticity Using Conic Optimization / Hugo Bastos de Sá Bruno; advisor: Luiz Fernando Campos Ramos Martha; co-advisor: Ivan Fábio Mota de Menezes. – Rio de Janeiro: PUC-Rio, Departamento de Engenharia Civil e Ambiental, 2020.

v., 103 f: il. color. ; 30 cm

Tese (doutorado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Engenharia Civil e Ambiental.

Inclui bibliografia

1. Engenharia Civil e Ambiental – Teses. 2. Algoritmos de Retorno à Superfície. 3. Análise Elastoplástica. 4. Equações algébricas-diferenciais. 5. Programação Cônica. 6. Derivadas paramétricas de primeira ordem. I. Martha, Luiz Fernando Campos Ramos. II. Menezes, Ivan Fábio Mota. III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Engenharia Civil e Ambiental. IV. Título.

CDD: 624

Acknowledgments

To professors Luiz Fernando Martha and Ivan Fábio Mota de Menezes for the enlightening discussions and contributions.

To my dear friend and colleague Guilherme Barros for the fruitful and entertaining deliberations.

To CNPq and PUC-Rio, for the financial support, without which this work could not have been carried out.

To my family for all the support and caring.

Abstract

Bruno, Hugo Bastos de Sá ; Martha, Luiz Fernando Campos Ramos (Advisor); Menezes, Ivan Fábio Mota (Co-Advisor). **Return-Mapping Algorithms for Associative Plasticity Using Conic Optimization**. Rio de Janeiro, 2020. 103p. Tese de doutorado – Departamento de Engenharia Civil e Ambiental, Pontifícia Universidade Católica do Rio de Janeiro.

This work presents a mathematical programming approach for elastoplastic constitutive initial boundary value problems. Considering associative plasticity, the local discrete constitutive equations are formulated as conic programs. Specifically, it is demonstrated that *implicit* return-mapping schemes for well-known yield criteria, such as the Rankine, von Mises, Tresca, Drucker-Prager, and Mohr-Coulomb criteria, can be expressed as second-order and semidefinite conic programs. Additionally, a novel scheme for the numerical evaluation of the consistent elastoplastic tangent operator, based on a first-order parameter derivative of the optimal solutions, is proposed.

Keywords

Return-Mapping Algorithms; Elastoplastic Analysis; Differential-Algebraic Equations; Conic Programming; First-Order Parameter Derivatives.

Resumo

Bruno, Hugo Bastos de Sá ; Martha, Luiz Fernando Campos Ramos; Menezes, Ivan Fábio Mota. **Algoritmos de Retorno à Superfície para Plasticidade Associativa Utilizando Programação Cônica**. Rio de Janeiro, 2020. 103p. Tese de Doutorado – Departamento de Engenharia Civil e Ambiental, Pontifícia Universidade Católica do Rio de Janeiro.

Esse trabalho apresenta uma abordagem baseada em programação matemática para a solução de problemas de valor inicial de contorno constitutivo elastoplástico. Considerando a plasticidade associativa, as equações constitutivas locais, em sua forma discreta, são formuladas como problemas de otimização cônica. Especificamente, é demonstrado que métodos implícitos de retorno a superfície para os critérios mais conhecidos da literatura, como o de Rankine, von Mises, Tresca, Drucker-Prager e Mohr Coulomb, podem ser expressos como problemas de otimização cônica de segunda ordem e semidefinida. Além disso, um novo método numérico para a determinação do operador elastoplástico consistente, baseado na derivada paramétrica de primeira ordem das soluções ótimas, é proposto.

Palavras-chave

Algoritmos de Retorno à Superfície; Análise Elastoplástica; Equações algébricas-diferenciais; Programação Cônica; Derivadas paramétricas de primeira ordem.

Table of contents

1	Introduction	11
1.1	Scope and aim	13
1.2	Literature review	14
1.3	Organization	16
2	Conic programming	18
2.1	Linear Programming	20
2.2	Second-Order Conic Programming	21
2.3	Semidefinite Programming	22
2.4	First-order parameter derivatives of conic programs	24
2.5	Computational aspects	26
3	The associative plasticity model	28
3.1	Material yield criteria	28
3.2	Flow rules	34
3.3	Principle of maximum plastic dissipation	36
3.4	Hardening models	37
3.5	Numerical integration of elastoplastic constitutive equations	40
4	The finite element method in nonlinear solid mechanics	43
4.1	The incremental initial value problem	43
4.2	Finite element discretization	45
4.3	The nonlinear solution scheme	48
4.4	Computational implementation	49
5	Return-mapping algorithms using conic programming	59
5.1	Consistent elastoplastic tangent operator numerical evaluation scheme	61
6	Numerical examples	67
6.1	Finite step accuracy and iso-error maps	67
6.2	Boundary value problems	72
7	Conclusions and future work	78
7.1	Conclusions	78
7.2	Future work	79
	Bibliography	84
	Appendix A	92
	Appendix B	93
	Appendix C	94

List of figures

Figure 2.1	A proper cone $\mathcal{K} \subset \mathbb{R}^2$.	18
Figure 2.2	The dual cone \mathcal{K}^* to \mathcal{K} .	19
Figure 2.3	Conic programming subfields.	20
Figure 3.1	Rankine yield surface in the space of principal stresses.	30
Figure 3.2	Tresca yield surface in the space of principal stresses.	30
Figure 3.3	Mohr-Coulomb yield surface in the space of principal stresses.	31
Figure 3.4	von Mises yield surface in the space of principal stresses.	32
Figure 3.5	Drucker-Prager yield surface in the space of principal stresses.	33
Figure 4.1	Schematic representation of a general initial BVP.	44
Figure 4.2	FEM discretization mesh of domain Ω .	45
Figure 4.3	A linear shape function.	46
Figure 4.4	Global interpolation.	46
Figure 4.5	Displacement control update routine	54
Figure 4.6	Main procedure flowchart	55
Figure 6.1	Stress increment directions for the Drucker-Prager model.	68
Figure 6.2	<i>Iso-error map</i> for the Drucker-Prager model.	69
Figure 6.3	Stress increment directions for the Mohr-Coulomb model.	70
Figure 6.4	<i>Iso-error map</i> for the Mohr-Coulomb model departing from point A.	70
Figure 6.5	<i>Iso-error map</i> for the Mohr-Coulomb model departing from point B.	71
Figure 6.6	<i>Iso-error map</i> for the Mohr-Coulomb model departing from point C.	71
Figure 6.7	Strip-footing finite-element model.	73
Figure 6.8	Strip-footing load-displacement curve.	74
Figure 6.9	End-loaded cantilever finite-element model.	74
Figure 6.10	End-loaded cantilever load-displacement curve.	75
Figure 6.11	Perforated plate finite-element model.	76
Figure 6.12	Perforated plate load-displacement curve.	76
Figure 6.13	Perforated plate plastic strain evolution.	77
Figure 7.1	Perforated circular plate under internal pressure.	94
Figure 7.2	Perforated circular plate example load-displacement curve.	103

There is no other way around hard word, embrace it.

Roger Federer, *Lessons from The Court.*

List of acronyms

MP *mathematical programming*

RMA *return-mapping algorithms*

FEM *finite element method*

RC *reinforced concrete*

PMPD *principle of maximum plastic dissipation*

SOCCP *second-order cone complementarity problem*

QP *quadratic programming*

KKT *Karush-Kuhn-Tucker*

CP *conic programming*

IVP *initial value problem*

CETO *consistent elastoplastic tangent operator*

SOCP *second-order conic programming*

SDP *semidefinite programming*

LP *linear programming*

BVP *boundary value problem*

DAE *differential-algebraic equations*

N-R *Newton-Raphson*

1

Introduction

The *theory of plasticity* involves the study of bodies which are stressed beyond the limit of the elastic range and, consequently, undergo permanent/plastic deformations. Thus, it may be regarded as an extension of the *theory of elasticity*, which deals with structures subject to stresses under the elastic limit, i.e., structures which can carry the load program without suffering plastic deformations.

According to Chen & Han [1], the mathematical foundations of plasticity dates back to the beginning of the 19th century, when Tresca first published a series of papers proposing the conditions which dictates the emergence of plastic deformations in metals, the so-called *maximum shear stress* condition. Later on, around 1870, St. Venant would then devise the mathematical foundations of plasticity which described the *constitutive relations* for *elastoplastic* materials under *plane stress* and, in that same year, Levy extended such formulation for the general 3D stress state. In 1913 von Mises published his famous paper on the *J2*-theory for metals, which included an *yield condition* based on the *maximum distortion energy*. The *constitutive equations*, developed by St. Venan, Levy, and von Mises, were extended to the plane continuum case by Prandtl, in 1924. The generalization to three dimensions was carried out by Reuss, in 1930. Von Mises, in 1928, published a paper which discussed the relation between the direction of plastic strain and *smooth* yield surfaces, thus introducing the *flow rule* concept for *regular* yield criteria. In 1933, Reuss published studies in which appropriate *flow rules* for *singular* yield surfaces, such as Tresca's, were formulated. The development of incremental *constitutive equations* for materials with hardening behavior was initiated by Prandtl, in 1928. A few years later, in 1938, Melan devised incremental relations for hardening materials with *smooth* yield surfaces. Independently, in 1949, Prager published a study proposing a general framework for elastoplastic constitutive relations involving materials with *smooth* yield surfaces. In this remarkable work, Prager also formulated the *loading/unloading* conditions, i.e., the conditions which established when plastic flow may occur. Additionally, Drucker devised *uniqueness* theorems for the solution of elastoplastic *boundary value problems* (BVPs) and recognized the interrelationship between convexity of

yield surfaces and the *normality* condition.

Fundamental theorems of *limit analysis*, the so-called *upper-* and *lower-bound theorems*, were stated in papers published by Drucker [2], Greenberg [3] and Prager [4], in 1951 and 1952, under the assumption of elastic-perfectly plastic behavior. Under the hypothesis of rigid-ideally plastic materials, Hill [5] developed similar theorems for the evaluation of load bearing capacity of elastoplastic structures. The application of these theorems for the analysis of beams, frames, plates and shells were carried out in the following years, including the analysis of reinforced concrete and metallic structures.

The generalization of elastoplastic constitutive relations for *singular* yield surfaces was conducted by Koiter [6], back in 1953. His contribution allowed the employment of multiples functions for the determination of general yield criteria. His studies showed that, in the case of *singular* yield surfaces, the plastic increment may be expressed as the combination of contributions from each *active* yield surface.

The growing interest from industry in simulation of solid mechanics involving elastoplastic materials has led to a rapid and extensive development of numerical tools for the solution of practical problems in plasticity. These techniques are currently applied to a wide variety of engineering problems, ranging from stress analysis in structures and soil mechanics, to the simulation of manufacturing processes such as metal forming. According to Souza Neto et al. [7], other less conventional applications includes food processing, mining operations and simulation of biological tissue behavior. At the present time of this thesis, the *finite element method* (FEM) is arguably the most commonly adopted numerical scheme in non-linear quasi-static solid mechanics. The computational techniques, developed so far, are capable of accurately predict elastoplastic phenomena of real life applications. Problems involving extreme large strains and complex constitutive models are now amenable to be solved in a routinely manner.

An emerging area of study on this subject regards the application of *mathematical programming* (MP) techniques in the formulation and numerical solutions of elastoplastic problems. From a theoretical point of view, the mathematical foundations of MP have been recognized as a suitable tool in introducing the formulation of the *theory of plasticity* [8]. Furthermore, the development of modern *numerical optimization* algorithms has provided attractive alternatives for the construction of new and efficient numerical techniques in elastoplastic analysis. In particular, *conic programming* (CP) has been successfully applied in a number of recent works involving applications of *limit* [9], *shakedown* [10] and incremental elastoplastic analyses [11]. Although

many studies have been published in this area, the implementation of incremental elastoplastic analysis using CP is still rare, thus it deserves its own attention.

1.1

Scope and aim

This work investigates the application of MP techniques for rate-independent elastoplastic analyses of structures. MP formulations of 2D/3D continuum structures considering complex constitutive models are introduced within the framework of the FEM. Specifically, the development of a numerical framework for the solution of the local constitutive *initial value problem* (IVP) based on CP is introduced. Although such scheme is restricted to *associative plasticity*, it is general in the sense that singular and multisurface yield criteria can be taken into account in a straightforward manner and it also allows the consideration of both isotropic and kinematic nonlinear hardening models. Additionally, it is shown that the *consistent elastoplastic tangent operator* (CETO) may be numerically evaluated according to a first-order parameter derivative of the corresponding CP problems' solutions. Compared with traditional *return-mapping algorithms* (RMA), such numerical scheme for the evaluation of the CETO relieves the burden of obtaining symbolic derivatives of the nonlinear elastoplastic constitutive equations, thus simplifying the implementation of complex models, specially if non-smooth yield functions are considered. Additionally, by simply suppressing the corresponding stress components in the optimization problem formulation, the proposed framework is able to handle the case of constrained stress states in a straightforward manner. It is worth to note that for traditional RMA the simple case of *plane stress* requires specific designed algorithms which handles the stress constraint as an additional restriction to the elastoplastic equations [7].

Objectives

The main objectives of this thesis are:

- Develop FEM codes for the elastoplastic analysis of 2D/3D continuum problems considering various constrained stress states. The main feature of the proposed implementation is the employment of CP algorithms for the solution of *local constitutive elastoplastic* IVPs and the evaluation of CETOs as a first-order parameter derivative of the optimal solutions.

- Provide a dictionary of transformation of complex yield criteria into conic inequalities, thus allowing the *local constitutive* IVPs to be solved as CP problems.
- Assess the accuracy and robustness of the proposed codes by simulation of elastoplastic benchmark problems commonly found in the literature.

1.2

Literature review

The early works on MP approaches for incremental elastoplastic analysis encompass frameworks based on general *nonlinear programming* such as *quadratic* [12], *sequential quadratic* [13], and *convex programming* [14]. One of the pioneers works on this subject proposed the employment of MP techniques of the elastoplastic analysis of structural slabs [15]. A notable contribution to this area is attributed to Simo & Taylor [16] which presented a consistent variational formulation for *associative plasticity* based on a complementary functional involving the Lagrangian potential that is related to the *principle of maximum plastic dissipation* (PMPD). Based on this result, and by employing a mixed FEM formulation, the global equilibrium equations are reduced to a convex MP problem in which the consistency condition and hardening law are enforced at a point-wise level, whereas the flow rule is enforced in a weak sense.

Following this variational formulation, in [17] Krabbenhoft et al. propose a *primal-dual interior-point* algorithm for the solution of rate-independent elastoplasticity problems considering small deformations and in [11] the same authors present the formulation and solution of some plasticity problems as conic programs. In another remarkable work [12], Maier demonstrates two extrema theorem which relate the incremental elastoplastic analysis to a *quadratic programming* (QP) problem, under the *normality* condition. The first theorem expresses each increment of the analysis as a minimum optimization problem in which the variables are given by the stresses and plastic multipliers. Alternatively, the second theorem express each increment as a maximum optimization problem in which the variables are comprised of the displacements and plastic multipliers. Both theorems are presented in continuum and discretized versions and both rely on the introduction of a quadratic functional for which the *Karush-Kuhn-Tucker* (KKT) conditions leads to the enforcement of equilibrium; compatibility; and the elastoplastic constitutive equations. However, these theorems only hold for yield criteria which are given as the intersection of affine functions. Therefore, in order to model constitutive laws based on nonlinear yield surfaces an approximation is required. Follow-

ing this idea, Pereira et al. [18] introduce a force method approach for the elastoplastic analysis for frames using QP.

An alternative approach is presented in [19] in which Christensen exposes an MP approach for the solution of J_2 -plasticity problems with linear isotropic/kinematic hardening models. Based on the PMPD, the constitutive law is restated as a set of non-smooth equations. By considering both the displacements and deviatoric stresses as variables, the global nonlinear equations are reformulated as a system of unconstrained non-smooth equations which enforce both equilibrium and constitutive equations in a strong form. The solution of such system is sought by means of a non-smooth Newton method [20]. Alternatively, in [21] the local elastoplastic equations are sought as the solution of a MP problem related to a discrete version of the PMPD. The numerical solution of such MP problems is conducted by general nonlinear optimization methods which unfortunately impairs the consideration of singular yield criteria.

In a recent study [22], Zhang et al. propose an MP approach towards the solution of rate-independent elastoplasticity in small strains by formulating the nonlinear equilibrium equations as *cone complementary* problems. It is shown that problems involving von Mises and Drucker-Prager criteria with non-associative flow rules and linear isotropic-kinematic hardening models may be formulated as *second-order cone complementarity problems* (SOCCPs). The local constitutive equations are enforced in a weak sense by discretizing the plastic multiplier vectors within each element. In light of such discretization, the global nonlinear equations of equilibrium are reformulated as an SOCCP for which the solution is sought by means of a semi-smooth Newton algorithm [23].

In the recent years, CP has been acknowledged as a powerful method for the solution of elastoplastic constitutive equations via the MP approach [14, 24]. The suitability of CP programming stems from the fact that many of the yield criteria used in real-world simulations are amenable to be transformed into conic constraints. For instance, in [25] Bisbos et al. present the cast of numerous yield criteria into second-order and semidefinite conic constraints. Moreover, recent developments on numerical methods for *second-order conic programming* (SOCP) [26] and *semidefinite programming* (SDP) [27] have provided efficient and robust algorithms for the solution of CP problems. Very recently, the extension of CP to the case of the *exponential* and *power cones* [28] have been proposed and implemented in several optimization codes [29], [30]. Although such improvement allows to enlarge the range of yield criteria which may be represented as conic constraint, it appears that such possibility has not been investigated in the literature yet.

Arguably, the majority of applications of CP in plasticity are related to *limit* [31, 32, 33, 34, 9, 35, 36, 37] and *shakedown analyses* [38, 10, 39]. However, many works involving the application of SOCP and SDP for incremental elastoplastic analysis have been published in recent years. For instance, following the variational foundations presented in [16], a general formulation of incremental elastoplastic analysis within the framework of CP was presented in [14] for a perfectly plastic model. With similar considerations, Yonekura & Kanno [24] restated each step of the elastoplastic analysis as an SOCP problem considering the von Mises criterion with both kinematic and isotropic linear hardening behaviors. It is important to stress out that the main applications of MP techniques in plasticity problems consists in formulating the nonlinear global equilibrium equations as a single optimization problem. Although this concept leads to a highly efficient and robust method, the consideration of large displacement and finite strains is often overlooked. Alternatively, the approach proposed in this work is based on the solution of *local elastoplastic constitutive equations* by means of CP algorithms, thus in the global sense it is similar to traditional state-update procedures. Therefore, just like in conventional elastoplastic FEM approaches, the proposed scheme allows large displacements as well as finite strains to be implemented in a straightforward fashion.

1.3 Organization

This thesis is organized as follows. Chapter 2 gives a brief exposition of CP and introduces the main properties of SOCP and SDP, including its first-order parameter derivative formulations. Chapter 3 introduces the continuum constitutive equations of associative plasticity as a consequence of the PMPD in its rate form. Based on this derivation, the equivalence between the incremental form of the local constitutive equations and the KKT conditions of a discrete counterpart of the PMPD is established. Chapter 4 introduces the finite element discretization employed in the formulation of the incremental discrete BVP in nonlinear solid mechanics and briefly presents the *Newton-Raphson* (N-R) scheme for the solution of the nonlinear equilibrium equations. In Chapter 5 it is shown that a complete state-update procedure for associative plasticity may be formulated as a CP problem. Based on a first-order parameter derivatives of such CP problems, a novel numerical scheme for the evaluation of CETOs is also presented. The accuracy of the proposed approach is assessed in Chapter 6 by constructing iso-error maps for different elastoplastic models and comparing the results to traditional RMA

codes. Additionally, numerical experiments are conducted to investigate the robustness and efficiency of the proposed methodology in solving benchmark BVPs commonly found in the literature. Conclusions are drawn in Chapter 7 along with the proposal of future work and developments.

2 Conic programming

CP constitutes the class of MP problems in which one seeks to optimize a linear objective function subjected to a constraint set given as the intersection of an affine subspace with a proper cone (convex, pointed, closed, and with a nonempty interior). A general CP problem may be written as

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & \mathbf{Ax} = \mathbf{b} , \\ & \mathbf{x} \in \mathcal{K} \end{aligned} \tag{2-1}$$

where $\mathbf{c} \in \mathbb{R}^n$ denotes the objective vector, $\mathbf{x} \in \mathbb{R}^n$ represents the design variables, $\mathbf{Ax} = \mathbf{b}$ is a $m \times n$ linear system of equations, and \mathcal{K} is a proper cone, as depicted in Figure 2.1.

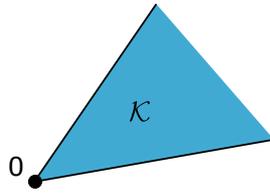


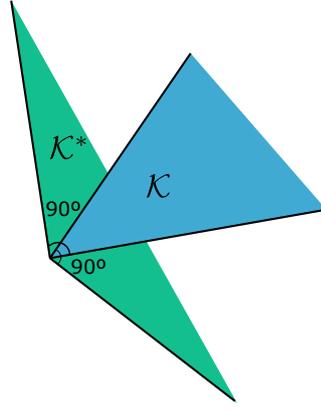
Figure 2.1: A proper cone $\mathcal{K} \subset \mathbb{R}^2$.

Owing to the partial ordering properties of proper cones [40], it is possible to develop a *conic duality* theory that closely connects CP to *linear programming* (LP). In the conic case, the *dual* problem to (2-1) is given as

$$\begin{aligned} \max \quad & \mathbf{b}^T \mathbf{y} \\ \text{s.t.} \quad & \mathbf{A}^T \mathbf{y} + \mathbf{z} = \mathbf{c} , \\ & \mathbf{z} \in \mathcal{K}^* \end{aligned} \tag{2-2}$$

where $\mathbf{y} \in \mathbb{R}^m$ and $\mathbf{z} \in \mathbb{R}^n$ are the *dual* variables, and \mathcal{K}^* is the dual cone to \mathcal{K} , i.e.,

$$\mathcal{K}^* = \{ \mathbf{z} \mid \langle \mathbf{z}, \mathbf{x} \rangle \geq 0 \quad \forall \mathbf{x} \in \mathcal{K} \}. \tag{2-3}$$

Figure 2.2: The dual cone \mathcal{K}^* to \mathcal{K} .

By construction of the dual problem (2-2), the following theorem holds in CP.

Theorem 2.1 (Weak Duality Theorem) *Given any dual feasible solution (\mathbf{y}, \mathbf{z}) to (2-2) and any primal feasible solution \mathbf{x} to (2-1), then*

$$\mathbf{c}^T \mathbf{x} - \mathbf{b}^T \mathbf{y} \geq 0. \quad (2-4)$$

In other words, the optimal value to (2-2) is a lower bound to the optimal value of (2-1), exactly as in the LP case. However, owing to the nonlinear geometry of general CP problems, a word-by-word extension the LP *strong duality theorem* is not possible. Nonetheless, the following slightly weaker statement holds for CP.

Theorem 2.2 (Strong Duality Theorem) *If both primal (2-1) and dual (2-2) problems have strictly feasible solutions ($\mathbf{x} \in \text{int } \mathcal{K}, \mathbf{z} \in \text{int } \mathcal{K}^*$), then they present optimal solutions \mathbf{x}^* and $(\mathbf{y}^*, \mathbf{z}^*)$, respectively, and attain the same optimal value, i.e.,*

$$\mathbf{c}^T \mathbf{x}^* = \mathbf{b}^T \mathbf{y}^* \implies \mathbf{x}^{*T} \mathbf{z}^* = 0, \quad (2-5)$$

where the value $\mathbf{x}^T \mathbf{z}$ is the so-called duality gap which follows from the same identity as in LP, i.e.,

$$\mathbf{c}^T \mathbf{x} - \mathbf{b}^T \mathbf{y} = (\mathbf{y}^T \mathbf{A} + \mathbf{z}^T) \mathbf{x} - \mathbf{x}^T \mathbf{A}^T \mathbf{y} = \mathbf{x}^T \mathbf{z}. \quad (2-6)$$

Owing to Theorem 2.2, it is straightforward to obtain the optimality conditions for a general CP problem, i.e.,

$$\begin{aligned}
\mathbf{Ax} &= \mathbf{b}, \quad \mathbf{x} \in \mathcal{K} \\
\mathbf{A}^T \mathbf{y} + \mathbf{z} &= \mathbf{c}, \quad \mathbf{z} \in \mathcal{K}^*, \\
\mathbf{x}^T \mathbf{z} &= 0
\end{aligned} \tag{2-7}$$

where the *duality gap* vanishing condition, expressed in the last equation, is known as the *complementary slackness condition*.

LP, *convex* QP, SOCP and SDP are all instances of CP. Each of these class of problems are distinguished by the underlying structure of the conic constraint associated with the problem. Figure 2.3 depicts the hierarchy between such subfields of CP.

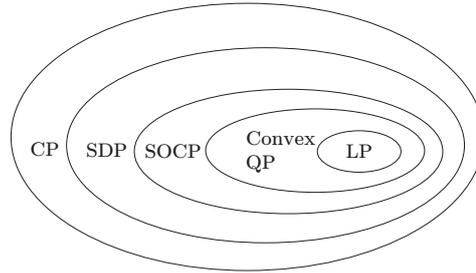


Figure 2.3: Conic programming subfields.

In the following, a brief exposition of LP within CP theory is presented. Such introduction is intended to show the similarities between LP and more general CP problems, in particular SOCP and SDP.

2.1 Linear Programming

A general LP problem, in its standard form, may be expressed in the following primal and dual versions

$$\begin{aligned}
\min \quad & \mathbf{c}^T \mathbf{x} & \max \quad & \mathbf{b}^T \mathbf{y} \\
\text{s.t.} \quad & \mathbf{Ax} = \mathbf{b} & \text{s.t.} \quad & \mathbf{A}^T \mathbf{y} + \mathbf{z} = \mathbf{c} \\
& \mathbf{x} \geq 0 & & \mathbf{z} \geq 0
\end{aligned} \tag{2-8}$$

The resemblance between (2-8) and (2.3) is clear, the only difference given by the inequality instead of the conic constraint. However, by taking the \mathbb{R}_+ cone into account, i.e.,

$$\mathbb{R}_+ = \{t \in \mathbb{R} \mid t \geq 0\}, \tag{2-9}$$

along with its dual

$$\mathbb{R}_+^* = \mathbb{R}_+ \tag{2-10}$$

it is straightforward to express (2-8) as a CP problem, i.e.,

$$\begin{array}{ll}
\min & \mathbf{c}^T \mathbf{x} \\
\text{s.t.} & \mathbf{A} \mathbf{x} = \mathbf{b} \\
& \mathbf{x} \in \mathbb{R}_+^n
\end{array}
\quad
\begin{array}{ll}
\max & \mathbf{b}^T \mathbf{y} \\
\text{s.t.} & \mathbf{A}^T \mathbf{y} + \mathbf{z} = \mathbf{c} \\
& \mathbf{z} \in \mathbb{R}_+^n
\end{array}
. \quad (2-11)$$

An important consequence of the self-duality of \mathbb{R}_+ leads to the following lemma.

Lemma 1 (LP complementary slackness condition) *Let $\mathbf{x}, \mathbf{z} \in \mathbb{R}_+^n$, then the slackness condition holds if and only if $x_i z_i = 0$, i.e., when either $x_i = 0$ or $z_i = 0 \forall i$.*

Proof. The proof follows from the simple statement

$$\mathbf{x}^T \mathbf{z} = \sum_i x_i z_i = 0 \implies x_i z_i = 0 \quad \forall i. \quad (2-12)$$

Otherwise, the sum would be positive. Conversely, if either $x_i = 0$ or $z_i = 0 \forall i$, then, obviously, $\mathbf{x}^T \mathbf{z} = 0$. ■

This equivalent form of the *slackness condition* allows us to rewrite the optimality conditions for an LP problem as an $N \times N^{**}$ system of nonlinear equations constrained by the inclusion conditions $\mathbf{x}, \mathbf{z} \in \mathbb{R}_+$, i.e.,

$$\begin{array}{ll}
\mathbf{A} \mathbf{x} = \mathbf{b}, & \mathbf{x} \in \mathbb{R}_+^n \\
\mathbf{A}^T \mathbf{y} + \mathbf{z} = \mathbf{c}, & \mathbf{z} \in \mathbb{R}_+^n, \\
\mathbf{x}^T \mathbf{z} = \mathbf{X} \mathbf{Z} \mathbf{e} = 0 &
\end{array} \quad (2-13)$$

where $\mathbf{X} = \text{diag}(x_1, \dots, x_n)$, $\mathbf{Z} = \text{diag}(z_1, \dots, z_n)$, and $\mathbf{e}^T = [1, \dots, 1]$.

Although the same optimality conditions can be obtained by simply applying the *first-order necessary conditions theorem* for general nonlinear optimization [41], the development outlined above should shed lights on the close relationship between LP and CP. The same steps made in the development of LP are followed for both SOCP and SDP.

2.2

Second-Order Conic Programming

SOCP constitutes the class of problems in which the conic constraint is given in terms of the second-order cone, also known as the ice-cream or Lorentz cone, defined as

$$\mathcal{C}^{SOC} = \{\mathbf{x} \in \mathbb{R}^n \mid x_1 \geq \|x_{2:n}\|, x_1 \geq 0\}. \quad (2-14)$$

Owing to its self-duality [42], the primal and dual problems corresponding to an SOCP problem are expressed as

** $N = 2n + m$.

$$\begin{array}{ll}
\min & \mathbf{c}^T \mathbf{x} \\
\text{s.t.} & \mathbf{A}^T \mathbf{x} = \mathbf{b} \\
& \mathbf{x} \in \mathcal{C}^{SOC}
\end{array}
\quad
\begin{array}{ll}
\max & \mathbf{b}^T \mathbf{y} \\
\text{s.t.} & \mathbf{A}^T \mathbf{y} + \mathbf{z} = \mathbf{c} \\
& \mathbf{z} \in \mathcal{C}^{SOC}
\end{array}
. \quad (2-15)$$

Likewise the LP case, it is possible to express the *complementary slackness condition* of SOCP problems with an enlarged system of equations, i.e.,

Lemma 2 (SOCP complementary slackness condition) *Let $\mathbf{x}, \mathbf{z} \in \mathcal{C}^{SOC}$, then the slackness condition holds if and only if $Arw(\mathbf{x})Arw(\mathbf{z})\mathbf{e} = \mathbf{0}$, with*

$$Arw(\mathbf{v}) = \begin{bmatrix} v_1 & v_{2:n}^T \\ v_{2:n} & v_1 \mathbf{I}_d \end{bmatrix}, \quad v_{2:n} = \begin{bmatrix} v_2 \\ \vdots \\ v_n \end{bmatrix}, \quad \mathbf{e} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad (2-16)$$

where \mathbf{I}_d is the identity matrix of an appropriate size. Equivalently,

$$\mathbf{x}, \mathbf{z} \in \mathcal{C}^{SOC}, \quad \mathbf{x}^T \mathbf{z} = 0 \iff \begin{array}{l} (i) \quad x_1 z_1 + \mathbf{x}_{2:n}^T \mathbf{z}_{2:n} = 0 \\ (ii) \quad x_1 \mathbf{z}_{2:n} + z_1 \mathbf{x}_{2:n} = \mathbf{0} \end{array}. \quad (2-17)$$

Proof. See Appendix A. ■

Owing to Lemma 2, it is possible to express the optimality conditions of an SOCP problem as

$$\begin{array}{ll}
\mathbf{A}\mathbf{x} = \mathbf{b}, & \mathbf{x} \in \mathcal{C}^{SOC} \\
\mathbf{A}^T \mathbf{y} + \mathbf{z} = \mathbf{c}, & \mathbf{z} \in \mathcal{C}^{SOC}, \\
\mathbf{x}^T \mathbf{z} = \mathbf{X}\mathbf{Z}\mathbf{e} = 0
\end{array} \quad (2-18)$$

where $\mathbf{X} = Arw(\mathbf{x})$, $\mathbf{Z} = Arw(\mathbf{z})$, and $\mathbf{e}^T = [1 \ 0 \ \dots \ 0]$.

2.3

Semidefinite Programming

In the SDP case, the conic constraint is given in terms of the positive semidefinite cone \mathcal{C}^{PSD} , which represents the space of all positive semidefinite matrices, i.e.,

$$\mathcal{C}^{PSD} = \{\mathbf{X} \in \mathcal{S}^n | \mathbf{X} \succeq 0\}, \quad (2-19)$$

where \mathcal{S}^n represents the space of all $n \times n$ real symmetric matrices and \succeq denotes

$$\mathbf{Y} \succeq \mathbf{Z} \iff \mathbf{w}^T (\mathbf{Y} - \mathbf{Z}) \mathbf{w} \geq 0 \quad \forall \mathbf{w} \in \mathbb{R}^n. \quad (2-20)$$

An SDP may thus be expressed as

$$\begin{array}{ll}
\min & \langle \mathbf{C}, \mathbf{X} \rangle \\
\text{s.t.} & \langle \mathbf{A}_i, \mathbf{X} \rangle = b_i \\
& \mathbf{X} \in \mathcal{C}^{PSD}
\end{array}
\quad
\begin{array}{ll}
\max & \mathbf{b}^T \mathbf{y} \\
\text{s.t.} & \sum_{i=1}^m y_i \mathbf{A}_i + \mathbf{Z} = \mathbf{C} \\
& \mathbf{Z} \in \mathcal{C}^{PSD}
\end{array}
, \quad (2-21)$$

where $\langle \cdot, \cdot \rangle$ is the standard inner product of real matrices, i.e.,

$$\langle \mathbf{X}, \mathbf{Y} \rangle = \sum_{i=1}^n \sum_{j=1}^n X_{ij} Y_{ij} = \text{tr}(\mathbf{X}^T \mathbf{Y}), \quad (2-22)$$

and the self-duality of the positive semidefinite cone [43] is implied.

At a first glance, the SDP problem (2-21) does not look like a CP problem as in (2-1). Therefore, in order to express it in a standard form, the same notation employed in [44] is adopted herein, i.e.,

“Let \mathbf{A} be an $m \times n$ matrix. Then, $\text{vec}(\mathbf{A})$ is an mn vector made up of the columns of \mathbf{A} stacked on top of each other. Furthermore, if \mathbf{v} is an mn column vector, $\text{Mat}_{m \times n}(\mathbf{v})$ is an $m \times n$ matrix that collects the components of \mathbf{v} column-wise.”

Based on these definitions, the linear constraints in (2-21) may be restated as

$$\begin{aligned} \langle \mathbf{A}_i, \mathbf{X} \rangle = b_i \quad (i = 1, \dots, m) &\iff \mathcal{A}\mathbf{x} = \mathbf{b} \\ \sum_{i=1}^m y_i \mathbf{A}_i + \mathbf{Z} = \mathbf{C} &\iff \mathcal{A}^T \mathbf{y} + \mathbf{z} = \mathbf{c} \end{aligned}, \quad (2-23)$$

where $\mathbf{x} = \text{vec}(\mathbf{X})$, $\mathbf{z} = \text{vec}(\mathbf{Z})$, $\mathbf{c} = \text{vec}(\mathbf{C})$, and $\mathcal{A} \in \mathbb{R}^{m \times n^2}$ is such that its i th-row is given by $\text{vec}(\mathbf{A}_i)$. Moreover, it is clear that $\langle \mathbf{C}, \mathbf{X} \rangle = \mathbf{c}^T \mathbf{x}$.

Following these results, the SDP problem (2-21) may be rewritten in the following standard form

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{x} & \max \quad & \mathbf{b}^T \mathbf{y} \\ \text{s.t.} \quad & \mathcal{A}\mathbf{x} = \mathbf{b} & \text{s.t.} \quad & \mathcal{A}^T \mathbf{y} + \mathbf{z} = \mathbf{c} \\ & \mathbf{x} \in \mathcal{C}^{PSD} & & \mathbf{z} \in \mathcal{C}^{PSD} \end{aligned}, \quad (2-24)$$

where \mathcal{C}^{PSD} is expressed in its vectorial version, i.e.,

$$\mathcal{C}^{PSD} = \left\{ \mathbf{x} \in \mathbb{R}^{n^2} \mid \text{Mat}_{n \times n}(\mathbf{x}) \succeq \mathbf{0} \right\}. \quad (2-25)$$

In the SDP case, the *complementary slackness condition* may be enlarged to an $N \times N^*$ † system of equations, as shown in the following Lemma 3.

Lemma 3 (SDP complementary slackness condition) *Let $\mathbf{X}, \mathbf{Z} \in \mathcal{C}^{PSD}$, then the slackness condition holds if and only if $\text{vec}(\mathbf{XZ}) = \mathbf{0}$. Equivalently,*

$$\mathbf{x}, \mathbf{z} \in \mathcal{C}^{PSD}, \quad \mathbf{x}^T \mathbf{z} = 0 \iff \mathbf{XZ} = \mathbf{0}. \quad (2-26)$$

Proof. See Appendix B. ■

Lemma 3 allows to express the optimality conditions for an SDP problem as

†* $N = 2n^2 + m$.

$$\begin{aligned}
\mathcal{A}\mathbf{x} &= \mathbf{b}, \quad \mathbf{x} \in \mathcal{C}^{PSD} \\
\mathcal{A}^T\mathbf{y} + \mathbf{z} &= \mathbf{c}, \quad \mathbf{z} \in \mathcal{C}^{PSD} \\
\text{vec}(\mathbf{XZ}) &= \mathbf{0}.
\end{aligned} \tag{2-27}$$

2.4

First-order parameter derivatives of conic programs

The sensitivity of optimal solutions $\mathbf{x}^*(\zeta)$ of a general nonlinear optimization problem, with respect to a given data perturbation ζ , is investigated in [45]. In such work Fiacco demonstrates that, under some constraints qualifications, differentiability of the objective function, and second-order sufficient conditions, the *implicit function theorem* can be applied in order to show that there exists a unique once continuously differentiable function $\mathbf{g}(\zeta) = [\mathbf{x}^*(\zeta), \boldsymbol{\lambda}^*(\zeta), \boldsymbol{\mu}^*(\zeta)]$ that yields the optimal point, as well as the corresponding Lagrange multipliers, in terms of a given perturbation ζ . Differentiation of such function, with respect to ζ , yields the sensitivity of the optimal solution. In other words, considering ζ as a parameter rather than a perturbation, it is possible to obtain a first-order parameter derivative of optimization problems. In fact, if the above conditions hold, the optimal solution derivative can be numerically evaluated as the solution of a system of linear equations stemming from the Jacobian of the KKT conditions with respect to the variables of the optimization problem [45].

Following the same idea, the sensitivity of SDP programs is derived in [46]. In this case, the conditions under which the *implicit function theorem* can be applied are given in terms of the nonsingularity of the Jacobian of optimality conditions (2-27) at the optimal solution, as well as differentiability of the problem data $(\mathcal{A}(\zeta), \mathbf{b}(\zeta), \mathbf{c}(\zeta))$ with respect to parameter ζ . Analogously, under nonsingularity of the Jacobian of optimality conditions (2-18), the same development is carried out in [42] to derive the sensitivity of SOCP optimal solutions. Likewise in the general nonlinear case, the derivative of the optimal solutions of SDP and SOCP programs can be numerically evaluated by solving the system of linear equations stemming from the differentiation of the corresponding optimality conditions. Moreover, since the Jacobian of KKT conditions at the optimal solution is assumed to be nonsingular, such system is guaranteed to have a unique solution.

In the following, the numerical evaluation of first-order parameters derivatives of SOCP and SDP problems is presented.

First-order parameter derivative for SOCP

The first-order parameter derivative of an SOCP optimal solution may be obtained via differentiation of the optimality conditions (2-18) with respect to a given parameter ζ , resulting in

$$\begin{bmatrix} \mathbf{A} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}^T & \mathbf{I}_d \\ \mathbf{Z} & \mathbf{0} & \mathbf{X} \end{bmatrix} \begin{bmatrix} \mathbf{x}' \\ \mathbf{y}' \\ \mathbf{z}' \end{bmatrix} = \begin{bmatrix} \mathbf{b}' \\ \mathbf{c}' \\ \mathbf{0} \end{bmatrix}, \quad (2-28)$$

where $(.)'$ represents the derivative of $(.)$ with respect to the parameter ζ and the left hand side matrix is the Jacobian of the KKT conditions.

The last row of the left-hand-side matrix in (2-28) may be directly obtained by simply noting that the product of matrices in (2-18) is commutative, i.e.,

$$\mathbf{XZ}\mathbf{e} = \mathbf{ZX}\mathbf{e} = \begin{bmatrix} x_1 z_1 + \mathbf{x}_{2:n}^T \mathbf{z}_{2:n} \\ z_1 \mathbf{x}_{2:n} + x_1 \mathbf{z}_{2:n} \end{bmatrix}, \quad (2-29)$$

and can also be expressed as

$$\mathbf{XZ}\mathbf{e} = \mathbf{Xz} = \mathbf{Zx} = \mathbf{ZX}\mathbf{e}. \quad (2-30)$$

Therefore, it is clear that

$$\frac{\partial(\mathbf{XZ}\mathbf{e})}{\partial \mathbf{x}} = \frac{\partial \mathbf{Zx}}{\partial \mathbf{x}} = \mathbf{Z}, \quad \frac{\partial(\mathbf{XZ}\mathbf{e})}{\partial \mathbf{z}} = \frac{\partial \mathbf{Xz}}{\partial \mathbf{z}} = \mathbf{X}. \quad (2-31)$$

In [42], it is shown that under primal and dual nondegeneracy and strict complementarity the Jacobian in (2-28) is guaranteed to be nonsingular. Therefore, if such conditions hold (and the problem data is differentiable with respect to ζ), then it is possible to show that the first-order parameter derivative is given as the unique solution of (2-28).

First-order parameter derivative for SDP

Analogously, the first-order parameter derivative of an SDP problem can be obtained by differentiating the optimality conditions of (2-27) with respect to a given parameter ζ . To make such equations amenable to direct differentiation, it is convenient first rewrite the last row in the following equivalent form:

$$\begin{aligned} \mathbf{XZ} = \mathbf{0} &\iff \text{vec}(\mathbf{XZ}) = (\mathbf{I}_d \otimes \mathbf{X}) \mathbf{z} = 0 \\ &= (\mathbf{Z} \otimes \mathbf{I}_d) \mathbf{x} = 0 \end{aligned} \quad (2-32)$$

where \otimes is the *Kronecker product* [47], $\mathbf{x} = \text{vec}(\mathbf{X})$, and $\mathbf{z} = \text{vec}(\mathbf{Z})$. Thus, it is straightforward to obtain the derivative of (2-27) with respect to ζ , i.e.,

$$\begin{bmatrix} \mathcal{A} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathcal{A}^T & \mathbf{I}_d \\ (\mathbf{Z} \otimes \mathbf{I}_d) & \mathbf{0} & (\mathbf{I}_d \otimes \mathbf{X}) \end{bmatrix} \begin{bmatrix} \mathbf{x}' \\ \mathbf{y}' \\ \mathbf{z}' \end{bmatrix} = \begin{bmatrix} \mathbf{b}' \\ \mathbf{c}' \\ \mathbf{0} \end{bmatrix}. \quad (2-33)$$

Similarly to the SOCP case, the matrix in (2-33) is guaranteed to be nonsingular if both primal-dual nondegeneracy and strict complementarity hold [46]. In this case, it is possible to show that the parameter derivative exists and it is given as the unique solution of system (2-33).

2.5

Computational aspects

The development of efficient codes for CP has its starting point in [48], in which Nesterov & Todd propose an efficient primal-dual interior-point algorithm for CP problems involving *self-scaled* cones. Both \mathcal{C}^{SOC} and \mathcal{C}^{PSD} are *self-scaled* cones, hence, not surprisingly, SOCP and SDP were the first optimization codes to emerge. A step-by-step numerical implementation of the primal-dual interior-point algorithm for conic quadratic optimization is introduced in [26], and the implementation and theoretical aspects of semidefinite optimization are reported in [42]. The efficiency of such primal-dual interior-point algorithm for CP has been evidenced in [49], in which, according to Lobo et al., “worst-case theoretical analysis shows that the number of iterations required to solve a SOCP problem grows at most as the square root of the number of design variables, while numerical experiments indicate that the typical number of iterations ranges between 5 and 50, almost independently of the problem size”. For a detailed comparison of the available software for the solution of conic programs the reader may refer to [50].

From a theoretical point of view, it is straightforward to observe that both the nonnegative orthant and the second-order cone are particular cases of the more general semidefinite cone [51]. On the other hand, from a computational perspective, it is clear that optimization algorithms for LP and SOCP provide a greater performance than the codes design to SDP. This indicates that the underlying structure of the constraints is crucial for efficiency in solving conic programs. On that account, most of the available codes for CP, e.g, MOSEK [52], SeDuMi [53], and SDPT3 [54], explicitly include multiple cones in the problem definition, hence providing the user with the flexibility to appropriately formulate the problem in order to achieve an optimal performance. In general, such codes define a mixed linear, conic quadratic, and semidefinite program as

$$\begin{aligned}
& \min \quad \mathbf{c}^l \mathbf{x}^l + \mathbf{c}^q \mathbf{x}^q + \mathbf{c}^s \mathbf{x}^s \\
& \text{s.t.} \quad \mathbf{A}^l \mathbf{x}^l + \mathbf{A}^q \mathbf{x}^q + \mathbf{A}^s \mathbf{x}^s = \mathbf{b} \quad , \\
& \quad \quad \mathbf{x}^l \in \mathbb{R}_+^n, \mathbf{x}^q \in \mathcal{C}^{SOC}, \mathbf{x}^s \in \mathcal{C}^{PSD}
\end{aligned} \tag{2-34}$$

with linear variables denoted as \mathbf{x}^l , conic quadratic variables \mathbf{x}^q and semidefinite variables \mathbf{x}^s . Since the CP formulation for elastoplastic state-update procedures leads to mixed conic quadratic and semidefinite programs, such flexibility is of great importance for the efficiency of the codes developed in this work.

3

The associative plasticity model

3.1

Material yield criteria

A yield criterion is a hypothesis defining a range, of stress states, in which a given material behaves elastically. If the load program causes stresses beyond such range, the material undergoes permanent deformations as yielding takes place. On that account, it is fair to say that yield criteria strongly dictates the mechanical behavior of elastoplastic materials and it is a fundamental component of the mathematical model of plasticity.

According to [55], a yield criterion may be mathematically stated as a local inclusion of a sublevel set of a given yield function Φ , i.e.,

$$\Sigma_p \in \mathcal{M} \quad , \quad \mathcal{M} = \{ \mathbf{X} \in \mathcal{S}^3 \mid \Phi(\mathbf{X}, \kappa_p) \leq 0 \} \quad , \quad (3-1)$$

where Σ_p is the Cauchy stress tensor at point p , \mathcal{M} is the set of admissible stress states, and \mathcal{S}^3 is the set of real 3×3 symmetric matrices. The function Φ represents the yield surface (for $\Phi(\mathbf{X}, \kappa_p) = 0$), and κ_p is the material yield limit at point p . It is worth noting that, according to [56], modern associative plasticity models usually require the yield criteria to be convex. This assumption ensures that fundamental thermodynamical principles are satisfied [7]. Therefore, only convex yield criteria are considered in this work.

It is also possible to define yield criteria based on multiple yield surfaces. Such multisurface criteria are given as the intersection of sublevel sets of two or more yield functions, i.e.,

$$\mathcal{M} = \bigcap_{i=1}^n \mathcal{M}_i = \{ \mathbf{X} \in \mathcal{S}^3 \mid \Phi_i(\mathbf{X}, \kappa_p^{(i)}) \leq 0 \} \quad . \quad (3-2)$$

Another property inherent to material yield criteria regards its differentiability. Although some classical plasticity models required the yield functions to be smooth, such restriction was latter relaxed by the introduction of the Koiter's rule [6] and the concept of *subdifferential sets* [57] which allowed the consideration of *singular* yield criteria into the plasticity model.

Conic representations of yield criteria has been extensively investigated in *limit* and *shakedown* analysis, a central theme in plasticity which aims

to estimate global collapse load factors of structures. Within the FEM, such analysis yield convex optimization problems with the following setting

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & \mathbf{A} \mathbf{x} = \mathbf{b} , \\ & \mathbf{x} \in \mathcal{M} \end{aligned} \tag{3-3}$$

where $\mathcal{M} = \{\mathbf{x} \in \mathbb{R}^n \mid g_i(\mathbf{x}) \leq 0\}$ are inequalities representing a given failure criteria. In the case that \mathcal{M} is a conic representable set, possibly with the introduction of some auxiliary variables, it is straightforward to cast problem (3-3) as a general CP problem (2-1). On that account, rather than solving a general convex programming problem, the analysis may be tackled by efficient algorithms, such as the ones described in Section 2.5. In light of this fact, many works on *limit* and *shakedown* analysis have extensively investigated how to cast practical yield criteria as conic constraints [55].

Although *limit* and *shakedown* analysis is beyond the scope of this text, the same conic representations of yield criteria are employed in the approach developed in this work. Therefore, for completeness, such casting of yield criteria as conic constraints is exposed in the following sections.

Positive semidefinite cone representable yield criteria*

Rankine criterion

Rankine criterion, which is also known as the *maximum tensile stress criterion* [58], is widely employed in the modeling of brittle materials. According to this criterion, yielding occurs when the maximum principal stress reaches a given tensile strength κ_R . The corresponding set of admissible stress states is

$$\mathcal{M}_R = \{\mathbf{X} \in \mathcal{S}^3 \mid \lambda_{max}(\mathbf{X}) \leq \kappa_R\} \tag{3-4}$$

where λ_{max} is the maximum eigenvalue of \mathbf{X} . The Rankine yield surface is depicted in Figure 3.1.

Tresca criterion

The Tresca criterion is considered as the first yield criterion for metals and suggests that yield occurs when the maximum shear stress reaches a critical

* This section was extracted from [44].

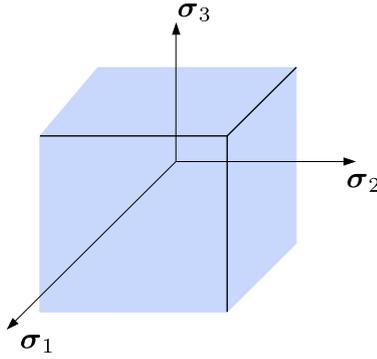


Figure 3.1: Rankine yield surface in the space of principal stresses.

value κ_T . In terms of the principal stresses, the set of admissible stress states may be given as

$$\mathcal{M}_T = \{\mathbf{X} \in \mathcal{S}^3 \mid 1/2(\sigma_1(\mathbf{X}) - \sigma_3(\mathbf{X})) \leq \kappa_T\}, \quad (3-5)$$

where $1/2(\sigma_1(\mathbf{X}) - \sigma_3(\mathbf{X})) = 1/2(\lambda_{max}(\mathbf{X}) - \lambda_{min}(\mathbf{X}))$ represents the absolute value of the maximum shear stress.

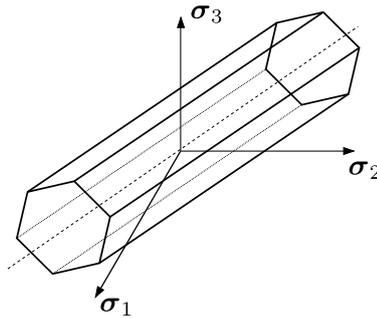


Figure 3.2: Tresca yield surface in the space of principal stresses.

Mohr-Coulomb criterion

The Tresca criterion may be considered as a particular case of the more general Mohr-Coulomb criterion. According to the Mohr-Coulomb theory, the shear strength τ in a plane is a function of the normal stress σ_n on the plane. In its simplest form, this function is given as

$$\tau = c - \sigma_n \tan \phi, \quad (3-6)$$

where c is the cohesion, and ϕ is the angle of internal friction.

In terms of the principal stresses, the elastic range of the Mohr-Coulomb criterion may be written as

$$\mathcal{M}_{MC} = \{\mathbf{X} \in \mathcal{S}^3 \mid \alpha \lambda_{max}(\mathbf{X}) - \beta \lambda_{min}(\mathbf{X}) \leq \kappa_{MC}\}, \quad (3-7)$$

where $\alpha = (1 + \sin \phi)$, $\beta = (1 - \sin \phi)$, and $\kappa_{MC} = 2c \cos \phi$.

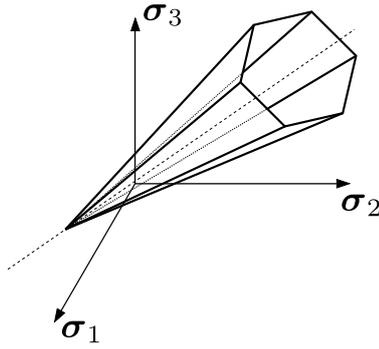


Figure 3.3: Mohr-Coulomb yield surface in the space of principal stresses.

Clearly, for $\phi = 0$ and $\kappa_{MC} = \kappa_T$, the Mohr-Coulomb criterion reduces to the Tresca criterion. Additionally, considering α and β as independent constants, one can retrieve the Rankine criterion by setting $\alpha = 1$, $\beta = 0$, and $\kappa_{MC} = \kappa_R$. Therefore, from a theoretical point of view, it is sufficient to cast the Mohr-Coulomb criterion as a semidefinite conic constraint.

Casting yield criteria as semidefinite constraints

The key point here is to observe that constraints regarding the maximum and minimum eigenvalues of a given symmetric matrix may be rewritten as semidefinite conic constraints. First, given $\mathbf{X} \in \mathbb{R}^{n \times n}$ and $t \in \mathbb{R}$, we have the following well-known identity:

$$\lambda_k(t\mathbf{I}_d - \mathbf{X}) = t - \lambda_{n+1-k}(\mathbf{X}) \quad (k = 1, \dots, n), \quad (3-8)$$

where $\lambda_k(\mathbf{X})$ is the k -th largest eigenvalue of $\mathbf{X} \in \mathbb{R}^{n \times n}$. It follows that

$$\lambda_{max}(t\mathbf{I}_d - \mathbf{X}) = t - \lambda_{min}(\mathbf{X}). \quad (3-9)$$

Therefore, it is clear that

$$\lambda_{max}(\mathbf{X}) \leq t \iff t\mathbf{I}_d - \mathbf{X} \succeq 0. \quad (3-10)$$

The case of the minimum eigenvalue constraints is directly derived from (3-10). We observe that

$$-\lambda_{min}(\mathbf{X}) = \lambda_{max}(-\mathbf{X}), \quad (3-11)$$

therefore,

$$-\lambda_{min}(\mathbf{X}) \leq t \iff t\mathbf{I}_d + \mathbf{X} \succeq 0. \quad (3-12)$$

Finally, using an auxiliary variable $y \in \mathbb{R}$, the inequality in (3-7) may be split into the following two inequalities [55]:

$$\begin{cases} \alpha\lambda_{max}(\mathbf{X}) \leq (\kappa_{MC} - y), \\ -\beta\lambda_{min}(\mathbf{X}) \leq y. \end{cases} \quad (3-13)$$

Therefore, the elastic range of the Mohr-Coulomb criterion may be written as an inclusion of the following semidefinite cone:

$$\mathcal{M}_{MC} = \left\{ \mathbf{W} \in \mathcal{S}^6 \mid \mathbf{W} = \begin{bmatrix} \mathbf{Y} & \mathbf{0} \\ \mathbf{0} & \mathbf{Z} \end{bmatrix} \succeq 0 \right\}. \quad (3-14)$$

where

$$\begin{aligned} \mathbf{Y} &= -\alpha\mathbf{X} + (\kappa_{MC} - y)\mathbf{I}_d, \\ \mathbf{Z} &= y\mathbf{I}_d + \beta\mathbf{X}. \end{aligned} \quad (3-15)$$

Second-order cone representable yield criteria*

von Mises criterion

The von Mises criterion is extensively employed in the modeling of metals and other ductile materials and suggests that yielding begins when the distortion strain energy reaches a critical value. In terms of the second deviatoric stress tensor invariant $J_2(\mathbf{X})$, the von Mises yield surface is given as

$$\sqrt{J_2(\mathbf{X})} = \kappa_{VM}, \quad (3-16)$$

where κ_{VM} is the yield stress in pure shear.

Therefore, the set of admissible stress states is expressed as

$$\mathcal{M}_{VM} = \{\mathbf{X} \in \mathcal{S}^3 \mid \sqrt{J_2(\mathbf{X})} \leq \kappa_{VM}\}. \quad (3-17)$$

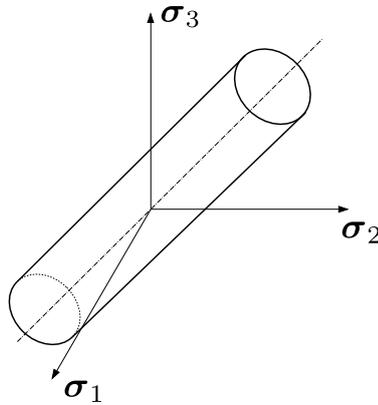


Figure 3.4: von Mises yield surface in the space of principal stresses.

* This section was extracted from [44].

Drucker-Prager criterion

The Drucker-Prager criterion is a generalization of the von Mises criterion in which a hydrostatic pressure sensitivity term is introduced. This sensitivity is enforced by adding a term in (3-16), which is related to the first stress tensor invariant $I_1(\mathbf{X})$, i.e.,

$$\frac{1}{3}\gamma I_1(\mathbf{X}) + \sqrt{J_2(\mathbf{X})} = \kappa_{DP}, \quad (3-18)$$

where γ and κ_{DP} are material parameters, that can be obtained from experimental data, and $I_1(\mathbf{X}) = \text{tr}(\mathbf{X})$ is related to the hydrostatic pressure $m(\mathbf{X})$ as $I_1(\mathbf{X}) = 3m(\mathbf{X})$. Clearly, for $\gamma = 0$, the von Mises criterion is retrieved.

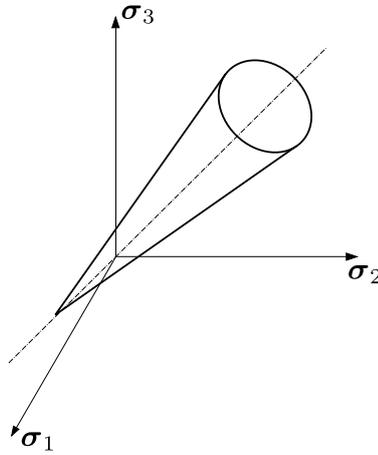


Figure 3.5: Drucker-Prager yield surface in the space of principal stresses.

Following the reasoning used in the previous section, it is sufficient to show how to cast the Drucker-Prager criterion into a second-order conic constraint. Although such criteria may also be represented by the more general case of semidefinite conic constraints, SOCP problems are easier to solve than SDP problems. Hence, second order conic representations should be considered whenever possible.

Casting yielding criteria as conic quadratic constraints

Collecting the six independent components of the stress tensor Σ into a vector σ yields

$$\sigma = [\Sigma_{11} \quad \Sigma_{22} \quad \Sigma_{33} \quad \Sigma_{12} \quad \Sigma_{13} \quad \Sigma_{23}]^T, \quad (3-19)$$

The second deviatoric stress tensor invariant may be written in the following matrix form:

$$J_2(\Sigma) = \sigma^T \mathbf{P} \sigma, \quad (3-20)$$

where

$$\mathbf{P} = \frac{1}{6} \begin{bmatrix} 2 & -1 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 & 0 \\ -1 & -1 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 6 & 0 & 0 \\ 0 & 0 & 0 & 0 & 6 & 0 \\ 0 & 0 & 0 & 0 & 0 & 6 \end{bmatrix}. \quad (3-21)$$

Given that \mathbf{P} is a symmetric positive semidefinite matrix it may be decomposed as

$$\mathbf{P} = \mathbf{V}^T \mathbf{D} \mathbf{V}, \quad (3-22)$$

where \mathbf{V} is a matrix that collects the eigenvectors of \mathbf{P} in a column-wise manner, and $\mathbf{D} = \text{diag}(\lambda_1(\mathbf{P}), \dots, \lambda_6(\mathbf{P}))$ with $\lambda_k(\mathbf{P}) \geq 0$ ($k = 1, \dots, 6$). Therefore, \mathbf{P} may be further decomposed as

$$\mathbf{L} = \mathbf{D}^{\frac{1}{2}} \mathbf{V} \implies \mathbf{P} = \mathbf{L}^T \mathbf{L}. \quad (3-23)$$

Such decomposition allows the square root of the second deviatoric stress tensor invariant to be expressed as the following quadratic norm:

$$\sqrt{J_2(\boldsymbol{\Sigma})} = \sqrt{\boldsymbol{\sigma}^T \mathbf{P} \boldsymbol{\sigma}} = \sqrt{\boldsymbol{\sigma}^T \mathbf{L}^T \mathbf{L} \boldsymbol{\sigma}} = \|\mathbf{L} \boldsymbol{\sigma}\|. \quad (3-24)$$

For the hydrostatic pressure term, the following vectorial form is sufficient:

$$m(\boldsymbol{\Sigma}) = \boldsymbol{\sigma}^T \mathbf{b}, \quad \mathbf{b} = [1/3 \quad 1/3 \quad 1/3 \quad 0 \quad 0 \quad 0]^T. \quad (3-25)$$

By collecting these results, the elastic range for the Drucker-Prager criterion may be rewritten as an inclusion of the following second-order cone:

$$\mathcal{M}_{DP} = \{(t, \mathbf{s}) \in \mathbb{R} \times \mathbb{R}^6 \mid t \geq \|\mathbf{s}\|, t \geq 0\} \quad (3-26)$$

where

$$\begin{aligned} t &= \kappa_{DP} - \gamma \boldsymbol{\sigma}^T \mathbf{b}, \\ \mathbf{s} &= \mathbf{L} \boldsymbol{\sigma}. \end{aligned} \quad (3-27)$$

3.2

Flow rules

Once the conditions upon which yielding may occur have been established, it still remains to define the mechanical behavior of the material when it undergoes permanent deformations. Specifically, a description of how plastic deformations evolve during the load program is required. Within plasticity, such evolution law is usually termed *plastic flow*, or *flow rule*, and it describes the rate of change of plastic strains in terms of the current stresses $\boldsymbol{\Sigma}$, i.e.,

$$\dot{\boldsymbol{\epsilon}}^p = \dot{\gamma} \mathbf{N}(\boldsymbol{\Sigma}) \quad (3-28)$$

where $\boldsymbol{\varepsilon}^p$ is the plastic strain tensor, $\dot{\gamma}$ is the so-called plastic multiplier, and $\mathbf{N}(\boldsymbol{\Sigma})$ is a function which defines the direction of evolution of plastic strains in terms of the stresses.

Another key component of a plastic model regards the changes to the yield surface during plastic deformation, phenomenon known as *hardening*. Such process is inherent to many materials but it was first observed in uniaxial tension tests of metal bars. In such experiment, it was detected that the tensile yield stress of the metal tended to grow as yielding took place. In order to describe the behavior of hardening, plastic models usually postulate an evolution law that gives the rate of change of *hardening variables* $\boldsymbol{\alpha}$ as

$$\dot{\boldsymbol{\alpha}} = -\dot{\gamma}\mathbf{H}(\boldsymbol{\Sigma}, \boldsymbol{\Theta}) \quad (3-29)$$

where $\boldsymbol{\alpha}$ are the *hardening variables*, $\dot{\gamma}$ is the same plastic multiplier as in (3-28), and $\mathbf{H}(\boldsymbol{\Sigma}, \boldsymbol{\Theta})$ is the function giving the direction of change of the *hardening variables* in terms of both the current stresses and *hardening forces* $\boldsymbol{\Theta}$. It is worth noting that in the presence of hardening, $\mathbf{N}(\boldsymbol{\Sigma}, \boldsymbol{\Theta})$ also becomes dependent of the *hardening forces*. The effects of *hardening* on the yield criterion consists in changes in size, shape, and orientation. Mathematically, such effects are described by incorporating the *hardening forces* into the yield criterion definition, i.e.,

$$\mathcal{M} = \{(\boldsymbol{\Sigma}, \boldsymbol{\Theta}) \in \mathcal{S}^3 \times \mathbb{R}^{n_h \times n_h} \mid \Phi(\boldsymbol{\Sigma}, \boldsymbol{\Theta}, \kappa_p) \leq 0\}, \quad (3-30)$$

where n_h denotes the number of hardening variables.

The model is called associative when the flow directions \mathbf{N} and \mathbf{H} are normal to the yield surface*, an assumption which is adopted in this work. In this case, an alternative and interesting manner to introduce such rate relations is by means of the PMPD. Instead of postulating the evolution laws, the formulation based on the PMPD follows from important results derived from the theory of MP. A rigorous mathematical analysis of the PMPD is carried out in [59]. By employing tools from convex analysis [60], Han & Reddy establish the elastoplastic equations for the general case of singular yield surfaces. Moreover, the PMPD allows to formulate multisurface plasticity in a straightforward manner. In fact, Koiter's rule [6] can be shown to be a direct consequence of the PMPD.

Based on the PMPD, the following section introduces the associative plasticity model for the general case of singular and multisurface criteria.

* w.r.t. the stresses and hardening forces.

3.3

Principle of maximum plastic dissipation

Let

$$\mathcal{A} = \{(\boldsymbol{\Sigma}, \boldsymbol{\Theta}) | f_i(\boldsymbol{\Sigma}, \boldsymbol{\Theta}) \leq 0, i = 1, \dots, n\} \quad (3-31)$$

be the set of all admissible pairs of stress $\boldsymbol{\Sigma}$ and hardening force $\boldsymbol{\Theta}$ for a given yield criterion defined by the intersection of sublevel sets of f_i . The PMPD states that, for a given plastic strain rate $\dot{\boldsymbol{\epsilon}}^p$ and internal hardening variable rate $\dot{\boldsymbol{\alpha}}$, the *actual state* $\{\boldsymbol{\Sigma}, \boldsymbol{\Theta}\} \in \mathcal{A}$ is, among all the admissible pairs, the one for which the plastic dissipation

$$\mathcal{D}^p(\boldsymbol{\Sigma}, \boldsymbol{\Theta}, \dot{\boldsymbol{\epsilon}}^p, \dot{\boldsymbol{\alpha}}) = \boldsymbol{\Sigma} : \dot{\boldsymbol{\epsilon}}^p - \boldsymbol{\Theta} : \dot{\boldsymbol{\alpha}} \quad (3-32)$$

attains its maximum value, i.e.,

$$\mathcal{D}^p(\boldsymbol{\Sigma}, \boldsymbol{\Theta}, \dot{\boldsymbol{\epsilon}}^p, \dot{\boldsymbol{\alpha}}) \geq \mathcal{D}^p(\boldsymbol{\Sigma}^*, \boldsymbol{\Theta}^*, \dot{\boldsymbol{\epsilon}}^p, \dot{\boldsymbol{\alpha}}) \quad \forall (\boldsymbol{\Sigma}^*, \boldsymbol{\Theta}^*) \in \mathcal{A}. \quad (3-33)$$

Accordingly, the PMPD postulates that the *actual state* $\{\boldsymbol{\Sigma}, \boldsymbol{\Theta}\} \in \mathcal{A}$ is the solution pair of the following optimization problem.

$$\begin{aligned} & \max_{\boldsymbol{\Sigma}, \boldsymbol{\Theta}} \mathcal{D}^p(\boldsymbol{\Sigma}, \boldsymbol{\Theta}, \dot{\boldsymbol{\epsilon}}^p, \dot{\boldsymbol{\alpha}}) \\ & \text{s.t.} \quad (\boldsymbol{\Sigma}, \boldsymbol{\Theta}) \in \mathcal{A} \end{aligned} \quad (3-34)$$

The Lagrangian associated with (3-34) is given by

$$\mathcal{L}(\boldsymbol{\Sigma}, \boldsymbol{\Theta}, \dot{\boldsymbol{\epsilon}}^p, \dot{\boldsymbol{\alpha}}, \dot{\boldsymbol{\gamma}}) = -\boldsymbol{\Sigma} : \dot{\boldsymbol{\epsilon}}^p + \boldsymbol{\Theta} : \dot{\boldsymbol{\alpha}} + \sum_{i=1}^n \dot{\gamma}_i f_i(\boldsymbol{\Sigma}, \boldsymbol{\Theta}), \quad (3-35)$$

where $\dot{\gamma}_i$ is a Lagrangian multiplier associated with constraint $f_i(\boldsymbol{\Sigma}, \boldsymbol{\Theta})$.

The corresponding KKT *optimality conditions* of such convex optimization problem [60] are precisely the associative plasticity constitutive equations, i.e.,

a. *Associative flow rule*

$$\dot{\boldsymbol{\epsilon}}^p = \sum_{i=1}^n \dot{\gamma}_i \mathbf{N}_i, \quad \mathbf{N}_i \in \partial_{\boldsymbol{\Sigma}} f_i \quad (3-36)$$

b. *Associative hardening law*

$$\dot{\boldsymbol{\alpha}} = \sum_{i=1}^n -\dot{\gamma}_i \mathbf{H}_i, \quad \mathbf{H}_i \in \partial_{\boldsymbol{\Theta}} f_i \quad (3-37)$$

c. *Loading/unloading complementary conditions*

$$\dot{\gamma}_i \geq 0, \quad f_i(\boldsymbol{\Sigma}, \boldsymbol{\Theta}) \leq 0, \quad \dot{\gamma}_i f_i(\boldsymbol{\Sigma}, \boldsymbol{\Theta}) = 0 \quad (3-38)$$

where $\partial_{\boldsymbol{\Sigma}} f_i$, $\partial_{\boldsymbol{\Theta}} f_i$ denote the *subdifferential* [60] of the potentially nonsmooth yield function f_i with respect to $\boldsymbol{\Sigma}$ and $\boldsymbol{\Theta}$, respectively. If the pair $(\boldsymbol{\Sigma}, \boldsymbol{\Theta})$ corresponds to a smooth portion of f_i , then \mathbf{N}_i and \mathbf{H}_i may be replaced by

$$\mathbf{N}_i = \frac{\partial f_i(\boldsymbol{\Sigma}, \boldsymbol{\Theta})}{\partial \boldsymbol{\Sigma}}, \quad \mathbf{H}_i = \frac{\partial f_i(\boldsymbol{\Sigma}, \boldsymbol{\Theta})}{\partial \boldsymbol{\Theta}}. \quad (3-39)$$

The constitutive model for associative plasticity considered in this work is completely defined by taking into account the following assumptions:

d. *Additive decomposition of total strains*

$$\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}^e + \boldsymbol{\varepsilon}^p, \quad (3-40)$$

where $\boldsymbol{\varepsilon}$ is the total strain tensor and $\boldsymbol{\varepsilon}^e$ is the elastic component,

e. *Isotropic and linear elastic behavior*

$$\boldsymbol{\Sigma} = \mathbf{C} : \boldsymbol{\varepsilon}^e, \quad (3-41)$$

where \mathbf{C} is the fourth-order *stiffness* tensor, and

f. *Linear hardening law*

$$\boldsymbol{\Theta} = \mathbf{D} : \boldsymbol{\alpha}. \quad (3-42)$$

where \mathbf{D} is the *generalized plastic modulus*.

Once the elastoplastic model is defined, the evolution of plastic strains and hardening variables is mathematically expressed by the following IVP.*

Problem 1 (Elastoplastic constitutive IVP) Consider p as a point of a body \mathcal{B} that obeys the constitutive model described in equations (3-36 – 3-42). The evolution of elastic strains $\boldsymbol{\varepsilon}^e(t)$, plastic strains $\boldsymbol{\varepsilon}^p(t)$ and hardening variables $\boldsymbol{\alpha}(t)$ at $p \in \mathcal{B}$, between an initial time t_0 and a subsequent instant T , are precisely the functions that solve the constrained system of differential-algebraic equations (DAE)

$$\left\{ \begin{array}{l} \dot{\boldsymbol{\varepsilon}}^e(t) = \dot{\boldsymbol{\varepsilon}}(t) - \dot{\gamma}(t)\mathbf{N}(t) \\ \dot{\boldsymbol{\alpha}}(t) = -\dot{\gamma}(t)\mathbf{H}(t) \\ \text{s.t. } \dot{\gamma}(t)f(\boldsymbol{\Sigma}(t), \boldsymbol{\Theta}(t)) = 0 \\ \quad \quad \quad f(\boldsymbol{\Sigma}(t), \boldsymbol{\Theta}(t)) \leq 0 \\ \quad \quad \quad \dot{\gamma}(t) \geq 0 \end{array} \right. \quad (3-43)$$

for a given strain history $\boldsymbol{\varepsilon}(t)$, $t \in [t_0, T]$, and initial conditions $\boldsymbol{\varepsilon}^e(t_0)$, $\boldsymbol{\varepsilon}^p(t_0)$, and $\boldsymbol{\alpha}(t_0)$.

3.4 Hardening models

The mathematical description of the hardening phenomenon relies on the selection of a suitable set of internal hardening variables and how those affect the shape of the yield surface. Such variables are usually chosen based on the

* From now on, for simplicity, the notation for the multisurface case is dropped. However, the next developments also holds for the multisurface case.

phenomenological aspects of the elastoplastic behavior of the material to be modeled. Most commonly, residual self-equilibrated stress and/or accumulated plastic strains are chosen as the hardening variables. In regard to the effect which these variables affect the shape of the yield surface, two distinct models are usually considered; the *isotropic hardening*, which defines the expansion/shrinkage of the yield surface uniformly in all directions, and the *kinematic hardening*, which denotes rigid body translations of the surface in space.

Isotropic hardening

In this work, the accumulated plastic strain

$$\bar{\varepsilon}^p = \int_0^t \dot{\gamma} dt \quad (3-44)$$

is chosen as the internal variable for modeling the *isotropic hardening* behavior of elastoplastic materials. Accordingly, a scalar *thermodynamical force* $\theta(\bar{\varepsilon}^p)$ is adopted and the hardening curve is postulated as

$$\kappa(\bar{\varepsilon}^p) = \kappa_0 + \theta(\bar{\varepsilon}^p). \quad (3-45)$$

Thus, the isotropic expansion/shrinkage is defined by changes in the yield stress limit in terms of the scalar hardening variable $\bar{\varepsilon}^p$. Moreover, it is assumed that the hardening law is linear, i.e., the relation between hardening forces $\theta(\bar{\varepsilon}^p)$ and accumulated plastic strains is given as

$$\theta(\bar{\varepsilon}^p) = h\bar{\varepsilon}^p \quad (3-46)$$

hence the linear hardening curve is expressed as

$$\kappa(\bar{\varepsilon}^p) = \kappa_0 + h\bar{\varepsilon}^p. \quad (3-47)$$

In terms of the model outlined in Section 3.3, the *isotropic hardening* is modeled as

$$\boldsymbol{\alpha} := \{\bar{\varepsilon}^p\}, \quad \boldsymbol{\Theta} := \{\theta\}, \quad \mathbf{D} := \{h\} \quad (3-48)$$

and a general yield criterion is restated as

$$\mathcal{M} = \{(\boldsymbol{\Sigma}, \theta) \in \mathcal{S}^3 \times \mathbb{R} \mid \Phi(\boldsymbol{\Sigma}, \kappa(\theta)) \leq 0\}. \quad (3-49)$$

In view of the associativity assumption adopted in this work, the rate equation for the internal variables is simply given as

$$\begin{aligned} \dot{\bar{\varepsilon}}^p &= -\dot{\gamma} \frac{\partial \Phi}{\partial \theta} \\ &= -\dot{\gamma} \frac{\partial \Phi}{\partial \kappa} \frac{\partial \kappa}{\partial \theta} \end{aligned} \quad (3-50)$$

which clearly leads to

$$\dot{\bar{\epsilon}}^P = \dot{\gamma}. \quad (3-51)$$

It is worth noting that such rate equation is in complete accordance with the definition of the accumulated plastic strain in Eq. (3-44). Also, the redefinition of the yield criterion due to isotropic hardening comprises the introduction of a linear equation in terms of the internal variable. Therefore, the conic representation described in Sections 3.1 and 3.1 can be easily extended to the isotropic hardening case by merely introducing a linear constraint with an additional variable.

Kinematic hardening

The so-called *kinematic hardening* is responsible for describing translations of the yield surface in stress space. This phenomenon, also known as the *Bauschinger effect*, is usually observed in uniaxial experiments which show that some materials, after being loaded beyond its elastic stress limit, tend to have its yielding limit decreased in compression whereas increased in tension. This process can be mathematically modeled by replacing the actual stresses Σ by the so-called *relative stress*

$$\boldsymbol{\eta} = \Sigma - \boldsymbol{\beta}, \quad (3-52)$$

where $\boldsymbol{\beta}$ is known as the *back-stress* tensor which denotes the *thermodynamical forces* of the *kinematic hardening* model. Accordingly, the tensor-valued *hardening variable* \mathbf{X} is related to self-equilibrated residual stresses that may affect the yielding resistance in certain directions.

As in the *isotropic hardening* case, a linear relation between the *internal variables* and *hardening forces* is considered, i.e.,

$$\boldsymbol{\beta} = \mathbf{K} : \mathbf{X}, \quad (3-53)$$

where \mathbf{K} is the fourth-order tensor known as *linear kinematic hardening modulus*.

According to the model described in Section 3, the *kinematic hardening* is thus modeled as

$$\boldsymbol{\alpha} := \{\mathbf{X}\}, \quad \boldsymbol{\Theta} := \{\boldsymbol{\beta}\}, \quad \mathbf{D} := \{\mathbf{K}\} \quad (3-54)$$

and the yield criterion is restated as

$$\mathcal{M} = \{(\Sigma, \boldsymbol{\beta}) \in \mathcal{S}^3 \times \mathbb{R}^{3 \times 3} \mid \Phi(\boldsymbol{\eta}, \kappa) \leq 0, \quad \boldsymbol{\eta} = \Sigma - \boldsymbol{\beta}\}. \quad (3-55)$$

It is thus clear that the description of yield criteria under *kinematic*

hardening accounts for a simple linear constraints which replaces the actual stresses with the relative measure $\boldsymbol{\eta}$. Therefore, likewise in the *isotropic hardening* case, it is straightforward to rewrite the conic representation of Sections 3.1 and 3.1 for the *kinematic hardening* case.

Under associativity, the evolution law for the *hardening variables* follows from

$$\begin{aligned}\dot{\mathbf{X}} &= -\dot{\gamma} \frac{\partial \Phi}{\partial \boldsymbol{\beta}} \\ &= -\dot{\gamma} \frac{\partial \Phi}{\partial \boldsymbol{\eta}} : \frac{\partial \boldsymbol{\eta}}{\partial \boldsymbol{\beta}}\end{aligned}\quad (3-56)$$

which is clearly equivalent to

$$\dot{\mathbf{X}} = \dot{\gamma} \mathbf{N} \quad (3-57)$$

with \mathbf{N} defined as in Section 3.2.

Mixed kinematic-isotropic hardening

It is also possible to consider the general case of *mixed kinematic-isotropic hardening*. In such case the yield surface is allowed both to scale and translate in space. Such hardening model may be directly modeled by simply stacking the isotropic and kinematic variables into enlarged diagonal matrices. Therefore, the variables of Section 3.3 specialize as

$$\boldsymbol{\alpha} := \begin{bmatrix} \bar{\epsilon}^p & 0 \\ 0 & \mathbf{X} \end{bmatrix}, \quad \boldsymbol{\Theta} := \begin{bmatrix} \theta & 0 \\ 0 & \boldsymbol{\beta} \end{bmatrix}, \quad \mathbf{D} := \begin{bmatrix} h & 0 \\ 0 & \mathbf{K} \end{bmatrix}. \quad (3-58)$$

3.5

Numerical integration of elastoplastic constitutive equations*

Exact solution to DAEs can only be obtained for extremely simple problems. In the case of elastoplastic constitutive IVPs, analytical solutions are only available for very simple strain histories involving elementary yield criteria. Thus, to solve complex models, numerical methods must be adopted. Although many integration schemes have been reported [61, 62], for simplicity, the fully implicit Euler scheme is adopted. Hence, considering a time interval $[t_n, t_{n+1}]$ with respect to a generic step of the Euler scheme, the discretized version of the elastoplastic constitutive IVP (3-43) is given as

* This section was extracted from [44].

$$\left\{ \begin{array}{l} \boldsymbol{\varepsilon}_{n+1}^e = \boldsymbol{\varepsilon}_n^e + \Delta\boldsymbol{\varepsilon} - \Delta\gamma\mathbf{N}_{n+1} \\ \boldsymbol{\alpha}_{n+1} = \boldsymbol{\alpha}_n - \Delta\gamma\mathbf{H}_{n+1} \\ \text{s.t. } \Delta\gamma f(\boldsymbol{\Sigma}_{n+1}, \boldsymbol{\Theta}_{n+1}) = 0 \\ \quad f(\boldsymbol{\Sigma}_{n+1}, \boldsymbol{\Theta}_{n+1}) \leq 0 \\ \quad \Delta\gamma \geq 0 \end{array} \right. , \quad (3-59)$$

where $\Delta(\cdot) = (\cdot)_{n+1} - (\cdot)_n$.

After the time-discretization step, a *predictor-corrector* scheme [7] is typically employed. In such framework, the solution for the constrained system of equations in (3-59) is solved via a two-step algorithm. First, in the *elastic trial step*, the variation of the total strains is assumed to be purely elastic. Accordingly, the potential (*trial*) solution is given as

$$\begin{aligned} \boldsymbol{\Sigma}_{n+1}^{\text{trial}} &= \mathbf{C} : (\boldsymbol{\varepsilon}_n^e + \Delta\boldsymbol{\varepsilon}) = \mathbf{C} : \boldsymbol{\varepsilon}_{n+1}^{e \text{ trial}}, \\ \boldsymbol{\Theta}_{n+1}^{\text{trial}} &= \boldsymbol{\Theta}_n, \\ \Delta\gamma &= 0. \end{aligned} \quad (3-60)$$

If this solution fulfills

$$f(\boldsymbol{\Sigma}_{n+1}^{\text{trial}}, \boldsymbol{\Theta}_{n+1}^{\text{trial}}) \leq 0, \quad (3-61)$$

the variables are simply updated as

$$(\cdot)_{n+1} = (\cdot)_{n+1}^{\text{trial}}. \quad (3-62)$$

Conversely, if condition (3-61) is not satisfied, the algorithm proceeds with the *plastic corrector step*, usually performed by means of RMAs. In this case, it is assumed that plastic deformation will occur; consequently, the following strictly positive condition must be satisfied:

$$\Delta\gamma > 0. \quad (3-63)$$

By virtue of this requirement, the problem in (3-59) yields the following system of nonlinear equations

$$\begin{aligned} \boldsymbol{\varepsilon}_{n+1}^e &= \boldsymbol{\varepsilon}_{n+1}^{\text{trial}} - \Delta\gamma\mathbf{N}_{n+1} \\ \boldsymbol{\alpha}_{n+1} &= \boldsymbol{\alpha}_{n+1}^{\text{trial}} - \Delta\gamma\mathbf{H}_{n+1}, \\ f(\boldsymbol{\Sigma}_{n+1}, \boldsymbol{\Theta}_{n+1}) &= 0 \end{aligned} \quad (3-64)$$

subjected to the strictly positive condition (3-63). In practice, the system of equations is first solved regardless of this constraint. Then, if the solution complies with condition (3-63), the result is accepted.

Alternatively, the solution to problem (3-59) may be sought as the optimum pair $(\boldsymbol{\Sigma}, \boldsymbol{\Theta})$ for the following convex optimization problem

$$\begin{aligned} \min_{\boldsymbol{\Sigma}, \boldsymbol{\Theta}} \quad & \mathcal{F}(\boldsymbol{\Sigma}, \boldsymbol{\Theta}) \\ \text{s.t.} \quad & f(\boldsymbol{\Sigma}, \boldsymbol{\Theta}) \leq 0 \end{aligned} \quad (3-65)$$

where $f(\boldsymbol{\Sigma}, \boldsymbol{\Theta})$ is a convex yield criterion function,

$$\begin{aligned} \mathcal{F}(\boldsymbol{\Sigma}, \boldsymbol{\Theta}) = & \frac{1}{2}(\boldsymbol{\Sigma}_{n+1}^{\text{trial}} - \boldsymbol{\Sigma}) : \mathbf{C}^{-1} : (\boldsymbol{\Sigma}_{n+1}^{\text{trial}} - \boldsymbol{\Sigma}) \\ & + \frac{1}{2}(\boldsymbol{\Theta}_{n+1}^{\text{trial}} - \boldsymbol{\Theta}) : \mathbf{D}^{-1} : (\boldsymbol{\Theta}_{n+1}^{\text{trial}} - \boldsymbol{\Theta}). \end{aligned} \quad (3-66)$$

The MP problem in (3-65) may also be interpreted as the discrete counterpart of the PMPD stated in (3-34). Additionally, as noted in [8], it allows for an intuitive geometric interpretation in the sense that the solution pair $(\boldsymbol{\Sigma}^*, \boldsymbol{\Theta}^*)$ is the *closest point projection* of the trial state $(\boldsymbol{\Sigma}_{n+1}^{\text{trial}}, \boldsymbol{\Theta}_{n+1}^{\text{trial}})$ onto the yield surface, considering the inner products induced by the inverse operators of both the *elastic constitutive tensor* \mathbf{C} and the *generalized hardening modulus* \mathbf{D} , i.e.

$$(\boldsymbol{\Sigma}^*, \boldsymbol{\Theta}^*) = \text{ARG} \left\{ \min_{(\boldsymbol{\Sigma}, \boldsymbol{\Theta}) \in \mathcal{A}} \frac{1}{2} \left\| \boldsymbol{\Sigma}_{n+1}^{\text{trial}} - \boldsymbol{\Sigma} \right\|_{\mathbf{C}^{-1}}^2 + \frac{1}{2} \left\| \boldsymbol{\Theta}_{n+1}^{\text{trial}} - \boldsymbol{\Theta} \right\|_{\mathbf{D}^{-1}}^2 \right\}. \quad (3-67)$$

The equivalence between problems (3-59) and (3-65) may be established in terms of the KKT conditions for problem (3-65). Considering the Lagrangian function of this problem

$$\mathcal{L}(\boldsymbol{\Sigma}, \boldsymbol{\Theta}, \Delta\gamma) = \mathcal{F}(\boldsymbol{\Sigma}, \boldsymbol{\Theta}) + \Delta\gamma f(\boldsymbol{\Sigma}, \boldsymbol{\Theta}), \quad (3-68)$$

it follows that the KKT conditions [60] may be expressed as

$$\begin{cases} \boldsymbol{\varepsilon}_{n+1}^e = \boldsymbol{\varepsilon}_n^e + \Delta\boldsymbol{\varepsilon} - \Delta\gamma \mathbf{N}_{n+1} \\ \boldsymbol{\alpha}_{n+1} = \boldsymbol{\alpha}_n - \Delta\gamma \mathbf{H}_{n+1} \\ \Delta\gamma f(\boldsymbol{\Sigma}_{n+1}, \boldsymbol{\Theta}_{n+1}) = 0 \\ f(\boldsymbol{\Sigma}_{n+1}, \boldsymbol{\Theta}_{n+1}) \leq 0 \\ \Delta\gamma \geq 0 \end{cases} . \quad (3-69)$$

Clearly, these are identical to the equations of problem (3-59).

4

The finite element method in nonlinear solid mechanics

4.1

The incremental initial value problem

Consider a domain $\Omega \subset \mathbb{R}^3$ with boundary $\partial\Omega$ occupied by a body \mathcal{B} subject to tractions forces \mathbf{t} at $\partial\Omega$ and body forces \mathbf{b} across Ω . Under infinitesimal deformations, the *principle of virtual work* yields the following weak statement of equilibrium at a given time t

$$\int_{\Omega} \rho \ddot{\mathbf{u}}(t) \cdot \boldsymbol{\eta} dv + \int_{\Omega} \boldsymbol{\sigma}(t) : \nabla \boldsymbol{\eta} dv - \int_{\Omega} \mathbf{b}(t) \cdot \boldsymbol{\eta} dv - \int_{\partial\Omega} \mathbf{t}(t) \cdot \boldsymbol{\eta} ds = 0 \quad \forall \boldsymbol{\eta} \in \mathcal{V}. \quad (4-1)$$

where ρ is the mass density distribution, \mathbf{u} is the displacement field, \mathcal{V} is the space of virtual displacements, i.e., the space of sufficiently regular functions, and $\boldsymbol{\eta}$ is an arbitrary virtual displacement field at time t .

Since this work is restricted to *quasi-static* problems, inertia effects are ignored and the first integral term in (4-1) is neglected. Therefore, the equilibrium is stated as

$$\int_{\Omega} \boldsymbol{\sigma}(t) : \nabla^s \boldsymbol{\eta} dv - \int_{\Omega} \mathbf{b}(t) \cdot \boldsymbol{\eta} dv - \int_{\partial\Omega} \mathbf{t}(t) \cdot \boldsymbol{\eta} ds = 0 \quad \forall \boldsymbol{\eta} \in \mathcal{V}. \quad (4-2)$$

where $\nabla^s(\cdot) = \frac{1}{2}(\nabla(\cdot) + \nabla^T(\cdot))$.

Considering an interval in time $t \in [t_0, T]$, the problem, as depicted in Figure 4.1, is completely defined by prescribing a history of body forces on Ω

$$\mathbf{b}(t) = \bar{\mathbf{b}}(t), \quad t \in [t_0, T], \quad (4-3)$$

natural boundary conditions on $\partial\Omega_t$

$$\mathbf{t}(t) = \bar{\mathbf{t}}(t), \quad t \in [t_0, T], \quad (4-4)$$

and essential boundary conditions on $\partial\Omega_u$

$$\mathbf{u}(t) = \bar{\mathbf{u}}(t), \quad t \in [t_0, T], \quad (4-5)$$

such that $\partial\Omega_t \cup \partial\Omega_u = \partial\Omega$.

In light of these definitions, the *infinitesimal quasi-static initial* BVP may be stated as [7]:

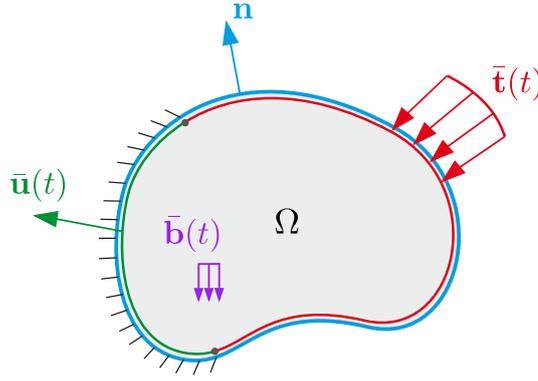


Figure 4.1: Schematic representation of a general initial BVP.

Problem 2 (The infinitesimal quasi-static initial BVP) Find a kinematically admissible displacement field $\mathbf{u} \in \mathcal{K}$ such that, for $t \in [t_0, T]$,

$$\int_{\Omega} \boldsymbol{\sigma}(t) : \nabla^s \boldsymbol{\eta} \, dv - \int_{\Omega} \mathbf{b}(t) \cdot \boldsymbol{\eta} \, dv - \int_{\partial\Omega_t} \mathbf{t}(t) \cdot \boldsymbol{\eta} \, ds = 0 \quad \forall \boldsymbol{\eta} \in \mathcal{V}. \quad (4-6)$$

with

$$\mathcal{V} = \{\boldsymbol{\eta} : \Omega \mapsto \mathcal{U} \mid \boldsymbol{\eta} = \mathbf{0} \text{ on } \partial\Omega_u\}. \quad (4-7)$$

and \mathcal{K} the space of all kinematically admissible displacements, i.e.,

$$\mathcal{K} = \{\mathbf{u} : \Omega \times \mathbb{R} \mapsto \mathcal{U} \mid \mathbf{u}(\mathbf{x}, t) = \bar{\mathbf{u}}(\mathbf{x}, t), \mathbf{x} \in \partial\Omega_u, t \in [t_0, T]\}. \quad (4-8)$$

Considering the model described in Chapter 3, it is clear that Eq. (4-6) becomes nonlinear in $\mathbf{u}(t)$, owing to the intrinsic non-proportional relation between stresses and strains. Moreover, due to the lack of analytical solutions for Problem 1, such relation is determined only in a time-discrete sense rather than in a continuum form. In such discrete sense, considering an arbitrary time increment $[t_n, t_{n+1}]$, the elastoplastic behavior is fully established by defining the following incremental functions:

$$\begin{aligned} \boldsymbol{\sigma}_{n+1} &= \hat{\boldsymbol{\sigma}}(\boldsymbol{\alpha}_n, \boldsymbol{\varepsilon}_{n+1}) = \hat{\boldsymbol{\sigma}}(\boldsymbol{\alpha}_{n+1}^{\text{trial}}, \boldsymbol{\varepsilon}_{n+1}^{\text{trial}}) \\ \boldsymbol{\alpha}_{n+1} &= \hat{\boldsymbol{\alpha}}(\boldsymbol{\alpha}_n, \boldsymbol{\varepsilon}_{n+1}) = \hat{\boldsymbol{\alpha}}(\boldsymbol{\alpha}_{n+1}^{\text{trial}}, \boldsymbol{\varepsilon}_{n+1}^{\text{trial}}) \end{aligned} \quad (4-9)$$

which gives the updated values of stresses $\boldsymbol{\sigma}_{n+1}$ and hardening variables $\boldsymbol{\alpha}_{n+1}$ in terms of the trial state $(\boldsymbol{\alpha}_{n+1}^{\text{trial}}, \boldsymbol{\varepsilon}_{n+1}^{\text{trial}})$. For instance, such incremental functions may be defined as in Eq. (3-67), i.e., as the optimal point of a convex optimization problem.

Owing to the time-discrete nature of the underlying constitutive model, it is useful to define the following incremental version of Problem 2 [7]:

Problem 3 (The incremental infinitesimal quasi-static initial BVP)

Given the internal variables $\boldsymbol{\alpha}_n$ at time t_n , find a displacement field $\mathbf{u}_{n+1} \in \mathcal{K}_{n+1}$ such that

$$\int_{\Omega} \hat{\boldsymbol{\sigma}}(\boldsymbol{\alpha}_n, \boldsymbol{\varepsilon}_{n+1}) : \nabla^s \boldsymbol{\eta} \, dv - \int_{\Omega} \mathbf{b}_{n+1} \cdot \boldsymbol{\eta} \, dv - \int_{\partial\Omega_t} \mathbf{t}_{n+1} \cdot \boldsymbol{\eta} \, ds = 0 \quad \forall \boldsymbol{\eta} \in \mathcal{V}. \quad (4-10)$$

where \mathbf{b}_{n+1} and \mathbf{t}_{n+1} are the prescribed body forces and surface tractions at time t_{n+1} , and the set of kinematically admissible displacements \mathcal{K}_{n+1} is rewritten as

$$\mathcal{K}_{n+1} = \{\mathbf{u}: \Omega \mapsto \mathcal{U} \mid \mathbf{u}(\mathbf{x}) = \bar{\mathbf{u}}_{n+1}(\mathbf{x}), \mathbf{x} \in \partial\Omega_u\}. \quad (4-11)$$

4.2 Finite element discretization

The starting point for the solution of BVPs via the FEM relies on an approximation statement of the principle of virtual work in which the domain of the body and related fields are discretized in a finite-dimensional space generated by the finite element interpolation functions. The introduction of such approximations reduces Problem 3 to a set of algebraic nonlinear equations which (approximately) express the equilibrium at time t_{n+1} .

Assume, without loss of generality, that both the body \mathcal{B} and the displacement field have dimension $n_{dim} = 3$. Let the domain $\Omega \subset \mathbb{R}^3$ be discretized with a mesh h consisting of n_{elem} elements, as depicted in Figure 4.2, such that the approximated domain is given as

$$\Omega^h = \bigcup_{e=1}^{n_{elem}} \Omega^{(e)} \quad (4-12)$$

where $\Omega^{(e)}$ is the domain of the e -th finite element.

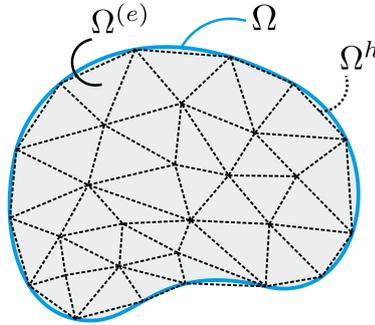


Figure 4.2: FEM discretization mesh of domain Ω .

Within each finite element e , the displacement field is approximated by interpolating the values at the element's nodes, i.e.,

$$\mathbf{u}(\mathbf{x})^h = \sum_{i=1}^{n_{nodes}} N_i^{(e)}(\mathbf{x}) \mathbf{u}^i \quad \text{on } \Omega^{(e)}, \quad (4-13)$$

where n_{nodes} is number of nodes of finite element e , $N_i^{(e)}$ is the local shape function associated with node i , and \mathbf{u}^i is displacement value at node i . An example of a linear shape function is depicted in Figure 4.3.

Based on such approximation, it is straightforward to rewrite such interpolation in a global sense, as depicted in Figure 4.4, i.e.,

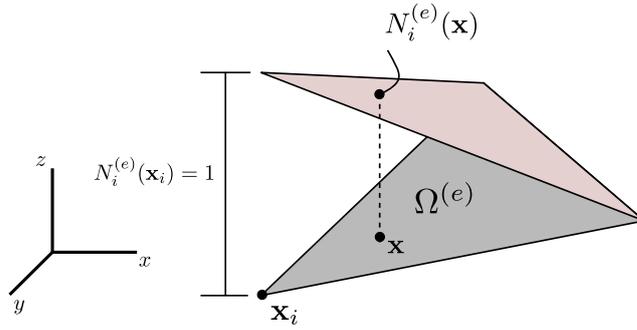


Figure 4.3: A linear shape function.

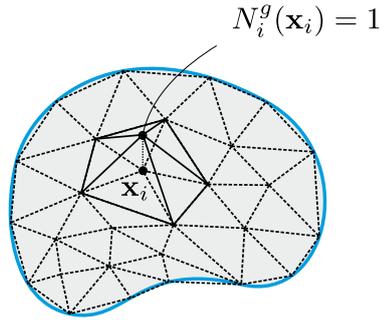


Figure 4.4: Global interpolation.

$$\mathbf{u}(\mathbf{x})^h = \sum_{i=1}^{n_{points}} N_i^g(\mathbf{x}) \mathbf{u}^i \quad \text{on } \Omega^h, \quad (4-14)$$

where n_{points} is the number of points of mesh h and N_i^g is the global shape function associated with point i .

The discretized version of the virtual work is accomplished by replacing the functional sets \mathcal{K} and \mathcal{V} by its discrete counterparts, i.e.,

$$\mathcal{K}_{n+1}^h = \left\{ \mathbf{u}^h(\mathbf{x}) = \sum_{i=1}^{n_{points}} N_i^g(\mathbf{x}) \mathbf{u}^i \mid \mathbf{u}^i = \bar{\mathbf{u}}_{n+1}(\mathbf{x}^i) \forall \mathbf{x}^i \in \partial\Omega_u \right\}, \quad (4-15)$$

$$\mathcal{V}^h = \left\{ \boldsymbol{\eta}^h = \sum_{i=1}^{n_{points}} N_i^g(\boldsymbol{\eta}) \boldsymbol{\eta}^i \mid \boldsymbol{\eta}^i = \mathbf{0} \forall \mathbf{x}^i \in \partial\Omega_u \right\}. \quad (4-16)$$

In order to rewrite the variational continuum statement in (4-10) as set of algebraic equations, it is convenient to introduce the following matrix notation.

The *global vector of nodal displacements* is written as

$$\mathbf{u}^T = [u_x^1, u_y^1, u_z^1, \dots, u_x^{n_{points}}, u_y^{n_{points}}, u_z^{n_{points}}]. \quad (4-17)$$

Accordingly, the *global vector of nodal virtual displacements* is expressed as

$$\boldsymbol{\eta}^T = [\eta_x^1, \eta_y^1, \eta_z^1, \dots, \eta_x^{n_{points}}, \eta_y^{n_{points}}, \eta_z^{n_{points}}]. \quad (4-18)$$

The global interpolation operator is defined as

$$\mathbf{N}^g(\mathbf{x}) = [\text{diag}[N_1^g(\mathbf{x})] \quad \text{diag}[N_2^g(\mathbf{x})] \quad \dots \quad \text{diag}[N_{n_{points}}^g(\mathbf{x})]], \quad (4-19)$$

where $\text{diag}[N_i^g(\mathbf{x})]$ corresponds to the $n_{dim} \times n_{dim}$ diagonal matrix with nonzero components equal to $N_i^g(\mathbf{x})$. In the three-dimensional case ($n_{dim} = 3$) it is explicitly given as

$$\begin{bmatrix} N_i^g(\mathbf{x}) & 0 & 0 \\ 0 & N_i^g(\mathbf{x}) & 0 \\ 0 & 0 & N_i^g(\mathbf{x}) \end{bmatrix}. \quad (4-20)$$

Consider also the following vectorial notation for the stresses and engineering strains

$$\boldsymbol{\sigma} = [\sigma_{11} \quad \sigma_{22} \quad \sigma_{33} \quad \sigma_{12} \quad \sigma_{23} \quad \sigma_{13}], \quad (4-21)$$

$$\boldsymbol{\varepsilon} = [\varepsilon_{11} \quad \varepsilon_{22} \quad \varepsilon_{33} \quad 2\varepsilon_{12} \quad 2\varepsilon_{23} \quad 2\varepsilon_{13}]. \quad (4-22)$$

Once such vectorial notation is imposed, it is useful to define the *global discrete symmetric gradient operator*, which, in the three-dimensional case, is expressed as

$$\mathbf{B}^g = \begin{bmatrix} N_{1,1}^g & 0 & 0 & N_{2,1}^g & 0 & 0 & \dots & N_{n_{points},1}^g & 0 & 0 \\ 0 & N_{1,2}^g & 0 & 0 & N_{2,2}^g & 0 & \dots & 0 & N_{n_{points},2}^g & 0 \\ 0 & 0 & N_{1,3}^g & 0 & 0 & N_{2,3}^g & \dots & 0 & 0 & N_{n_{points},3}^g \\ N_{1,2}^g & N_{1,1}^g & 0 & N_{2,2}^g & N_{2,1}^g & 0 & \dots & N_{n_{points},2}^g & N_{n_{points},1}^g & 0 \\ 0 & N_{1,3}^g & N_{1,2}^g & 0 & N_{2,3}^g & N_{2,2}^g & \dots & 0 & N_{n_{points},3}^g & N_{n_{points},2}^g \\ N_{1,3}^g & 0 & N_{1,1}^g & N_{2,3}^g & 0 & N_{2,1}^g & \dots & N_{n_{points},3}^g & 0 & N_{n_{points},1}^g \end{bmatrix}.$$

where $(\cdot)_{,i}$ denotes the derivative of (\cdot) with respect to the i -th coordinate.

Considering an arbitrary interpolated field $\mathbf{v}(\mathbf{x}) = \mathbf{N}^g(\mathbf{x})\mathbf{v}$, such operator defines the following linear transformation

$$\mathbf{B}^g(\mathbf{x})\mathbf{v} = \begin{bmatrix} \nabla^s(\mathbf{v}(\mathbf{x}))_{11} \\ \nabla^s(\mathbf{v}(\mathbf{x}))_{22} \\ \nabla^s(\mathbf{v}(\mathbf{x}))_{33} \\ 2\nabla^s(\mathbf{v}(\mathbf{x}))_{12} \\ 2\nabla^s(\mathbf{v}(\mathbf{x}))_{23} \\ 2\nabla^s(\mathbf{v}(\mathbf{x}))_{13} \end{bmatrix}. \quad (4-23)$$

and in the case of the displacement field, such operator defines the strain-displacement relation, i.e.,

$$\boldsymbol{\varepsilon} = \mathbf{B}^g\mathbf{u}, \quad (4-24)$$

with $\boldsymbol{\varepsilon}$ denoting the engineering strains according to Eq. (4-22).

In light of such matrix notation, the finite element approximation of the virtual work statement is expressed as

$$\int_{\Omega^h} \boldsymbol{\sigma} \cdot \mathbf{B}^g\boldsymbol{\eta} \, dv - \int_{\Omega^h} \mathbf{b} \cdot \mathbf{N}^g\boldsymbol{\eta} \, dv - \int_{\partial\Omega_t^h} \mathbf{t} \cdot \mathbf{N}^g\boldsymbol{\eta} \, ds = 0 \quad \forall \boldsymbol{\eta} \in \mathcal{V}^h. \quad (4-25)$$

which can be conveniently rewritten as

$$\left\{ \int_{\Omega^h} (\mathbf{B}^g)^T \boldsymbol{\sigma} dv - \int_{\Omega^h} (\mathbf{N}^g)^T \mathbf{b} dv - \int_{\partial\Omega_t^h} (\mathbf{N}^g)^T \mathbf{t} ds \right\} \boldsymbol{\eta} = 0 \quad \forall \boldsymbol{\eta} \in \mathcal{V}^h. \quad (4-26)$$

Since such equality must hold for any vector $\boldsymbol{\eta}$ it follows that the term in brackets must vanish, thus

$$\int_{\Omega^h} (\mathbf{B}^g)^T \boldsymbol{\sigma} dv - \int_{\Omega^h} (\mathbf{N}^g)^T \mathbf{b} dv - \int_{\partial\Omega_t^h} (\mathbf{N}^g)^T \mathbf{t} ds = 0. \quad (4-27)$$

If we denote the internal and external force vectors as

$$\mathbf{f}^{\text{ext}} = \int_{\Omega^h} (\mathbf{N}^g)^T \mathbf{b} dv + \int_{\partial\Omega_t^h} (\mathbf{N}^g)^T \mathbf{t} ds, \quad (4-28)$$

$$\mathbf{f}^{\text{int}} = \int_{\Omega^h} (\mathbf{B}^g)^T \boldsymbol{\sigma} dv, \quad (4-29)$$

then the system of algebraic equations defining the equilibrium is given as

$$\mathbf{r} = \mathbf{f}^{\text{int}} - \mathbf{f}^{\text{ext}} = \mathbf{0} \quad (4-30)$$

where \mathbf{r} denotes the residuum forces.

If we apply the same discretization to Problem 3 we arrive at the following nonlinear equilibrium equations

$$\mathbf{r}(\mathbf{u}_{n+1}) = \mathbf{f}^{\text{int}}(\mathbf{u}_{n+1}) - \mathbf{f}_{n+1}^{\text{ext}} = \mathbf{0} \quad (4-31)$$

with

$$\mathbf{f}^{\text{int}}(\mathbf{u}_{n+1}) = \int_{\Omega^h} (\mathbf{B}^g)^T \hat{\boldsymbol{\sigma}}(\boldsymbol{\alpha}_n, \boldsymbol{\varepsilon}_{n+1}) dv, \quad (4-32)$$

and

$$\mathbf{f}_{n+1}^{\text{ext}} = \int_{\Omega^h} (\mathbf{N}^g)^T \mathbf{b}_{n+1} dv + \int_{\partial\Omega_t^h} (\mathbf{N}^g)^T \mathbf{t}_{n+1} ds. \quad (4-33)$$

4.3

The nonlinear solution scheme

Although several numerical schemes are available in the literature of nonlinear FEM, for the sake of simplicity, this work adopts the well-known N-R scheme for the numerical solution of the nonlinear equilibrium equations. Owing to its asymptotic and quadratic rates of convergence, this numerical method is particularly suitable for the present problem.

The N-R method relies on taking steps towards the solution by solving the linearized versions of the corresponding nonlinear equations. The linearization of Eq. (4-31) may be derived either by discretizing a linearized version of the principle of virtual work [7] or by explicit differentiation of the discretized virtual work statement. The latter alternative yields

$$\begin{aligned}
\frac{\partial \mathbf{r}(\mathbf{u}_{n+1})}{\partial \mathbf{u}_{n+1}} &= \frac{\partial \mathbf{f}^{\text{int}}(\mathbf{u}_{n+1})}{\partial \mathbf{u}_{n+1}} = \int_{\Omega^h} (\mathbf{B}^g)^T \frac{\partial \hat{\boldsymbol{\sigma}}(\boldsymbol{\alpha}_n, \boldsymbol{\varepsilon}_{n+1})}{\partial \mathbf{u}_{n+1}} dv \\
&= \int_{\Omega^h} (\mathbf{B}^g)^T \frac{\partial \hat{\boldsymbol{\sigma}}(\boldsymbol{\alpha}_n, \boldsymbol{\varepsilon}_{n+1})}{\partial \boldsymbol{\varepsilon}_{n+1}} \frac{\partial \boldsymbol{\varepsilon}_{n+1}}{\partial \mathbf{u}_{n+1}} dv \\
&= \int_{\Omega^h} (\mathbf{B}^g)^T \mathbf{C}^{ep} \mathbf{B}^g dv
\end{aligned} \tag{4-34}$$

where

$$\mathbf{C}^{ep} = \frac{\partial \hat{\boldsymbol{\sigma}}(\boldsymbol{\alpha}_n, \boldsymbol{\varepsilon}_{n+1})}{\partial \boldsymbol{\varepsilon}_{n+1}} \tag{4-35}$$

is known as the *consistent elastoplastic tangent operator* (CETO) and

$$\mathbf{K}_T = \int_{\Omega^h} (\mathbf{B}^g)^T \mathbf{C}^{ep} \mathbf{B}^g dv \tag{4-36}$$

is called the *consistent elastoplastic tangent matrix*.

The first order approximation of Eq. (4-31), in a given point $\mathbf{u}_{n+1}^{(k)}$, is given as

$$\begin{aligned}
\mathbf{r}(\mathbf{u}_{n+1}) &\approx \mathbf{r}(\mathbf{u}_{n+1}^{(k)}) + \mathbf{K}_T|_{\mathbf{u}_{n+1}^{(k)}} (\mathbf{u}_{n+1} - \mathbf{u}_{n+1}^{(k)}) \\
&= \mathbf{r}(\mathbf{u}_{n+1}^{(k)}) + \mathbf{K}_T^{(k)} \delta \mathbf{u}^{(k+1)}
\end{aligned} \tag{4-37}$$

where

$$\mathbf{K}_T^{(k)} = \mathbf{K}_T|_{\mathbf{u}_{n+1}^{(k)}}, \quad \delta \mathbf{u}^{(k+1)} = (\mathbf{u}_{n+1} - \mathbf{u}_{n+1}^{(k)}). \tag{4-38}$$

Each iteration N-R consists in finding the root of such linearized equation and updating the current guess, i.e.,

$$\begin{aligned}
\delta \mathbf{u}^{(k+1)} &= -(\mathbf{K}_T^{(k)})^{-1} \mathbf{r}(\mathbf{u}_{n+1}^{(k)}), \\
\mathbf{u}_{n+1}^{(k+1)} &= \mathbf{u}_{n+1}^{(k)} + \delta \mathbf{u}^{(k+1)}.
\end{aligned} \tag{4-39}$$

until a given convergence criterion is satisfied. In particular, the convergence criterion adopted in this work is given as

$$\frac{\|\mathbf{r}_{n+1}^{(k)}\|_{\infty}}{\|\mathbf{f}_{n+1}^{ext}\|_{\infty}} \leq \epsilon_{\text{tol}}, \tag{4-40}$$

where $\mathbf{r}_{n+1}^{(k)}$ represents the residuum of the equilibrium balance at iteration k , \mathbf{f}_{n+1}^{ext} represents the external forces of at time t_{n+1} , and ϵ_{tol} is a given numerical tolerance.

4.4 Computational implementation

The FEM code for the solution of nonlinear solid mechanics BVPs developed in this work was implemented in MATLAB. The code is able to perform material nonlinear analyses of 2D/3D continuum problems under the hypothesis of small displacements and infinitesimal deformations. The computational implementation of a given material constitutive behavior is

carried out by providing the incremental constitutive functions of Eq. (4-9) as a MATLAB function handle. This allows the user to model a wide range of materials using any kind of integration procedure or any desired RMA strategy. An overview of the program structure and detailed instructions of how to use it are given in the following.

Program structure

The overall program is divided in three parts:

- Input: which comprises the finite element mesh, the materials' constitutive functions and the analysis parameters for the N-R scheme.
- Main procedure: which consists in the actual nonlinear FEM by means of the N-R scheme.
- Output: which account for the resulting data representing the solution of the problem, i.e., nodal displacements, elastic strains, plastic strains, reactions, etc.

and is comprised of the following MATLAB scripts:

```

/fem
├── nlfem_disp_ctrl.m
├── nlfem_load_ctrl.m
└── /elems
    ├── /Q4
    │   ├── Q4.m
    │   ├── Q4BMatrix.m
    │   ├── Q4ElementStiffness.m
    │   ├── Q4Stress.m
    │   ├── Q4BMatrixAxisymmetric.m
    │   ├── Q4ElementStiffnessAxisymmetric.m
    │   └── Q4StressAxisymmetric.m
    └── /Q8
        ├── Q8.m
        ├── Q8BMatrix.m
        ├── Q8ElementStiffness.m
        ├── Q8Stress.m
        ├── Q8BMatrixAxisymmetric.m
        ├── Q8ElementStiffnessAxisymmetric.m
        └── Q8StressAxisymmetric.m

```

All the scripts in the `/elems` folder are related to the computational implementation of specific finite elements. The `Q4.m` and `Q8.m` scripts consists of utility functions which emulates the construction of four-noded and eight-noded quadrilateral elements, respectively. The header of such functions are given as

```
function [Q8_class] = Q8(analysis_type)
function [Q4_class] = Q4(analysis_type)

% Analysis type
% 1 -> Plane Stress
% 2 -> Plane Strain
% 3 -> Axisymmetric
```

The output arguments `Q4_class` and `Q8_class` are MATLAB **structs** with the following fields:

```
      BMatrix: [function_handle]
    stiffnessMatrix: [function_handle]
linearStiffnessMatrix: [function_handle]
      stressFunction: [function_handle]
      elem_data: [ngptsx1 struct]
        nodes: [nx1 integer]
```

The `nodes` parameter is a $n \times 1$ array of integers which stores the identities of the element's nodes representing the connectivity of the mesh and `elem_data` is a $ngpts \times 1$ array of **structs** with fields:

```
sigma: [4x1 double]
   ep: [4x1 double]
alpha: [double]
```

where `sigma`, `ep`, `alpha` represents the stresses, plastic strains and hardening variable on a given integration point.

The most important aspect of the element implementation consists in the local stiffness matrix assembly function which is also responsible both for evaluating the internal forces due to the stresses and updating the state of the `elem_data` field, at each integration point. For instance, if one builds a `Q4` element subjected to plane stress, the `stiffnessMatrix` function handle field is set with the routine detailed in Listing 4.1.

The crucial part of this code consists in the evaluation of both the updated state (`elem_data`) and the CETO (`Cep`) at each integration point. This procedure takes place at the piece of code in red, which involves the use of the function handle stored in the `elem_prop.constitutive_model` field. Such function handle is provided as an input by the user and is responsible for

```

function [k,f,elem_data] = Q4ElementStiffness(x, y,...
                                           elem_data,...
                                           elem_prop,...
                                           du_elem)

t = elem_prop.t;

k=zeros(8,8);
f=zeros(8,1);

PG = [-1/sqrt(3) 1/sqrt(3)];

BMatrix = @(csi,eta) Q4BMatrix(csi,eta,x,y);

index = 0;

for i=1:2
    csi=PG(1,i);
    for j=1:2
        index = index + 1;
        eta=PG(1,j);
        [B,J] = BMatrix(csi,eta);
        dlt_strain = B*du_elem;
        [elem_data(index),Cep] = ...
            elem_prop.constitutive_model(dlt_strain,...
                                         elem_data(index),...
                                         elem_prop);

        k = k + B'*Cep*B*det(J)*t;
        f = f + B'*elem_data(index).sigma(1:3)*det(J)*t;
    end
end
return

```

Listing 4.1: Q4ElementStiffness function

modeling the elastoplastic behavior of the material.

An outline of each division of the program structure is briefly presented in the following.

Input

The starting point for using the code are the scripts `nl fem_disp_ctrl.m` and `nl fem_load_ctrl.m`, which employ displacement and load control for the nonlinear FEM incremental procedure, respectively. The header of these scripts are given in Listing 4.2.

```

function [output] = nlfem_disp_ctrl(coords    , ...
                                   connect   , ...
                                   dbc       , ...
                                   fbc      , ...
                                   elems_prop, ...
                                   properties, ...
                                   params)

function [output] = nlfem_load_ctrl(coords    , ...
                                    connect   , ...
                                    dbc       , ...
                                    fbc      , ...
                                    elems_prop, ...
                                    properties, ...
                                    params)

```

Listing 4.2: NLFEM headers

A detailed description of the input arguments are given in the following:

- **coords** - Mesh coordinates

```

% coords: [numNodes x 2 double] -> Array of 2D coordinates of
%                                     mesh nodes
%
%      : (x, y)
%      : x -> x coordinate value
%      : y -> y coordinate value

```

- **connect** - Mesh connectivity

```

% connect: [numElems x 1 struct] -> Array of finite element structs
%      : See Q4.m and Q8.m files for example of construction of
%      such structs

```

- **dbc** - Displacement boundary conditions

```

%      dbc: [numDBC x 5 double] -> Array of displacement boundary
%                                     conditions
%
%      : (node, ux, uy, is_prescribed_x, is_prescribed_y)
%      : node -> node id
%      : ux          -> prescribed x displacement value
%      : uy          -> prescribed y displacement value
%      : is_prescribed_x -> 1 if x displacement is prescribed,
%                                     0 otherwise
%      : is_prescribed_y -> 1 if y displacement is prescribed,
%                                     0 otherwise

```

- **fbc** - Force boundary conditions

```

%      fbc: [numDBC x 5 double] -> Array of force boundary
%                                     conditions
%
%      : (node, fx, fy)
%      : node -> node id

```

```
%      : fx  -> prescribed x nodal force value
%      : fy  -> prescribed y nodal force value
```

- **elems_prop** - Elements' properties

```
% elems_prop: [numElems x 1 integer] -> Array of element's property
%          identity
%          : Indexed according to the order of the properties
%          cells
%          : i.e. elems_prop(i) = k, then the i-th element has
%          : properties given by properties{k}.
```

- **properties** - Material properties

```
% properties: [numProps x 1 struct] -> Array of material properties
%          : (E, v, t, constitutive_model)
%          : E -> elastic modulus
%          : v -> Poisson coefficient
%          : t -> thickness
%          : constitutive_model -> function handle representing
%          the incremental constitutive
%          function
```

- **params** - Analysis parameters

```
% params      : [1x1 struct] Struct of analysis' parameters
%          : (max_it, steps, tol)
%          : max_it -> maximum number of iterations at each
%          increment
%          : step   -> array of increments from 0.0 to 1.0
%          : tol    -> numerical tolerance for the stopping
%          criterion
```

Main procedure

The main procedure is comprised of the actual FEM implementation for nonlinear quasi-static mechanics using the full N-R scheme. A schematic description of the algorithm, considering a load control scheme, is depicted in the flowchart of Figure 4.6. If the displacement control is selected the first box of the incremental loop is replaced by the box of Figure 4.5

update prescribed and free delta displacements
<pre>du(nDOF_Free+1:end,1) = u_enf(nDOF_Free+1:end,1)*params.step(inc) ... - u(nDOF_Free+1:end,1); du(1:nDOF_Free,1) = -K_init(1:nDOF_Free,1:nDOF_Free)\ ... (K_init(1:nDOF_Free,nDOF_Free+1:end)*du(nDOF_Free+1:end,1));</pre>

Figure 4.5: Displacement control update routine

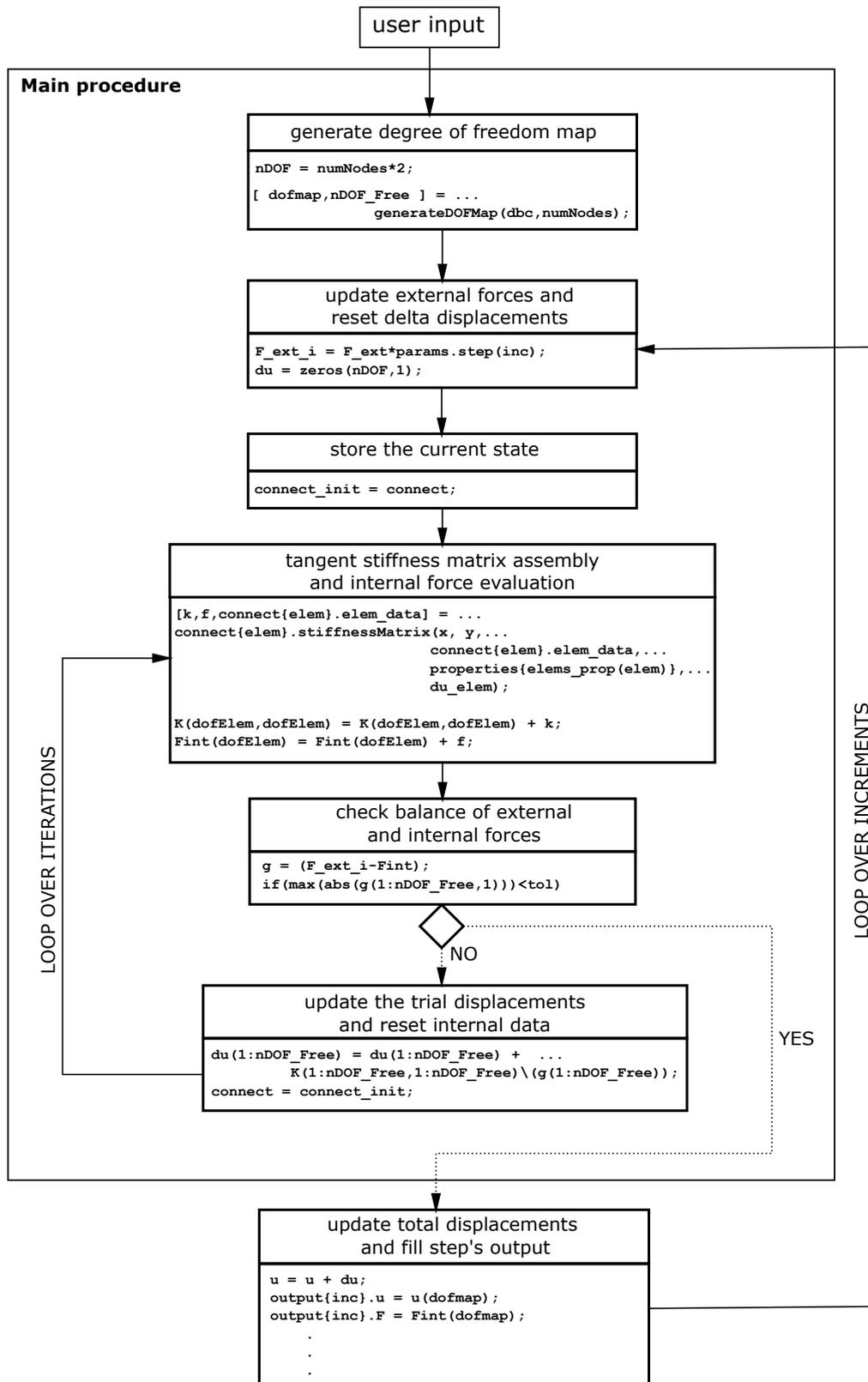


Figure 4.6: Main procedure flowchart

```

% Populate the output data for the current increment
u = u + du;
output{inc}.u = u(dofmap);
output{inc}.F = Fint(dofmap);
for elem = 1:numElem
    nGpts = length(elem_data{elem}(:));
    output{inc}.sxx{elem} = zeros(nGpts,1);
    output{inc}.syy{elem} = zeros(nGpts,1);
    output{inc}.sxy{elem} = zeros(nGpts,1);
    output{inc}.ep_xx{elem} = zeros(nGpts,1);
    output{inc}.ep_yy{elem} = zeros(nGpts,1);
    output{inc}.ep_xy{elem} = zeros(nGpts,1);
    output{inc}.alpha{elem} = zeros(nGpts,1);
    for i = 1:length(elem_data{elem}(:))
        output{inc}.sxx{elem}(i) = elem_data{elem}(i).sigma(1);
        output{inc}.syy{elem}(i) = elem_data{elem}(i).sigma(2);
        output{inc}.sxy{elem}(i) = elem_data{elem}(i).sigma(3);
        output{inc}.ep_xx{elem}(i) = elem_data{elem}(i).ep(1);
        output{inc}.ep_yy{elem}(i) = elem_data{elem}(i).ep(2);
        output{inc}.ep_xy{elem}(i) = elem_data{elem}(i).ep(3);
        output{inc}.alpha{elem}(i) = elem_data{elem}(i).alpha;
    end
end
end

```

Listing 4.3: Output data structure population routine

Output

The program's output is provided by routines `nlfe_m_disp_ctrl.m` and `nlfe_m_load_ctrl.m` as the output argument, as shown in headers of Listing 4.2. The complete routine for populating the output data structure is given in Listing 4.3. A detailed description of the output data structure is given in Listing 4.4.

```

% output: [numSteps x 1 struct] Array of output structs
% : (u, f, sxx, syy, sxy, szz, ep_xx, ep_yy, ep_xy, ep_zz, alp)
% : u [numNodes x 2 double] -> Array of nodal displacements
% : f [numNodes x 2 double] -> Array of nodal internal forces
% : sij [numElems x 1 cell] -> Cell of stress components in ij
% direction at each integration pt
% : ep_ij [numElems x 1 cell] -> Cell of plastic strains in ij
% direction at each integration pt
% : alp [numElems x 1 cell] -> Array of hardening variables at
% each integration pt
% * Each cell is a
% [ngpts x 1 double] array

```

Listing 4.4: Output data structure description

How to use

The coded is intended to allow the implementation of a material elastoplastic behavior in a straightforward manner. For this purpose, the user must provide a function handle with the following header:

```
function [ elem_data, Cep ] = <function_name>( dlt_strain, ...
                                             elem_data, ...
                                             elem_prop )

%Inputs:

% dlt_strain: [4 x 1 double] -> Array of total strains increment

% elem_data: [cell] -> Cell containing the current elastoplastic
%               variables of a given integration pt:
%               : sigma: [4x1 double] -> stresses
%               : ep: [4x1 double] -> plastic strains
%               : alpha: [double] -> hardening variables

% elem_prop: [cell] -> Cell containing the elastoplastic
%               material properties:
%               : E: [double] -> elastic modulus
%               : v: [double] -> Poisson coefficient
%               : t: [double] -> thickness
%               : constitutive_model [function handle]->
%               function representing the
%               incremental constitutive function

%Output:

% elem_data: [cell] -> Cell containing the updated elastoplastic
%               variables of a given integration pt:
%               : sigma: [4x1 double] -> stresses
%               : ep: [4x1 double] -> plastic strains
%               : alpha: [double] -> hardening variables

% Cep: [4 x 4 double] -> Consistent elastoplastic tangent matrix
%               associated with the implemented
%               state-update procedure
```

which must be capable of updating the elastoplastic variables in the `elem_data` field from a previous state plus the total strain increment `dlt_strain`. Additionally, the function must return the CETO consistent with the state-update procedure. For instance, a linear elastic material, under an axisymmetric state, may be simply implemented as:

```
function [ elem_data, C ] = linear_axisymmetric( strain, ...
                                                elem_data, ...
                                                elem_prop )

E = elem_prop.E;
v = elem_prop.v;

C = E/(1+v)/(1-2*v)*[1-v  v    0    v;
                    v 1-v    0    v;
                    0  0  (1-2*v)/2  0;
                    v  v    0    1-v];

elem_data.sigma = C*strain;

end
```

A minimal example of a elastoplastic analysis of a circular perforated plate under pressure using the aforementioned code is provided in Appendix C. Hopefully such example may guide the user on how to use the program and may serve as a template for other problems.

5

Return-mapping algorithms using conic programming*

In light of the model outlined in Chapter 3 and considering the vectorial notation of Eqs. (4-21) and (4-22), the optimization problem of (3-65) may be rewritten as

$$\begin{aligned} \min_{\boldsymbol{\sigma}, \boldsymbol{\alpha}} \quad & \mathcal{F}(\boldsymbol{\sigma}, \boldsymbol{\alpha}) \\ \text{s.t.} \quad & (\boldsymbol{\sigma}, \boldsymbol{\alpha}) \in \mathcal{M} \end{aligned} \quad (5-1)$$

with

$$\begin{aligned} \mathcal{F}(\boldsymbol{\sigma}, \boldsymbol{\alpha}) = \frac{1}{2}(\boldsymbol{\sigma}_{n+1}^{\text{trial}} - \boldsymbol{\sigma})^T \mathbf{S}(\boldsymbol{\sigma}_{n+1}^{\text{trial}} - \boldsymbol{\sigma}) \\ + \frac{1}{2}(\boldsymbol{\alpha}_{n+1}^{\text{trial}} - \boldsymbol{\alpha})^T \mathbf{D}(\boldsymbol{\alpha}_{n+1}^{\text{trial}} - \boldsymbol{\alpha}) \end{aligned}, \quad (5-2)$$

where $\mathbf{S} \in \mathcal{S}^{6 \times 6}$ is the symmetric compliance matrix associated with the fourth-order compliance tensor \mathbf{C}^{-1} .

Considering \mathcal{M} to be a conic representable set, the optimization problem in (5-1) consists of the convex quadratic objective function (5-2) and a general conic constraint. It is worth mentioning that such problems, in general, cannot be tackled using classical nonlinear gradient-based algorithms. This follows from the unsuitability of such algorithms to handle nonsmooth constraints, which would prevent singular yield criteria from being taken into account.

To express problem (5-2) in standard form, as in (2.3), the convex quadratic objective function is reduced to a second-order conic constraint. In the following paragraphs the main steps of this reduction are presented.

First, introducing the auxiliary variables $y_1, y_2 \in \mathbb{R}$, problem (5-1) is restated as

$$\begin{aligned} \min_{\boldsymbol{\sigma}, \boldsymbol{\alpha}, y_1, y_2} \quad & y_1 + y_2 \\ \text{s.t.} \quad & \frac{1}{2}(\boldsymbol{\sigma}_{n+1}^{\text{trial}} - \boldsymbol{\sigma})^T \mathbf{S}(\boldsymbol{\sigma}_{n+1}^{\text{trial}} - \boldsymbol{\sigma}) \leq y_1 \\ & \frac{1}{2}(\boldsymbol{\alpha}_{n+1}^{\text{trial}} - \boldsymbol{\alpha})^T \mathbf{D}(\boldsymbol{\alpha}_{n+1}^{\text{trial}} - \boldsymbol{\alpha}) \leq y_2 \\ & (\boldsymbol{\sigma}, \boldsymbol{\alpha}) \in \mathcal{M} \end{aligned}. \quad (5-3)$$

Next, owing to its positive definite property, the compliance matrix adheres to the following decomposition:

$$\mathbf{S} = \mathbf{M}^T \mathbf{M}, \quad (5-4)$$

thus

$$\frac{1}{2}(\boldsymbol{\sigma}_{n+1}^{\text{trial}} - \boldsymbol{\sigma})^T \mathbf{S}(\boldsymbol{\sigma}_{n+1}^{\text{trial}} - \boldsymbol{\sigma}) = \mathbf{w}^T \mathbf{w} \quad (5-5)$$

with

* This section was extracted from [44].

$$\mathbf{w} = \sqrt{\frac{1}{2}} \mathbf{M}(\boldsymbol{\sigma}_{n+1}^{\text{trial}} - \boldsymbol{\sigma}). \quad (5-6)$$

Similarly, assuming strict positiveness for the *hardening modulus* \mathbf{D} , it follows that $\mathbf{D} = \mathbf{N}^T \mathbf{N}$ thus

$$\frac{1}{2}(\boldsymbol{\alpha}_{n+1}^{\text{trial}} - \boldsymbol{\alpha})^T \mathbf{D}(\boldsymbol{\alpha}_{n+1}^{\text{trial}} - \boldsymbol{\alpha}) = \mathbf{z}^T \mathbf{z} \quad (5-7)$$

with

$$\mathbf{z} = \sqrt{\frac{1}{2}} \mathbf{N}(\boldsymbol{\alpha}_{n+1}^{\text{trial}} - \boldsymbol{\alpha}). \quad (5-8)$$

By taking the following identity into account

$$t = (t+1)^2/4 - (t-1)^2/4, \quad t \in \mathbb{R}, \quad (5-9)$$

both convex quadratic forms of (5-3) may expressed as the following second-order constraints:

$$\frac{y_1 + 1}{2} \geq \left\| \begin{bmatrix} \mathbf{w} \\ \frac{y_1 - 1}{2} \end{bmatrix} \right\|, \quad (5-10)$$

$$\frac{y_2 + 1}{2} \geq \left\| \begin{bmatrix} \mathbf{z} \\ \frac{y_2 - 1}{2} \end{bmatrix} \right\|. \quad (5-11)$$

Collecting these results and assuming \mathcal{M} to be a conic representable set, it is straightforward to express problem (5-1) as a general conic program, i.e.,

$$\begin{aligned} \min_{\mathbf{x}} \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & \mathbf{A} \mathbf{x} = \mathbf{b} \\ & \mathbf{x} \in \mathcal{K} = \mathcal{M} \times \mathcal{Q}_6 \times \mathcal{Q}_1 \end{aligned}, \quad (5-12)$$

where the design variables are collected in \mathbf{x} as

$$\mathbf{x} = [\boldsymbol{\sigma}, \boldsymbol{\alpha}, y_1, \mathbf{w}, y_2, \mathbf{z}]^T, \quad (5-13)$$

the vector \mathbf{c} is given as

$$\mathbf{c} = [0, 0, 1, 0, 1, 0]^T, \quad (5-14)$$

the linear constraints are expressed as

$$\mathbf{A} = \begin{bmatrix} \sqrt{\frac{1}{2}} \mathbf{M} & 0 & 0 & \mathbf{I}_d & 0 & 0 \\ 0 & \sqrt{\frac{1}{2}} \mathbf{N} & 0 & 0 & 0 & \mathbf{I}_d \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} \sqrt{\frac{1}{2}} \mathbf{M} \boldsymbol{\sigma}_{n+1}^{\text{trial}} \\ \sqrt{\frac{1}{2}} \mathbf{N} \boldsymbol{\alpha}_{n+1}^{\text{trial}} \end{bmatrix}, \quad (5-15)$$

whereas \mathcal{Q}_6 and \mathcal{Q}_1 are second order cones associated with inequalities (5-10) and (5-11), i.e.,

$$\mathcal{Q}_n = \left\{ (p, \mathbf{q}) \in \mathbb{R} \times \mathbb{R}^n \mid \frac{p+1}{2} \geq \left\| \begin{bmatrix} \mathbf{q} \\ \frac{p-1}{2} \end{bmatrix} \right\| \right\}. \quad (5-16)$$

Therefore, \mathcal{K} (in Eq. (5-12)) is a proper cone given as the Cartesian product of convex cones \mathcal{M} , \mathcal{Q}_6 , and \mathcal{Q}_1 [51].

Considering second-order cone representable yield criteria, such as those introduced in Section 3.1, the discrete elastic constitutive IVP (3-43) is reduced

to a second order cone program:

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & \mathbf{A} \mathbf{x} = \mathbf{b} \\ & \mathbf{x} \in \mathcal{C}^{\text{SOC}} = \mathcal{M}^{\text{CQR}} \times \mathcal{Q}_6 \times \mathcal{Q}_1 \end{aligned} \quad (5-17)$$

where \mathcal{M}^{CQR} is a conic quadratic representable set of admissible stresses and \mathcal{C}^{SOC} is a second order cone given as the Cartesian product of three second-order cones.

For semidefinite cone representable yield criteria, such as those presented in Section 3.1, the discrete elastoplastic constitutive IVP (3-43) is reduced to a mixed linear, conic quadratic and semidefinite program:

$$\begin{aligned} \min_{\mathbf{x}^l, \mathbf{x}^q, \mathbf{x}^s} \quad & \mathbf{c}^q T \mathbf{x}^q \\ \text{s.t.} \quad & \mathbf{A}^l \mathbf{x}^l + \mathbf{A}^q \mathbf{x}^q + \mathcal{A}^s \mathbf{x}^s = \hat{\mathbf{b}} \\ & \mathbf{x}^l \in \mathbb{R}^6, \mathbf{x}^q \in \mathcal{Q}_6 \times \mathcal{Q}_1, \mathbf{x}^s \in \mathcal{C}^{\text{PSD}} \end{aligned} \quad (5-18)$$

where \mathbf{c}^q is given as

$$\mathbf{c}^q = [1, 0, 1, 0]^T. \quad (5-19)$$

The design variables are collected as

$$\begin{aligned} \mathbf{x}^l &= [\boldsymbol{\sigma}, \boldsymbol{\alpha}]^T \\ \mathbf{x}^q &= [y_1, \mathbf{w}, y_2, \mathbf{z}]^T \end{aligned} \quad (5-20)$$

and the linear constraints are defined as

$$\mathbf{A}^l \mathbf{x}^l + \mathbf{A}^q \mathbf{x}^q + \mathcal{A}^s \mathbf{x}^s = \hat{\mathbf{b}} \equiv \begin{cases} \mathbf{w} = \sqrt{\frac{1}{2}} \mathbf{M} (\boldsymbol{\sigma}_{n+1}^{\text{trial}} - \boldsymbol{\sigma}) \\ \mathbf{z} = \sqrt{\frac{1}{2}} \mathbf{N} (\boldsymbol{\alpha}_{n+1}^{\text{trial}} - \boldsymbol{\alpha}) \\ \text{Mat}_{3 \times 3}(\mathbf{x}^s) = \begin{bmatrix} \sigma_1 & \sigma_4 & \sigma_5 \\ \sigma_4 & \sigma_2 & \sigma_6 \\ \sigma_5 & \sigma_6 & \sigma_3 \end{bmatrix} \end{cases}. \quad (5-21)$$

It is worth noting that CP problems (5-17) and (5-18) are guaranteed to be primal and dual feasible, i.e., both admit primal and dual optimal solutions. This result follows from the fact that, given an unstressed state, i.e., $(\mathbf{0}, \boldsymbol{\alpha}) \in \text{int } \mathcal{M}$, we can arbitrarily choose the free variables y_1, y_2 such that $(y_1, \mathbf{w}) \in \text{int } \mathcal{Q}_6$ and $(y_2, \mathbf{z}) \in \text{int } \mathcal{Q}_1$. Therefore, clearly, $\exists(\mathbf{x}, \mathbf{z}) \in \text{int } \mathcal{K} \times \text{int } \mathcal{K}^*$.

5.1

Consistent elastoplastic tangent operator numerical evaluation scheme

Linearization of the *virtual work principle*, introduced in Section 4.3, requires the evaluation of the CETO, which is given as

$$\mathbf{C}^{\text{ep}} = \frac{\partial \hat{\boldsymbol{\sigma}}(\boldsymbol{\varepsilon}_{n+1}, \boldsymbol{\alpha}_n)}{\partial \boldsymbol{\varepsilon}_{n+1}} = \frac{\partial \hat{\boldsymbol{\sigma}}(\boldsymbol{\varepsilon}_{n+1}^{\text{trial}}, \boldsymbol{\alpha}_{n+1}^{\text{trial}})}{\partial \boldsymbol{\varepsilon}_{n+1}^{\text{trial}}} \quad (5-22)$$

where the *incremental* constitutive function $\hat{\sigma}$ implicitly defines the stress at a pseudo-time t_{n+1} in terms of *trial* strains and *hardening variables*.

According to the MP approach employed herein, the implicit constitutive function $\hat{\sigma}$ may be defined as the argument of the optimization problem in (5-1), i.e.,

$$\hat{\sigma}(\boldsymbol{\varepsilon}_{n+1}^{e \text{ trial}}, \boldsymbol{\alpha}_{n+1}^{\text{trial}}) = \text{ARG} \left[\min_{(\boldsymbol{\sigma}, \boldsymbol{\alpha}) \in \mathcal{M}} \mathcal{F}(\boldsymbol{\sigma}, \boldsymbol{\alpha}) \right]. \quad (5-23)$$

In this setting, the corresponding CETO may be obtained as the first-order parameter derivative of the corresponding optimal solution.

For instance, consider a second-order cone representable yield criterion for which the discrete elastoplastic constitutive IVP (3-43) is solved using the second-order cone program (5-17). Differentiating the optimality conditions of such problem, with respect to $\boldsymbol{\varepsilon}_{n+1}^{e \text{ trial}}$, yields a system of linear equations as in (2-28), with the right-hand side given as

$$\mathbf{b}' = \begin{bmatrix} \sqrt{\frac{1}{2}} \mathbf{M} \mathbf{C}^e \\ \mathbf{0} \end{bmatrix}, \quad (5-24)$$

where $\mathbf{C}^e \in \mathcal{S}^{6 \times 6}$ is the symmetric elasticity matrix corresponding to the array representation of the elasticity tensor \mathbf{C} , obtained from

$$\mathbf{C}^e = \frac{\partial \boldsymbol{\sigma}_{n+1}^{\text{trial}}}{\partial \boldsymbol{\varepsilon}_{n+1}^{\text{trial}}}, \quad (5-25)$$

and

$$\mathbf{c}' = \frac{\partial [0, 0, 1, 0, 1, 0]^T}{\partial \boldsymbol{\varepsilon}_{n+1}^{\text{trial}}} = \mathbf{0}. \quad (5-26)$$

Such expressions evidence the differentiability of the problem data with respect to $\boldsymbol{\varepsilon}^{e \text{ trial}}$. Therefore, under strict complementarity and primal-dual nondegeneracy, the solution of system (2-28) leads to the derivative of the optimal solutions of problem (5-17) with respect to the elastic trial strain $\boldsymbol{\varepsilon}_{n+1}^{e \text{ trial}}$, i.e.,

$$\mathbf{x}' = \left[\frac{\partial \hat{\sigma}(\boldsymbol{\varepsilon}_{n+1}^{\text{trial}}, \boldsymbol{\alpha}_{n+1}^{\text{trial}})}{\partial \boldsymbol{\varepsilon}_{n+1}^{\text{trial}}}, \frac{\partial \hat{\boldsymbol{\alpha}}(\boldsymbol{\varepsilon}_{n+1}^{\text{trial}}, \boldsymbol{\alpha}_{n+1}^{\text{trial}})}{\partial \boldsymbol{\varepsilon}_{n+1}^{\text{trial}}}, \dots \right]^T. \quad (5-27)$$

Therefore, the CETO is readily recovered by collecting the relevant elements of \mathbf{x}' , i.e.,

$$\mathbf{C}^{ep} = \frac{\partial \hat{\sigma}(\boldsymbol{\varepsilon}_{n+1}^{\text{trial}}, \boldsymbol{\alpha}_{n+1}^{\text{trial}})}{\partial \boldsymbol{\varepsilon}_{n+1}^{\text{trial}}}. \quad (5-28)$$

An identical scheme may be employed in the evaluation of the CETO in the case of semidefinite cone representable yield criteria for which the discrete elastic constitutive IVP (3-43) is solved using the semidefinite cone program (5-18).

KKT Jacobian Nonsingularity Condition

Owing to the projective feature of the *closest point projection* (3-65) and the assumed convexity of the yield criteria, MP problem (5-1) is guaranteed to have an unique solution [40]. Therefore, it is possible to cast (5-1) as an equivalent CP problem which will also have primal-dual unique solutions. In [46] and [42], it is shown that under uniqueness of optimal solutions, primal-dual nondegeneracy follows from strict complementarity. Hence, in order to show that the CETO may be recovered as a first-order parameter derivative, it suffices to demonstrate that the optimal solutions of problems (5-17) and (5-18) are strictly complementary. Although this property must be analyzed for each case, it is possible to prove that strict complementarity holds for all yield criteria considered in this text. The proof of strict complementarity for CP problems involving the Drucker-Prager and von Mises criteria is detailed in the following.

Strict Complementarity solutions of discrete IVPs via CP

Definition 5.1 Let \mathcal{C}^{SOC} be a second-order cone. Its interior and boundary are defined as

$$\begin{aligned} \text{int } \mathcal{C}^{SOC} &= \{\mathbf{x} \in \mathbb{R}^n \mid x_1 > \|x_{2:n}\|, x_1 > 0\}, \\ \text{bd } \mathcal{C}^{SOC} &= \{\mathbf{x} \in \mathbb{R}^n \mid x_1 = \|x_{2:n}\|, x_1 \geq 0\}. \end{aligned} \quad (5-29)$$

Definition 5.2 Let \mathbf{x} (\mathbf{y}, \mathbf{z}) be the optimal primal and solutions for problem (2-15). Then $(\mathbf{x}, \mathbf{y}, \mathbf{z})$ satisfies strict complementarity if and only if $\mathbf{x} + \mathbf{z} \in \text{int } \mathcal{C}^{SOC}$.

Every second-order cone representable yield criterion can be written, in vectorial form, as

$$\mathcal{M}_{\text{SOCR}} = \{(\boldsymbol{\sigma}, \alpha) \in \mathbb{R}^6 \times \mathbb{R} \mid \|\mathbf{L}\boldsymbol{\sigma}\| \leq (\kappa_0 + \bar{h}\alpha) - \mathbf{b}^T \boldsymbol{\sigma}\} \quad (5-30)$$

with arbitrary constants $\mathbf{L} \in \mathbb{R}^{6 \times m}$, $\mathbf{b} \in \mathbb{R}^6$, $\kappa_0 \in \mathbb{R}$, and $\bar{h} \in \mathbb{R}$.

Formulating (5-17) as a dual conic program leads to

$$\begin{aligned} \max_{\mathbf{y}, \mathbf{z}} \quad & \mathbf{b}^T \mathbf{y} \\ \text{s.t.} \quad & \mathbf{A}_i^T \mathbf{y} + \mathbf{z}_i = \mathbf{c}_i \\ & \mathbf{z}_i \in \mathcal{Q}_{\text{SOC}} \end{aligned} \quad (5-31)$$

with dual variables

$$\begin{aligned}
\mathbf{z}_1 &= (\kappa, \mathbf{s}) & \mathbf{y}_1 &= (\boldsymbol{\sigma}, \alpha) \\
\mathbf{z}_2 &= (t_{11}, t_{12}, \mathbf{a}_1) & \mathbf{y}_2 &= t_1 \\
\mathbf{z}_3 &= (t_{21}, t_{22}, a_2) & \mathbf{y}_3 &= t_2
\end{aligned} \tag{5-32}$$

where the objective function is given as

$$\mathbf{b}^T \mathbf{y} = -\frac{1}{2}t_1 - \frac{1}{2}t_2 \tag{5-33}$$

and the linear constraints are given as

$$\begin{aligned}
\mathbf{A}_1^T \mathbf{y} + \mathbf{z}_1 = \mathbf{c}_1 &\equiv \begin{cases} \kappa = (\kappa_0 + \bar{h}\alpha) - \mathbf{p}^T \boldsymbol{\sigma} \\ \mathbf{s} = \mathbf{L}\boldsymbol{\sigma} \end{cases} \\
\mathbf{A}_2^T \mathbf{y} + \mathbf{z}_2 = \mathbf{c}_1 &\equiv \begin{cases} t_{11} = \frac{t_1+1}{2} \\ t_{12} = \frac{t_1-1}{2} \\ \mathbf{a}_1 = \mathbf{S}^{\frac{1}{2}} (\boldsymbol{\sigma}^{\text{trial}} - \boldsymbol{\sigma}) \end{cases} \\
\mathbf{A}_3^T \mathbf{y} + \mathbf{z}_3 = \mathbf{c}_3 &\equiv \begin{cases} t_{21} = \frac{t_2+1}{2} \\ t_{22} = \frac{t_2-1}{2} \\ a_2 = h^{\frac{1}{2}} (\alpha^{\text{trial}} - \alpha) \end{cases}
\end{aligned} \tag{5-34}$$

The primal problem, dual to (5-31), is thus given as

$$\begin{aligned}
\min_{\mathbf{x}} \quad & \mathbf{c}^T \mathbf{x} \\
\text{s.t.} \quad & \sum_i \mathbf{A}_i \mathbf{x}_i = \mathbf{b} \\
& x_i \in \mathcal{Q}_{\text{SOC}}
\end{aligned} \tag{5-35}$$

with primal variables

$$\begin{aligned}
\mathbf{x}_1 &= (\bar{\kappa}, \bar{\mathbf{s}}) \\
\mathbf{x}_2 &= (\bar{t}_{11}, \bar{t}_{12}, \bar{\mathbf{a}}_1) \\
\mathbf{x}_3 &= (\bar{t}_{21}, \bar{t}_{22}, \bar{a}_2)
\end{aligned} \tag{5-36}$$

and the linear constraints are given as

$$\sum_i \mathbf{A}_i \mathbf{x}_i = \mathbf{b} \equiv \begin{cases} \mathbf{p}\bar{\kappa} - L^T \bar{\mathbf{s}} + \mathbf{S}^{\frac{1}{2}} \bar{\mathbf{a}}_1 = \mathbf{0} \\ -\bar{h}\bar{\kappa} = -h^{\frac{1}{2}} \bar{a}_2 \\ -\frac{1}{2}\bar{t}_{11} - \frac{1}{2}\bar{t}_{12} = -\frac{1}{2} \\ -\frac{1}{2}\bar{t}_{21} - \frac{1}{2}\bar{t}_{22} = -\frac{1}{2} \end{cases} \tag{5-37}$$

In order to show that the optimal solutions for (5-17) satisfy strict complementarity, it is left to demonstrate that

$$\mathbf{x}_i + \mathbf{z}_i \in \mathcal{Q}_{\text{SOC}}^{(i)}, \quad i = 1, 2, 3. \tag{5-38}$$

Equivalently, as noted in [42], we can show that none of the following

states are possible:

$$\begin{aligned}
 (1) \quad & \mathbf{x}_i \in \text{bd } \mathcal{Q}_{\text{SOC}}^{(i)}, \quad \mathbf{z}_i = \mathbf{0} \\
 (2) \quad & \mathbf{x}_i = \mathbf{0}, \quad \mathbf{z}_i \in \text{bd } \mathcal{Q}_{\text{SOC}}^{(i)} . \\
 (3) \quad & \mathbf{x}_i = \mathbf{0}, \quad \mathbf{z}_i = \mathbf{0}
 \end{aligned} \tag{5-39}$$

Using the third and fourth rows of Eq. (5-37), and taking Lemma 3 into account, we note that

$$\begin{aligned}
 \frac{t_{11} - t_{12}}{\beta_1} = 1, \quad \beta_1 &= \frac{t_{11}}{\bar{t}_{11}}, \\
 \frac{t_{21} - t_{22}}{\beta_2} = 1, \quad \beta_2 &= \frac{t_{21}}{\bar{t}_{21}}.
 \end{aligned} \tag{5-40}$$

Hence, according to rows (3,4,5,6) of Eq. (5-34), we have

$$\begin{aligned}
 t_{11} - t_{12} = 1 \implies \beta_1 = 1 \implies t_{11} = \bar{t}_{11}, \quad (t_{12}, \mathbf{a}_1) &= -(\bar{t}_{12}, \bar{\mathbf{a}}_1), \\
 t_{21} - t_{22} = 1 \implies \beta_1 = 1 \implies t_{21} = \bar{t}_{21}, \quad (t_{22}, \mathbf{a}_2) &= -(\bar{t}_{22}, \bar{\mathbf{a}}_2).
 \end{aligned} \tag{5-41}$$

Moreover, according to the second row of Eq. (5-37)

$$-\bar{h}\bar{\kappa} = h^{\frac{1}{2}}\bar{\mathbf{a}}_2 = -h^{\frac{1}{2}}\mathbf{a}_2 = h(\alpha^{\text{trial}} - \alpha) \implies \bar{\kappa} = \frac{h}{\bar{h}}\Delta\alpha > 0 \tag{5-42}$$

Therefore, given that the trial state will always lie outside the yield criterion and that t_1, t_2 are strict positive, it is clear that

$$\mathbf{x}_i \neq \mathbf{0}, \quad i = 1, 2, 3. \tag{5-43}$$

Thus states (2) and (3) of Eq. (5-39) are not possible. To show that state (1) is also not possible, a particular yield criterion must be analyzed. For instance, if we select the von Mises yield criterion it is clear that state (1) is unreachable. In this case the strict positiveness of $\kappa = (\kappa_0 + \bar{h}\alpha) > 0$ implies that $\mathbf{z}_1 \neq \mathbf{0}$. Considering the Drucker-Prager criterion, condition $\mathbf{z}_1 \neq \mathbf{0}$ only takes place when the updated state belongs to the apex of the yield surface. In this case, as noted in [7], the trial state must belong to the complementary cone. Moreover, if we solve the first row of Eq. (5-37), considering a hydrostatic stress state, we can identify the norm of $\bar{\mathbf{s}}$ as

$$\|\bar{\mathbf{s}}\| = \frac{J_2(\Sigma_{n+1}^{\text{trial}})}{G} \tag{5-44}$$

whereas the dual variable $\bar{\kappa}$ is equivalent to the plastic multiplier $\Delta\gamma$, i.e.,

$$\bar{\kappa} = \Delta\gamma. \tag{5-45}$$

Therefore, according to [7] (Section 8.3.1), whenever the updated stress lies at the apex, the following must hold

$$\frac{J_2(\Sigma_{n+1}^{\text{trial}})}{G} - \Delta\gamma < 0. \quad (5-46)$$

This implies that $(\bar{\kappa}, \bar{\mathbf{s}}) \in \text{int } \mathcal{Q}_{\text{SOC}}^{(1)}$. Hence, even when the updated stress lies at the apex, the optimal solutions for the CP problem involving the Drucker-Prager yield criterion satisfy strict complementarity.

With similar considerations, it is also possible to prove that the positive semidefinite representable yield criteria considered in this work satisfy the Jacobian nonsingularity condition. In particular, considering the Mohr-Coulomb criterion in the framework of CP problem (5-18), it is possible to prove that the Jacobian of KKT conditions will be nonsingular even when the updated stress lies on the edges or the apex of the Mohr-Coulomb surface.

6

Numerical examples*

6.1

Finite step accuracy and iso-error maps

A key aspect in the numerical analysis of elastoplastic problems is the finite step accuracy of the integration algorithms employed in the solution of the elastoplastic IVPs. In theory, it is possible to demonstrate accuracy order of an integration algorithm in terms of sufficiently small strain increments [63]. However, for practical problems, it is crucial to investigate the accuracy of such algorithms for increasing increments. The finite step accuracy of the proposed integration algorithms is assessed using *iso-error maps* [64]. This is a systematic method for testing the accuracy of state-update procedures and has been employed in many studies [62], [65], [66], [67]. Such maps are generated by prescribing a sequence of strain increments to an initial stress state corresponding to a point on the yield surface. The approximate numerical solution is obtained by applying the integration algorithm in a single step. The error is then evaluated as the difference between the numerical σ^{num} and exact solutions σ^{exact} . In this work the following error definition is adopted:

$$\varepsilon = \frac{\sqrt{(\sigma^{\text{exact}} - \sigma^{\text{num}}) \cdot (\sigma^{\text{exact}} - \sigma^{\text{num}})}}{\sqrt{\sigma^{\text{exact}} \cdot \sigma^{\text{exact}}}} \times 100. \quad (6-1)$$

Owing to the lack of analytical solutions for practical problems, the “exact” solution is obtained by dividing the strain increment into a sufficiently large number of subincrements. This procedure is valid whenever the integration algorithm is, at least, first-order accurate, which is true for the *fully implicit* scheme employed in this study. In the following sections, the *iso-error maps* of the proposed integration algorithm are presented. For comparison with other traditional RMA, the same *iso-error maps* are generated using the state-update procedures of the HYPLAS library [7]. In accordance with [7] and [62], the “exact” solutions are obtained by dividing each strain increment into 1000 sub-steps.

* This chapter was extracted from [44].

Drucker-Prager iso-error map

To assess the accuracy of the integration algorithm based on the second-order conic program (5-17), consider a perfectly plastic Drucker-Prager model with the Young's modulus $E = 20\text{MPa}$, Poisson's ratio $\nu = 0.4$, and remaining parameters chosen to match the outer edges of the Mohr-Coulomb surface with friction angle of $\phi = 36^\circ$ and cohesion of $c = 0.2\text{MPa}$, i.e.,

$$\gamma = \frac{6\sin(\phi)}{\sqrt{3}(3 - \sin(\phi))}, \quad \kappa_{DP} = \frac{6\cos(\phi)c}{\sqrt{3}(3 - \sin(\phi))}. \quad (6-2)$$

According to [68], these material parameters are within a realistic range of values for a typical saturated soil present in coastal ranges of southeastern Brazil.

Considering a point $\boldsymbol{\sigma}^i$ on the yield surface, the unit stress increment directions \mathbf{n} and \mathbf{t} are chosen as the normal and tangential directions at $\boldsymbol{\sigma}^i$, respectively, as depicted in Figure 6.1.

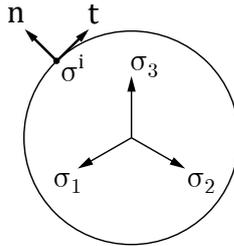


Figure 6.1: Stress increment directions for the Drucker-Prager model.

The *iso-error map* shown in Figure 6.2 is generated by evaluating the error between the “exact” and numerical solutions of the following trial stress states:

$$\boldsymbol{\sigma}^{\text{trial}} = \boldsymbol{\sigma}^i + \Delta\sigma_T \mathbf{t} + \Delta\sigma_N \mathbf{n}, \quad (6-3)$$

where $\Delta\sigma_T$ and $\Delta\sigma_N$ are scalar parameters that control the magnitude of the stress increment in the tangent and normal directions, respectively. The initial point $\boldsymbol{\sigma}^i$ employed in this example corresponds to a pure shear stress state in which the nonzero components are $\sigma_{xy} = \sigma_{yx} = \kappa_{DP}$.

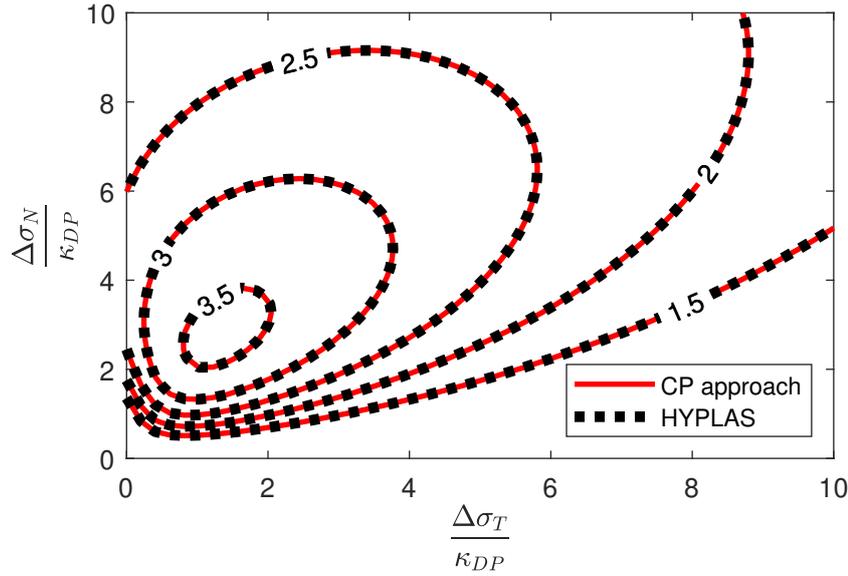


Figure 6.2: *Iso-error map* for the Ducker-Prager model.

As expected, both integration algorithms give virtually the same results. This is because both algorithms are based on the *fully implicit* scheme, which is the main factor for the finite step accuracy of the method. Nevertheless, the agreement between the results shows that the proposed approach is as accurate as traditional RMAs.

Mohr-Coulomb iso-error maps

The accuracy of the integration algorithm based on the positive semidefinite conic program (5-18) is investigated by generating *iso-error maps* for a perfectly plastic Mohr-Coulomb model. The same material parameters employed in [69] for the modeling of a sandy shale layer present in a cross section of a deep coal mine in China, i.e., Young's modulus $E = 6200\text{MPa}$, Poisson's ratio $\nu = 0.23$, friction angle of $\phi = 28^\circ$ and cohesion of $c = 1\text{MPa}$ are adopted. For the construction of the *iso-error maps*, three different initial stress states on the Mohr-Coulomb yield surface are considered: point σ^A at the middle of an edge; points σ^B and σ^C at the intersections of the adjacent edges. The directions \mathbf{t}^A , \mathbf{t}^B , and \mathbf{t}^C are chosen as the tangential directions to the edges belonging to the deviatoric plane, as shown in Figure 6.3. The direction \mathbf{n}^A is the normal direction to the yield surface at point σ^A , whereas \mathbf{n}^B and \mathbf{n}^C are given by the arithmetic mean of the normals of the adjacent edges at σ^B and σ^C , respectively. The nonzero components of the specific initial stress states employed in this example are:

$$\begin{aligned}
 \sigma_{xx}^A &= \frac{c \cos(\phi)}{2(\sin(\phi)+3)} - \frac{c \cos(\phi)}{\sin(\phi)-3} & \sigma_{xx}^B &= -\frac{2c \cos(\phi)}{\sin(\phi)-3} & \sigma_{xx}^C &= \frac{c \cos(\phi)}{\sin(\phi)+3} \\
 \sigma_{yy}^A &= \frac{c \cos(\phi)}{2(\sin(\phi)+3)} - \frac{c \cos(\phi)}{\sin(\phi)-3} & \sigma_{yy}^B &= -\frac{2c \cos(\phi)}{\sin(\phi)-3} & \sigma_{yy}^C &= \frac{c \cos(\phi)}{\sin(\phi)+3} \\
 \sigma_{zz}^A &= \frac{2c \cos(\phi)}{\sin(\phi)-3} - \frac{c \cos(\phi)}{\sin(\phi)+3} & \sigma_{zz}^B &= \frac{4c \cos(\phi)}{\sin(\phi)-3} & \sigma_{zz}^C &= -\frac{2c \cos(\phi)}{\sin(\phi)+3} \\
 \sigma_{xy}^A &= \frac{3c \cos(\phi)}{2(\sin(\phi)+3)} & & & \sigma_{xy}^C &= \frac{3c \cos(\phi)}{\sin(\phi)+3}
 \end{aligned}
 \tag{6-4}$$

The *iso-error maps* for each initial state are presented in Figures 6.4, 6.5, and 6.6.

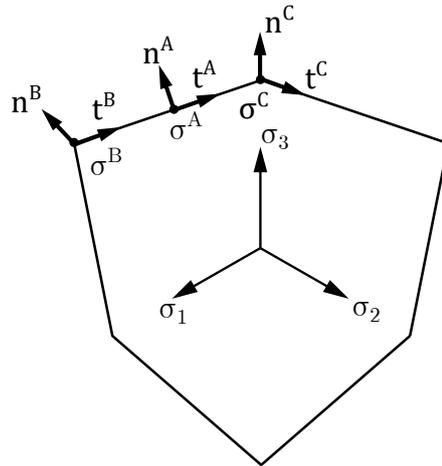


Figure 6.3: Stress increment directions for the Mohr-Coulomb model.

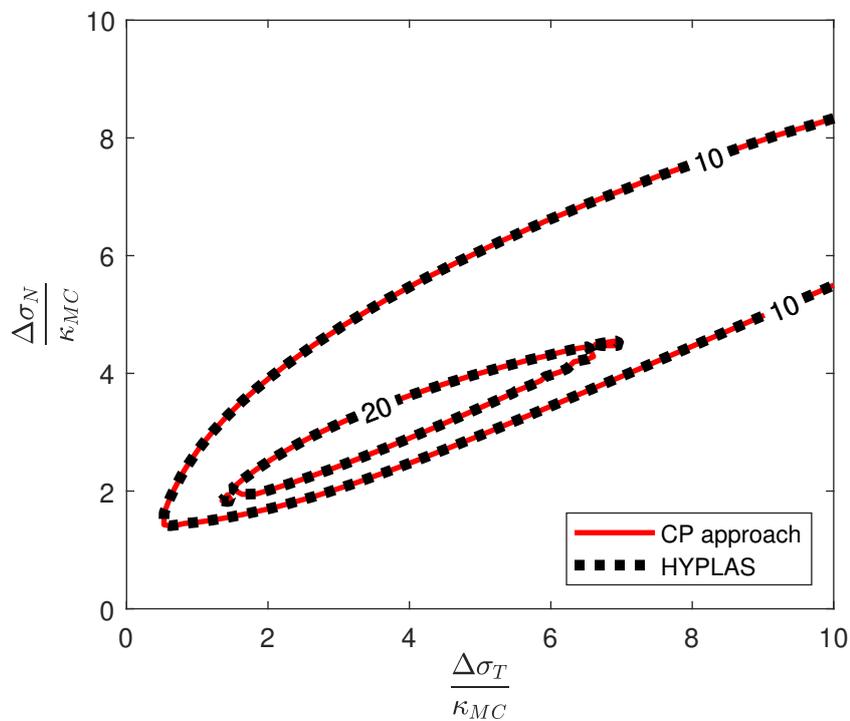


Figure 6.4: *Iso-error map* for the Mohr-Coulomb model departing from point A.

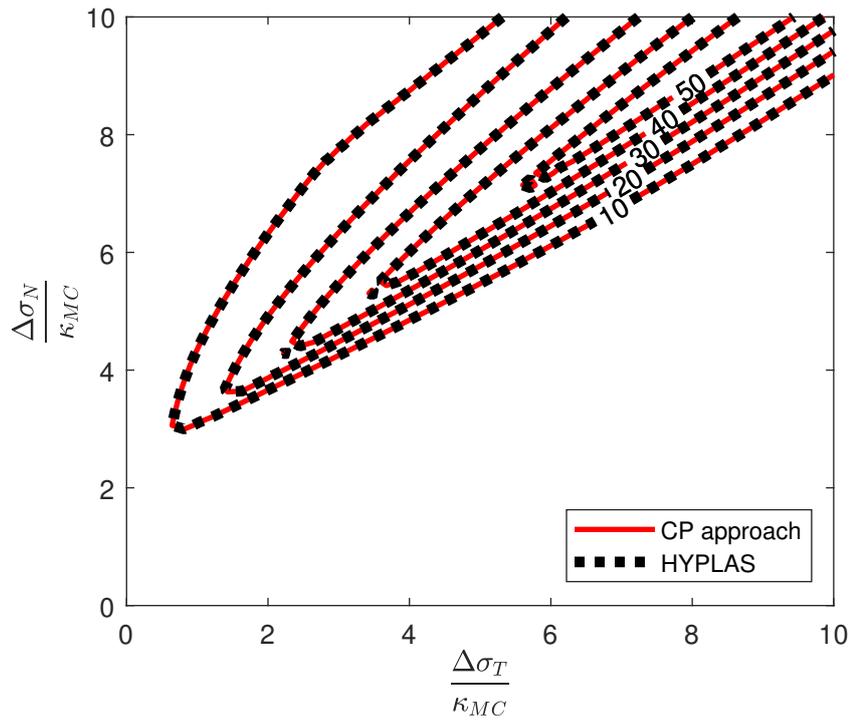


Figure 6.5: *Iso-error map* for the Mohr-Coulomb model departing from point B.

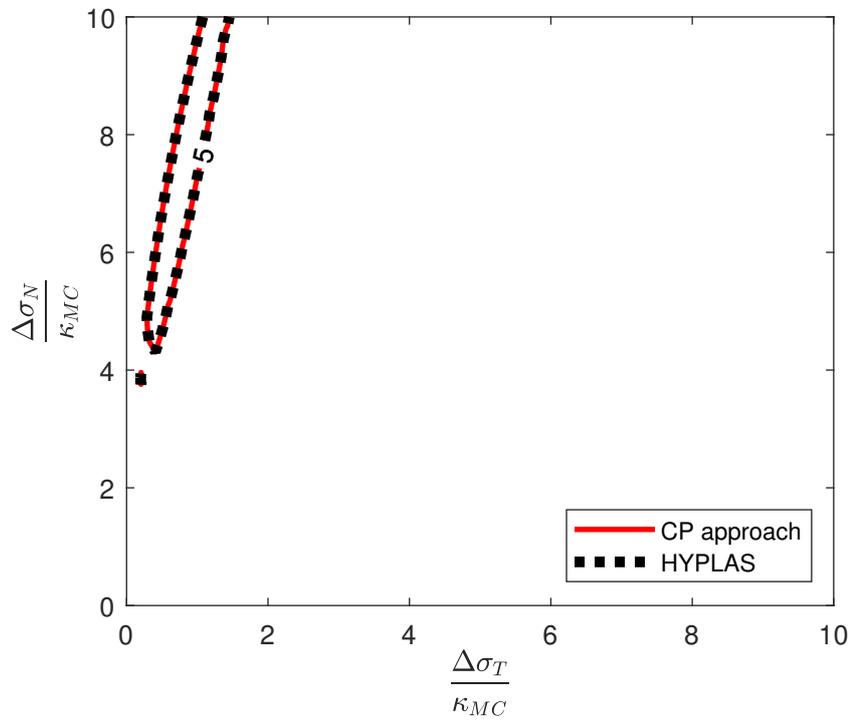


Figure 6.6: *Iso-error map* for the Mohr-Coulomb model departing from point C.

Similar to the experiment using the Drucker-Prager model, both the integration algorithms yield virtually the same results.

6.2

Boundary value problems

In this section, assessment of the performance and accuracy of the proposed methodology is carried out by solving numerical examples commonly found in the literature. In all the cases, the full N-R scheme is employed for obtaining the global solution of the nonlinear equilibrium equations, whereas the discrete local constitutive IVPs are solved via the conic optimization problem (5-12) using the general-purpose solver *MOSEK* [52]. The convergence criteria is given as in Eq. (4-40) with a numerical tolerance of $\epsilon_{\text{tol}} \leq 1.0 \times 10^{-8}$.

Strip-footing collapse

In this example, the elastoplastic analysis of a strip-footing on frictional-cohesive soil is performed in order to determine its bearing capacity. The footing has a width of $B = 2\text{m}$ and is subjected to a vertical pressure of p . Additionally, the footing is long enough so that a *plane strain* analysis may be conducted. The soil is assumed as an infinite medium and is modeled as a weightless Mohr-Coulomb perfectly plastic material with an associative flow rule, elastic modulus $E = 3000\text{kPa}$, Poisson's ratio $\nu = 0.3$, frictional angle $\phi = 30^\circ$, and cohesion $c = 1\text{kPa}$. The corresponding finite-element model is shown in Figure 6.7, where only one half of the cross section is discretized, owing to the symmetry of the problem. The mesh consists of 1033 nodes and 324 eight-noded quadrilaterals with reduced integration (2×2 Gaussian points) to avoid *volume locking*. The footing is assumed to be rigid and the footing/soil interface is assumed to be frictionless. The analysis is conducted by prescribing the vertical displacements of the nodes under the footing corresponding to the settlement of the soil. The pressure p is then determined as the total vertical reaction under the footing divided by the width B . Two incremental analyses are performed by applying a total displacement of $u = 0.5\text{m}$ in 25 and 50 equal increments. The analytical limit load for this problem is given by the well-known Prandtl's solution [14], which is expressed as

$$p_{lim} = c \left(\tan^2 \left(\frac{\pi}{4} + \frac{\phi}{2} \right) e^{\pi \tan \phi} - 1 \right) \cot \phi. \quad (6-5)$$

For the present parameters, the Prandtl's solution predicts a limit pressure of $p_{lim} = 30.1396\text{kN/m}^2$.

The load-displacement curve obtained in the analysis, depicted in Figure

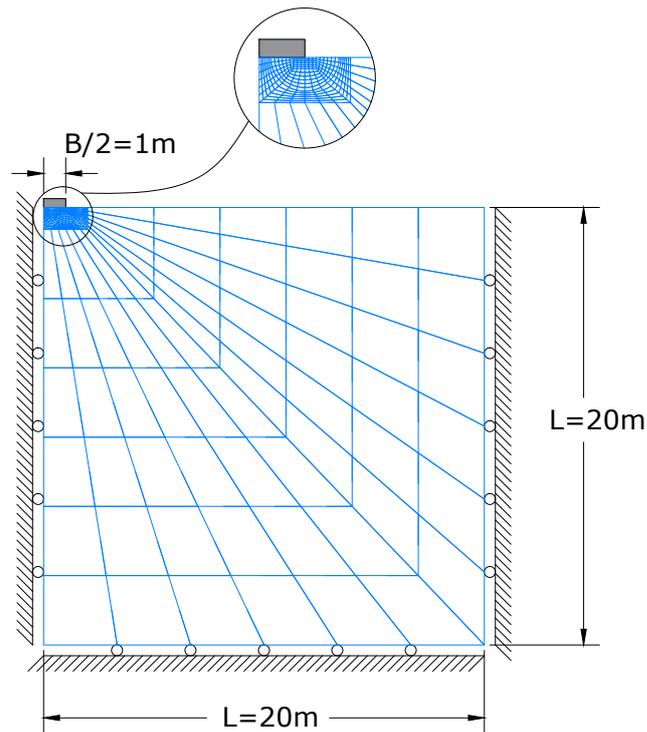


Figure 6.7: Strip-footing finite-element model.

6.8, shows that the numerical solution is in excellent agreement with the analytical limit load. Moreover, the accordance between the two analyses shows that 25 increments are sufficient for attaining convergence of the numerical solution. The error between the analytical and numerical solutions ($p_{lim}^N = 30.8953\text{kN}$) is approximately equal to 2.51%.

Collapse of end-loaded cantilever

In this example, the plastic collapse of a cantilever beam subjected to a vertical load f at its free end is studied. The beam is $L = 1000\text{mm}$ long and has a rectangular cross-section with a width of $b = 50\text{mm}$ and a height of $h = 100\text{mm}$. The finite-element model employed in the elastoplastic analysis is depicted in Figure 6.9. The mesh consists of 729 nodes and 640 four-noded quadrilateral elements. Owing to the geometry and boundary conditions, it is expected that the out-of-plane stresses are small compared with the in-plane ones; thus, a *plane stress* analysis is conducted. The *plane stress* state does not require any special treatment for the incompressibility constraints. Hence, a full integration scheme (2×2 Gaussian points) is employed. The beam is modeled as a von Mises perfectly plastic material with an associative flow rule, elastic modulus $E = 210\text{GPa}$, Poisson's ratio $\nu = 0.3$, and yield limit $\sigma_y = 0.24\text{GPa}$. The analysis is performed by increasing the load until collapse occurs and

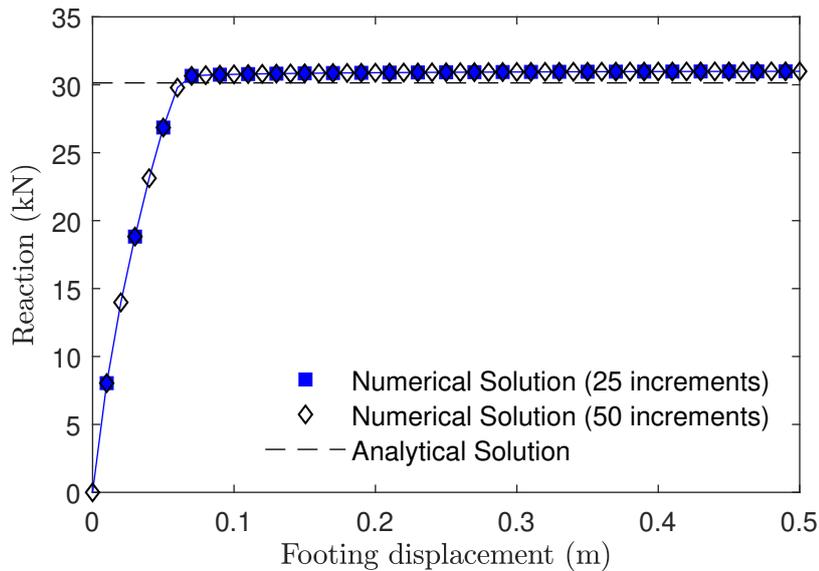


Figure 6.8: Strip-footing load-displacement curve.

the equilibrium can no longer be found for relatively small load increments. According to [70], considering the *plastic hinge* approach, the analytical limit load for this problem is given as

$$f_{lim} = \frac{\sigma_y b h^2}{4L}. \quad (6-6)$$

For the present parameters, the analytical limit load is $f_{lim} = 30\text{kN}$.

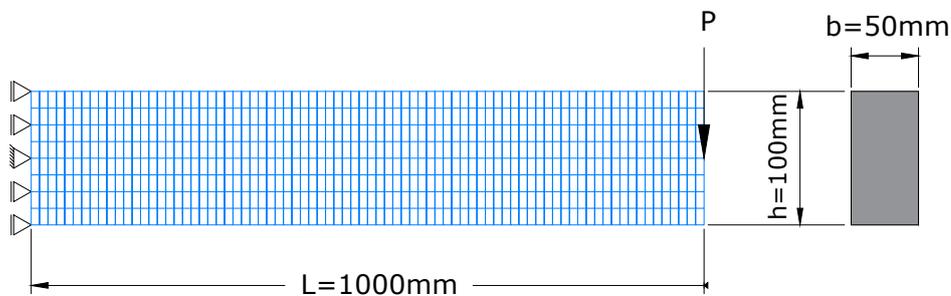


Figure 6.9: End-loaded cantilever finite-element model.

Figure 6.10 shows the load-displacement curve obtained in the analysis. The numerical limit load obtained via the analysis $f_{lim}^N = 30.79\text{kN}$ is in excellent agreement with the analytical solution, considering the plastic hinge approach.

Stretching of perforated rectangular plate

In this classical example, the stretching of a thin perforated plate is studied. The plate has a center hole with a radius of $r = 5\text{mm}$, a width

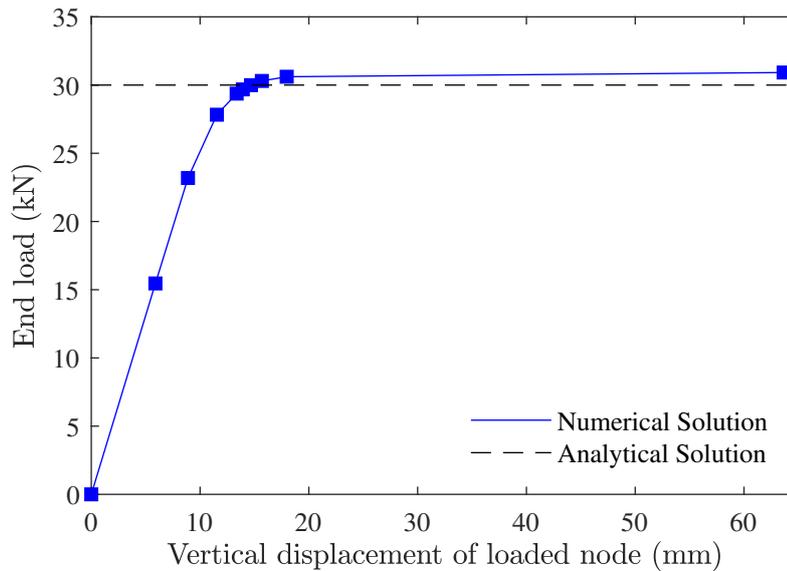


Figure 6.10: End-loaded cantilever load-displacement curve.

of $w = 36\text{mm}$, a height of $h = 20\text{mm}$, and a depth of $t = 1\text{mm}$. The stretching occurs along its longitudinal axis owing to the prescribed edge displacements. Because of the geometry and boundary conditions, a *plane stress* state is assumed throughout the analysis. The finite-element model employed in the analysis is depicted in Figure 6.11. Only one quarter of the model is discretized, owing to the double symmetry of the problem. The mesh contains 861 nodes and 800 four-noded quadrilateral elements. A linearly hardening von Mises model with the associative flow rule, elastic modulus $E = 70\text{GPa}$, Poisson's ratio $\nu = 0.2$, yield limit $\sigma_y = 0.243\text{GPa}$, and hardening modulus $H = 0.2\text{GPa}$ is adopted. The analysis is performed by applying a total horizontal displacement of $u = 6.15\text{mm}$ in 22 increments.

The analysis yields the load-displacement curve shown in Figure 6.12, where the load is obtained as the total reaction force at the edge. The evolution of the plastic strains is illustrated in Figure 6.13, where the accumulated plastic strain $\bar{\epsilon}^p$ is plotted for different steps of the analysis. Such spreading of the plastic front is in excellent agreement with that reported in [62].

The normalized residuals obtained during the global N-R iterations are shown in Table 6.1, indicating the accuracy of the numerical scheme for the evaluation of the CETO. The exponential decay of the residuals shows that the numerical evaluation of the CETO is sufficiently accurate to achieve convergence in the N-R scheme. In fact, the convergence rate is in agreement with the one presented in [71].

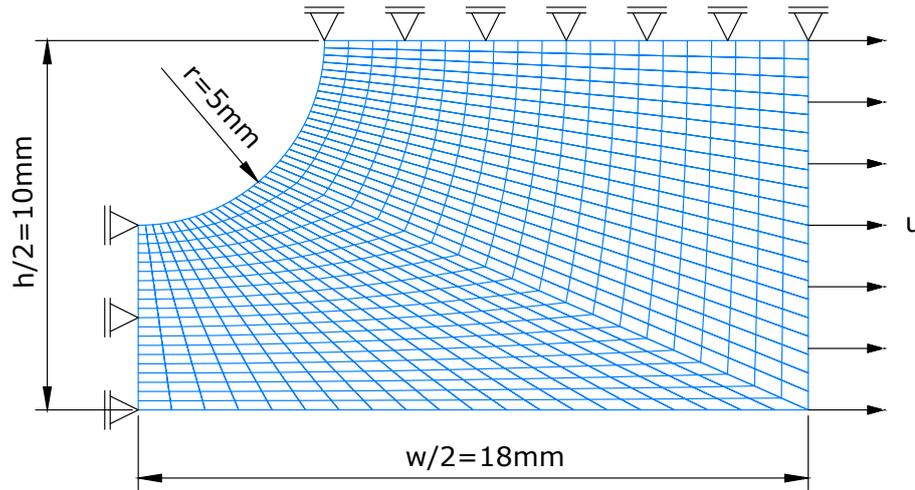


Figure 6.11: Perforated plate finite-element model.

Table 6.1: Stretching of a perforated plate example normalized residuals

Iteration	Increment 9	Increment 10	Increment 11
1	0.31×10^{-2}	0.20×10^{-2}	0.25×10^{-2}
2	7.84×10^{-4}	1.99×10^{-4}	2.28×10^{-5}
3	1.38×10^{-11}	1.73×10^{-12}	1.42×10^{-11}

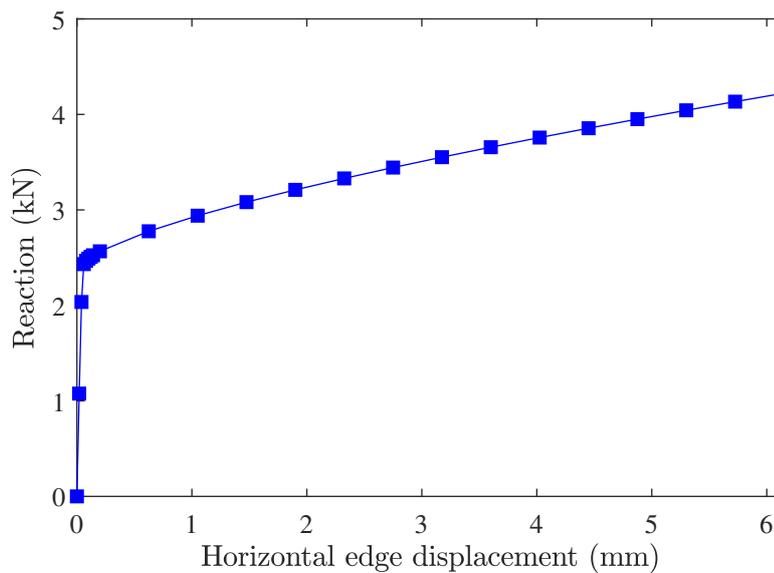


Figure 6.12: Perforated plate load-displacement curve.

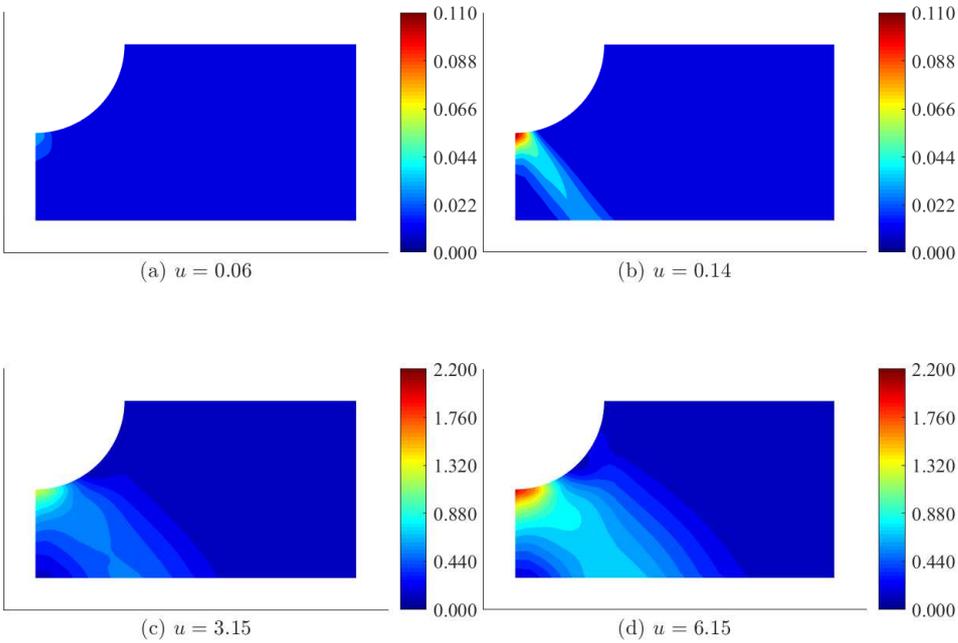


Figure 6.13: Perforated plate plastic strain evolution.

7

Conclusions and future work

7.1

Conclusions

Considering associative plasticity and a linear isotropic hardening model, the proposed methodology consists of a general framework for the solution of local constitutive equations using a fully implicit scheme. Other hardening models, such as kinematic and nonlinear, may also be considered. In fact, any hardening model that can be represented by a conic representable *potential* function, i.e., a function which gives the hardening law when differentiated with respect to the internal variables, can be integrated into our framework. Although RMA can find analytical solutions for the model and yield criteria considered in this work, there are several advantages of the CP approach over conventional RMA.

First, the proposed approach is general in the sense that non-smooth and multisurface yield criteria are naturally accommodated in the formulation. Additionally, in contrast to classical *return-mapping algorithms*, a dedicated implementation is not needed for each yield criterion. In fact, the structure of the program for the solution of elastoplastic constitutive IVPs via the CP problem (5-12) remains the same; only the conic constraint representation of each yield criterion is modified. Many of these representations are already available in the literature [55]. However, any conic representable yield criterion may systematically be cast into a conic form by consulting dictionaries of conic representable function and sets [40, 49, 42, 43]. Furthermore, constrained stress states can be easily implemented by suppressing the corresponding stress variables in the optimization problem formulation, whereas in the classical RMA, such a case requires special treatment.

Second, as noted in [72], to ensure the strictly positive condition of the *plastic multiplier*, i.e., $\Delta\gamma > 0$, RMAs must artificially identify the set of active constraints. On the other hand, in the presented approach, this condition is enforced naturally as the KKT conditions of the CP problem (5-12).

Lastly, the proposed scheme for the numerical evaluation of the CETO relieves the burden of finding the analytical derivatives for the implicit RMA.

Additionally, the proposed scheme is sufficiently accurate to obtain a monotonic decay of residuals owing to the out-of-balance forces.

The excellent agreement between the numerical and analytical solutions indicates that the proposed methodology is capable of solving elastoplastic problems involving many of the most commonly found yield criteria in the literature. In [55], Bisbos et al. show that many complex yield functions, including anisotropic and multisurface criteria, can be cast into second-order and/or semidefinite constraints. Therefore, complex constitutive models can be easily implemented by employing the conic representations of the corresponding yield criteria.

7.2

Future work

The theory on CP problems involving *power* and *exponential* cones has been established over the last decades, notably in [73] and [28]. Such works propose the extension of primal-dual interior-point algorithms for unsymmetric cone optimization by devising new self-concordant barrier functions. Despite such theoretical developments, it was only recently that efficient software for conic optimization over *power* and *exponential* cones became available (e.g., [29], [30]). These new classes of CP methods open the possibility to include additional sets of problems which may be cast and solve as conic programs. Nonetheless, to the knowledge of this author, the employment of such modern methods in plasticity remains an open field.

Within the context of this work, one of the benefits of considering these new classes of cones is the extension of the proposed framework for other yield criteria (e.g. Cam-Clay [74], Lade-Duncan [75], Matsuoka-Nakai [76]). Additionally, by taking the *exponential cone* into account, it is possible to formulate a nonlinear hardening curve based on the so-called *modified power law* [77]. In the following, the casting of the Cam-Clay yield criterion as a mixed *exponential second-order* conic constraint is presented. Subsequently, the consideration of a nonlinear hardening is introduced and the particular case of the *modified power law* is formulated.

Exponential cone representable yield criteria

The exponential cone comprises a subset of \mathbb{R}^3 defined as

$$K_{\text{exp}} = \text{cl} \left\{ \mathbf{x} \in \mathbb{R}^3 \mid x_1 \geq x_2 e^{\frac{x_3}{x_2}}, x_1, x_2 > 0 \right\} \quad (7-1)$$

where $\text{cl} \{ \cdot \}$ denotes the closure of set $\{ \cdot \}$.

A particular instance of inequalities which may be represented as a exponential conic constraint is related to the *entropy function*, defined as

$$H(x) = x \log(x). \quad (7-2)$$

The sublevel set of such function may be directly represented by an exponential cone, i.e.,

$$\begin{aligned} H(x) \leq t &\iff x \log(x) \leq t \\ &\iff -x \log(x) \geq -t \\ &\iff \log\left(\frac{1}{x}\right) \geq \frac{-t}{x} \\ &\iff \frac{1}{x} \geq e^{\frac{-t}{x}} \\ &\iff 1 \geq x e^{\frac{-t}{x}} \\ &\iff (1, x, -t) \in K_{\text{exp}} \end{aligned} \quad (7-3)$$

In particular, this representation allows to cast the Cam-Clay yield criterion [74] as a mixed exponential second-order conic constraint. In terms of the von Mises effective stress $q = \sqrt{3J_2}$ and pressure p , the Cam-Clay yield criterion is defined as

$$f(p, q, p_c) = q + Mp \log\left(-\frac{p}{p_c}\right) \leq 0, \quad (7-4)$$

where M and p_c are positive material constants and the yield criteria is define only for negative pressure states, i.e., $p \leq 0$.

Splitting this inequality in

$$q \leq y \quad (7-5)$$

and

$$Mp \log\left(-\frac{p}{p_c}\right) \leq -y, \quad (7-6)$$

its is possible recast the Cam-Clay criterion as the intersection of a second-order cone with an exponential cone.

Firstly, just as in Section 3.1, it is clear that inequality (7-5) may be represented as the following second-order cone

$$\begin{aligned} q = \sqrt{J_2(\boldsymbol{\Sigma})} \leq y &\iff \sqrt{3}\|\mathbf{s}\| \leq y \\ &\iff \|\mathbf{s}\| \leq \frac{y}{\sqrt{3}} \\ &\iff \left(\frac{y}{\sqrt{3}}, \mathbf{s}\right) \in \mathcal{C}^{\text{SOC}} \end{aligned} \quad (7-7)$$

with \mathbf{s} defined in Eq. (3-27).

As for inequality (7-6), a exponential conic representation is possible, i.e.,

$$\begin{aligned}
Mp \log \left(-\frac{p}{p_c} \right) \leq -y &\iff -Mp \log \left(-\frac{p}{p_c} \right) \geq y \\
&\iff Mp \log \left(-\frac{p_c}{p} \right) \geq y \\
&\iff -\log \left(-\frac{p_c}{p} \right) \geq -\frac{y}{Mp} \\
&\iff p_c \geq -pe^{-\frac{y}{Mp}} \\
&\iff \left(p_c, -p, \frac{y}{M} \right) \in \mathcal{K}_{\text{exp}}
\end{aligned} \tag{7-8}$$

Therefore, the Cam-Clay yield criterion may be cast into the following convex cone

$$\mathcal{M}_{CC} = \left\{ (\boldsymbol{\sigma}, y) \in \mathbb{R}^6 \times \mathbb{R} \mid \left(\frac{y}{\sqrt{3}}, \mathbf{s} \right) \in \mathcal{C}^{SOC}, \left(p_c, -p, \frac{y}{M} \right) \in \mathcal{K}_{\text{exp}} \right\} \tag{7-9}$$

where

$$\mathbf{s} = \mathbf{L}\boldsymbol{\sigma}, \quad p = \mathbf{b}^T \boldsymbol{\sigma}, \tag{7-10}$$

with \mathbf{L} and \mathbf{b} defined as in Eqs. (3-23) and (3-25), respectively.

Nonlinear hardening laws

The functional in Eq. (3-66), which characterize the *closest point projection* problem of Eq. (3-65), was derived by assuming a linear relation between *hardening forces* and *internal variables*, as depicted in Eq. (3-42). If such assumption is dropped, it is still possible to consider the same convex optimization, as long as the objective function is changed accordingly. Specifically, if a general nonlinear hardening law as

$$\boldsymbol{\Theta} := \mathcal{H}(\boldsymbol{\alpha}) \tag{7-11}$$

is considered, then the functional

$$\mathcal{F}(\boldsymbol{\Sigma}, \boldsymbol{\Theta}) = \frac{1}{2} (\boldsymbol{\Sigma}_{n+1}^{\text{trial}} - \boldsymbol{\Sigma}) : \mathbf{C}^{-1} : (\boldsymbol{\Sigma}_{n+1}^{\text{trial}} - \boldsymbol{\Sigma}) + \mathcal{G}(\boldsymbol{\Theta}) \tag{7-12}$$

must be constructed such that the KKT conditions of problem (3-65) yield the elastoplastic equations of (3-69). For such purpose, the functional $\mathcal{G}(\boldsymbol{\Theta})$ must satisfy the following condition

$$\left. \frac{\partial \mathcal{G}(\boldsymbol{\Theta})}{\partial \boldsymbol{\Theta}} \right|_{n+1} = \boldsymbol{\alpha}_{n+1} - \boldsymbol{\alpha}_n. \tag{7-13}$$

In the following, a nonlinear isotropic hardening model based on a *modified power law* [77] is considered and the corresponding functional is

formulated.

Modified power-law isotropic hardening model

Consider a hardening curve such as in Eq. (3-45) with the following exponential hardening law

$$\theta(\bar{\varepsilon}^p) = \kappa_{\text{lim}} \left(1 - e^{-b\bar{\varepsilon}^p}\right) \quad (7-14)$$

where κ_{lim} and b are material constants which may be calibrated according to experimental data. In terms of the variables of Section 3.3, such model is represented as

$$\boldsymbol{\alpha} = \{\bar{\varepsilon}^p\}, \quad \Theta = \{\theta(\bar{\varepsilon}^p)\}. \quad (7-15)$$

In this setting, the functional to be constructed must satisfy

$$\left. \frac{\partial \mathcal{G}(\theta)}{\partial \theta} \right|_{n+1} = \bar{\varepsilon}_{n+1}^p - \bar{\varepsilon}_n^p. \quad (7-16)$$

Owing to the strict monotonicity of Eq. (7-14), it is possible to obtain the inverse relation, i.e.,

$$\theta^{-1} = \bar{\varepsilon}^p(\theta) = \frac{1}{b} \log \left(\frac{\kappa_{\text{lim}}}{\kappa_{\text{lim}} - \theta} \right). \quad (7-17)$$

Substitution into Eq. (7-16) yield the following ordinary differential equation

$$\left. \frac{\partial \mathcal{G}(\theta)}{\partial \theta} \right|_{n+1} = \frac{1}{b} \log \left(\frac{\kappa_{\text{lim}}}{\kappa_{\text{lim}} - \theta} \right) - \bar{\varepsilon}_n^p. \quad (7-18)$$

which upon integration with respect to θ gives

$$\mathcal{G}(\theta) = \frac{1}{b} (\theta - \kappa_{\text{lim}}) \log \left(\frac{\kappa_{\text{lim}}}{\kappa_{\text{lim}} - \theta} \right) + \theta(1 - \bar{\varepsilon}_n^p). \quad (7-19)$$

Therefore, the required objective function is defined as

$$\mathcal{F}(\boldsymbol{\Sigma}, \theta) = \frac{1}{2} (\boldsymbol{\Sigma}_{n+1}^{\text{trial}} - \boldsymbol{\Sigma}) : \mathbf{C}^{-1} : (\boldsymbol{\Sigma}_{n+1}^{\text{trial}} - \boldsymbol{\Sigma}) + \mathcal{G}(\theta) \quad (7-20)$$

with $\mathcal{G}(\theta)$ given in Eq. (7-19).

The corresponding convex optimization problem may be expressed as

$$\begin{aligned} \min_{\boldsymbol{\sigma}, \theta, y_1, y_2} \quad & y_1 + y_2 + \theta(1 - \bar{\varepsilon}_n^p) \\ \text{s.t.} \quad & \frac{1}{2} (\boldsymbol{\sigma}_{n+1}^{\text{trial}} - \boldsymbol{\sigma})^T \mathbf{S} (\boldsymbol{\sigma}_{n+1}^{\text{trial}} - \boldsymbol{\sigma}) \leq y_1 \\ & \frac{1}{b} (\theta - \kappa_{\text{lim}}) \log \left(\frac{\kappa_{\text{lim}}}{\kappa_{\text{lim}} - \theta} \right) \leq y_2 \\ & (\boldsymbol{\sigma}, \theta) \in \mathcal{M} \end{aligned} \quad (7-21)$$

As aforementioned, the first inequality may be directly cast into a second-order conic constraint. As for the second inequality, according to Eq. (7-3), it can be cast into the following exponential conic constraint

$$\frac{1}{b}(\theta - \kappa_{lim}) \log \left(\frac{\kappa_{lim}}{\kappa_{lim} - \theta} \right) \leq y_2 \iff (k_{lim}, k_{lim} - \theta, -by_2) \in \mathcal{K}_{exp}. \quad (7-22)$$

Therefore, it is straightforward to cast the convex programming problem (7-21) into a mixed exponential second-order cone program. The optimal point to such problem yields the solution of the discrete elastoplastic IVP (3-59) under the nonlinear hardening law of Eq. (7-14).

Bibliography

- [1] CHEN, W. F.; HAN, D. J.. **Plasticity for Structural Engineers**. Springer-Verlag, 1987.
- [2] DRUCKER, D.. **Limit analysis of two and three dimensional soil mechanics problems**. *Journal of the Mechanics and Physics of Solids*, 1(4):217–226, jul 1953.
- [3] GREENBERG, H. J.; PRAGER, W.. **Limit Design of Beams and Frames**. *Transactions of the American Society of Civil Engineers*, 117(1):447–458, 1952.
- [4] DRUCKER, D.; PRAGER, W.. **Soil Mechanics and Plastic Analysis or Limit Design**. *Quarterly of Applied Mathematics*, 10(2):157–165, 1952.
- [5] HILL, R.. **LXXXVIII. On the state of stress in a plastic-rigid body at the yield point**. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 42(331):868–875, aug 1951.
- [6] KOITER, W. T.. **Stress-strain relations, uniqueness and variational theorems for elastic-plastic materials with a singular yield surface**. *Quarterly of Applied Mathematics*, 11(3):350–354, oct 1953.
- [7] DE SOUZA NETO, E. A.; PERI, D. ; OWEN, D. R. J.. **Computational Methods for Plasticity**. John Wiley & Sons, Ltd, Chichester, UK, oct 2008.
- [8] SIMO, J.; HUGHES, T.. **Computational Inelasticity**, volumen 7 de **Interdisciplinary Applied Mathematics**. Springer-Verlag, New York, 1998.
- [9] KRABBENHØFT, K.; LYAMIN, A. V. ; SLOAN, S. W.. **Three-dimensional Mohr-Coulomb limit analysis using semidefinite programming**. *Communications in Numerical Methods in Engineering*, 24(11):1107–1119, jun 2007.
- [10] MAKRODIMOPOULOS, A.. **Computational formulation of shake-down analysis as a conic quadratic optimization problem**. *Mechanics Research Communications*, 33(1):72–83, jan 2006.

- [11] KRABBENHØFT, K.; LYAMIN, A. V. ; SLOAN, S. W.. **Formulation and solution of some plasticity problems as conic programs.** International Journal of Solids and Structures, 44(5):1533–1549, 2007.
- [12] MAIER, G.. **Quadratic programming and theory of elastic-perfectly plastic structures.** Meccanica, 3(4):265–273, dec 1968.
- [13] WIENERS, C.. **Nonlinear solution methods for infinitesimal perfect plasticity.** ZAMM, 87(8-9):643–660, sep 2007.
- [14] KRABBENHOFT, K.; LYAMIN, A. V.; SLOAN, S. W. ; WRIGGERS, P.. **An interior-point algorithm for elastoplasticity.** International Journal for Numerical Methods in Engineering, 69(3):592–626, jan 2007.
- [15] ÉBOLI, C. R.. **Análise Elasto-Plástica de Lajes via Programação Matemática.** PhD thesis, PUC-Rio, 1994.
- [16] SIMO, J. C.; HUGHES, T. J. R.. **On the Variational Foundations of Assumed Strain Methods.** Journal of Applied Mechanics, 53(1):51, 1986.
- [17] KRABBENHOFT, K.; LYAMIN, A. V.; SLOAN, S. W. ; WRIGGERS, P.. **An interior-point algorithm for elastoplasticity.** International Journal for Numerical Methods in Engineering, 69(3):592–626, jan 2007.
- [18] PEREIRA, N. Z.; BORGES, L. A. ; HECKE, M. B.. **A force method for elastic-plastic analysis of frames by quadratic optimization.** International Journal of Solids and Structures, 24(2):211–230, jan 1988.
- [19] CHRISTENSEN, P. W.. **A nonsmooth Newton method for elastoplastic problems.** Computer Methods in Applied Mechanics and Engineering, 191(11-12):1189–1219, 2002.
- [20] PANG, J. S.. **Newton’s Method for B-differentiable Equations.** Math. Oper. Res., 15(2):311–341, 1990.
- [21] LUIS MULLER, A.. **Análise numérica da estabilidade de poços de petróleo considerando a variabilidade espacial e acoplamento fluido-mecânico.** PhD thesis, Pontifícia Universidade Católica Do Rio De Janeiro, Rio de Janeiro, Brazil, apr 2007.
- [22] ZHANG, L. L.; LI, J. Y.; ZHANG, H. W. ; PAN, S. H.. **A second order cone complementarity approach for the numerical solution of elastoplasticity problems.** Computational Mechanics, 51(1):1–18, 2013.

- [23] PAN, S.; CHEN, J.-S.. **A semismooth Newton method for SOCCPs based on a one-parametric class of SOC complementarity functions.** *Computational Optimization and Applications*, 45(1):59–88, 2010.
- [24] YONEKURA, K.; KANNO, Y.. **Second-order cone programming with warm start for elastoplastic analysis with von Mises yield criterion.** *Optimization and Engineering*, 13(2):181–218, jun 2012.
- [25] BISBOS, C. D.; MAKRODIMOPOULOS, A. ; PARDALOS, P. M.. **Second-order cone programming approaches to static shakedown analysis in steel plasticity.** *Optimization Methods and Software*, 20(1):25–52, feb 2005.
- [26] ANDERSEN, E.; ROOS, C. ; TERLAKY, T.. **On implementing a primal-dual interior-point method for conic quadratic optimization.** *Mathematical Programming*, 95(2):249–277, feb 2003.
- [27] BORCHERS, B.. **CSDP, A C library for semidefinite programming.** *Optimization Methods and Software*, 11(1-4):613–623, jan 1999.
- [28] SERRANO, S. A.. **Algorithms for unsymmetric cone optimization and an implementation for problems with the exponential cone.** PhD thesis, Stanford, 2015.
- [29] DAHL, J.; ANDERSEN, E. D.. **A primal-dual interior-point algorithm for nonsymmetric exponential-cone optimization.** Unpublished work, 2019.
- [30] O'DONOGHUE, B.; CHU, E.; PARIKH, N. ; BOYD, S.. **Conic Optimization via Operator Splitting and Homogeneous Self-Dual Embedding.** *Journal of Optimization Theory and Applications*, 169(3):1042–1068, 2016.
- [31] ANDERSEN, K. D.; CHRISTIANSEN, E. ; OVERTON, M. L.. **Computing Limit Loads by Minimizing a Sum of Norms.** *SIAM Journal on Scientific Computing*, 19(3):1046–1062, may 1998.
- [32] MARTIN, C. M.; MAKRODIMOPOULOS, A.. **Finite-Element Limit Analysis of Mohr–Coulomb Materials in 3D Using Semidefinite Programming.** *Journal of Engineering Mechanics*, 134(4):339–347, apr 2008.
- [33] MAKRODIMOPOULOS, A.; MARTIN, C. M.. **Upper bound limit analysis using simplex strain elements and second-order cone**

- programming**. International Journal for Numerical and Analytical Methods in Geomechanics, 31(6):835–865, may 2007.
- [34] MAKRODIMOPOULOS, A.; MARTIN, C. M.. **Upper bound limit analysis using simplex strain elements and second-order cone programming**. International Journal for Numerical and Analytical Methods in Geomechanics, 31(6):835–865, may 2007.
- [35] CIRIA, H.; PERAIRE, J. ; BONET, J.. **Mesh adaptive computation of upper and lower bounds in limit analysis**. International Journal for Numerical Methods in Engineering, 75(8):899–944, aug 2008.
- [36] DURAND, A. F.; VARGAS, E. A. ; VAZ, L. E.. **Applications of numerical limit analysis (NLA) to stability problems of rock and soil masses**. International Journal of Rock Mechanics and Mining Sciences, 43(3):408–425, 2006.
- [37] BORGES, L.; ZOUAIN, N. ; HUESPE, A.. **Nonlinear optimization procedure for limit analysis**. European Journal of Mechanics. A. Solids, 15:203–250, 01 1996.
- [38] MAKRODIMOPOULOS, A.; BISBOS, C.. **Shakedown analysis of plane stress problems via SOCP**, p. 185–216. John von Neumann Institute, Jülich, 2003.
- [39] BISBOS, C. D.. **Semidefinite optimization models for limit and shakedown analysis problems involving matrix spreads**. Optimization Letters, 1(1):101–109, nov 2006.
- [40] BEN-TAL, A.; NEMIROVSKI, A.. **Lectures on Modern Convex Optimization**. Society for Industrial and Applied Mathematics, jan 2001.
- [41] NOCEDAL, J.; WRIGHT, S. J.. **Numerical Optimization**. Springer Series in Operations Research and Financial Engineering. Springer New York, 2006.
- [42] ALIZADEH, F.; GOLDFARB, D.. **Second-order cone programming**. Mathematical Programming, 95(1):3–51, jan 2003.
- [43] ALIZADEH, F.. **Interior Point Methods in Semidefinite Programming with Applications to Combinatorial Optimization**. SIAM Journal on Optimization, 5(1):13–51, feb 1995.

- [44] BRUNO, H.; BARROS, G.; MENEZES, I. F. ; MARTHA, L. F.. **Return-Mapping Algorithms for Associative Isotropic Hardening Plasticity Using Conic Optimization**. *Applied Mathematical Modelling*, 78:724–748, 2019.
- [45] FIACCO, A. V.. **Introduction to Sensitivity and Stability Analysis in Nonlinear Programming**. Academic Press, 1983.
- [46] MILLER, S. A.. **Sensitivity of Solutions to Semidefinite Programs**. Technical report, University of California, Santa Barbara, CA, 1997.
- [47] SYDSÆTER, K.; STRØM, A. ; BERCK, P.. **Kronecker products and the vec operator. Differentiation of vectors and matrices**. In: *ECONOMISTS' MATHEMATICAL MANUAL*, p. 155–158. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.
- [48] NESTEROV, Y. E.; TODD, M. J.. **Self-Scaled Barriers and Interior-Point Methods for Convex Programming**. *Mathematics of Operations Research*, 22(1):1–42, feb 1997.
- [49] LOBO, M. S.; VANDENBERGHE, L.; BOYD, S. ; LEBRET, H.. **Applications of second-order cone programming**. *Linear Algebra and its Applications*, 284(1-3):193–228, nov 1998.
- [50] MITTELMANN, H.. **An independent benchmarking of SDP and SOCP solvers**. *Mathematical Programming*, 95(2):407–430, feb 2003.
- [51] BOYD, S.; VANDENBERGHE, L.. **Convex optimization theory**, volumen 1. Cambridge University Press, jun 2004.
- [52] ANDERSEN, E. D.; ANDERSEN, K. D.. **The Mosek Interior Point Optimizer for Linear Programming: An Implementation of the Homogeneous Algorithm**. In: Frenk, H.; Roos, K.; Terlaky, T. ; Zhang, S., editors, *HIGH PERFORMANCE OPTIMIZATION*, p. 197–232. Springer US, Boston, MA, 2000.
- [53] STURM, J. F.. **Using SeDuMi 1.02, A Matlab toolbox for optimization over symmetric cones**. *Optimization Methods and Software*, 11(1-4):625–653, jan 1999.
- [54] TÜTÜNCÜ, R. H.; TOH, K. C. ; TODD, M. J.. **Solving semidefinite-quadratic-linear programs using SDPT3**. *Mathematical Programming*, 95(2):189–217, feb 2003.

- [55] BISBOS, C. D.; PARDALOS, P. M.. **Second-Order Cone and Semidefinite Representations of Material Failure Criteria**. *Journal of Optimization Theory and Applications*, 134(2):275–301, aug 2007.
- [56] R. O., D.; A. P. S., S.. **Plasticity and Geomechanics**. Cambridge University Press, 2002.
- [57] DUVAUT, G.; LIONS, J. L.. **Inequalities in Mechanics and Physics**, volumen 219 de **Grundlehren der mathematischen Wissenschaften**. Springer Berlin Heidelberg, Berlin, Heidelberg, 1976.
- [58] JIRÁSEK, M.; BAZANT, Z. P.. **Inelastic Analysis of Structures**. Wiley, West Sussex, 1st edition, 2002.
- [59] HAN, W.; REDDY, B. D.. **Plasticity**, volumen 9 de **Interdisciplinary Applied Mathematics**. Springer New York, New York, NY, 2013.
- [60] ROCKAFELLAR, R.. **Book - Convex analysis**. Princeton University Press, 1997.
- [61] ORTIZ, M.; POPOV, E. P.. **Accuracy and stability of integration algorithms for elastoplastic constitutive relations**. *International Journal for Numerical Methods in Engineering*, 21(9):1561–1576, sep 1985.
- [62] FUSCHI, P.; PERIĆ, D. ; OWEN, D.. **Studies on generalized mid-point integration in rate-independent plasticity with reference to plane stress J2-flow theory**. *Computers & Structures*, 43(6):1117–1133, jun 1992.
- [63] ASCHER, U.; PETZOLD, L. R.. **Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations**, volumen 1. Society for Industrial and Applied Mathematics, 1998.
- [64] KRIEG, R. D.; KRIEG, D. B.. **Accuracies of Numerical Solution Methods for the Elastic-Perfectly Plastic Model**. *Journal of Pressure Vessel Technology*, 99(4):510–515, nov 1977.
- [65] SIMO, J. C.; TAYLOR, R. L.. **A return mapping algorithm for plane stress elastoplasticity**. *International Journal for Numerical Methods in Engineering*, 22(3):649–670, mar 1986.
- [66] ARTIOLI, E.; AURICCHIO, F. ; BEIRÃO DA VEIGA, L.. **A novel 'optimal' exponential-based integration algorithm for von-Mises plasticity with linear hardening: Theoretical analysis on yield consistency**,

- accuracy, convergence and numerical investigations. *International Journal for Numerical Methods in Engineering*, 67(4):449–498, 2006.
- [67] ARTIOLI, E.; AURICCHIO, F. ; BEIRÃO DA VEIGA, L.. **Second-order accurate integration algorithms for von-Mises plasticity with a nonlinear kinematic hardening mechanism.** *Computer Methods in Applied Mechanics and Engineering*, 196(9-12):1827–1846, 2007.
- [68] MÜLLER, A.; VARGAS, E. A.. **Stability analysis of a slope under impact of a rock block using the generalized interpolation material point method (GIMP).** *Landslides*, 16(4):751–764, apr 2019.
- [69] MANCHAO, H.; LEAL E SOUSA, R.; MÜLLER, A.; VARGAS, E.; RIBEIRO E SOUSA, L. ; XIN, C.. **Analysis of excessive deformations in tunnels for safety evaluation.** *Tunnelling and Underground Space Technology*, 45:190–202, jan 2015.
- [70] LUBLINER, J.. **Plasticity Theory.** Dover books on engineering. Dover Publications, 2008.
- [71] SIMO, J. C.; TAYLOR, R. L.. **Consistent tangent operators for rate-independent elastoplasticity.** *Computer Methods in Applied Mechanics and Engineering*, 48(1):101–118, 1985.
- [72] SCALET, G.; AURICCHIO, F.. **Computational Methods for Elastoplasticity: An Overview of Conventional and Less-Conventional Approaches.** *Archives of Computational Methods in Engineering*, 25(3):545–589, jul 2018.
- [73] CHARES, P. R.. **Cones and Interior-Point Algorithms for Structured Convex Optimization involving Powers and Exponentials.** PhD thesis, Université Catholique de Louvain, 2009.
- [74] THURAIRAJAH, A.. **Critical state soil mechanics**, volumen 16. Cambridge University Press, Cambridge, 1988.
- [75] ZHU, M. Z.; YANG, Y.; GAO, F. ; LIU, J.. **Analytical Solution of Tunnel Surrounding Rock for Stress and Displacement Based on Lade-Duncan Criterion.** *Advances in Civil Engineering*, 2018, 2018.
- [76] BORJA, R. I.; SAMA, K. M. ; SANZ, P. F.. **On the numerical integration of three- invariant elastoplastic constitutive models.** *Computer Methods in Applied Mechanics and Engineering*, 192(9-10):1227–1258, 2003.

- [77] ALLAN F. BOWER. **Applied Mechanics of Solids**. CRC Press, 2010.

Appendix A

Proof of Lemma 2-17

Proof. Owing to the Cauchy-Schwarz inequality, it follows that

$$|\mathbf{x}_{2:n}^T \mathbf{z}_{2:n}| \leq \|\mathbf{x}_{2:n}\| \|\mathbf{z}_{2:n}\| \implies \mathbf{x}_{2:n}^T \mathbf{z}_{2:n} \geq -\|\mathbf{x}_{2:n}\| \|\mathbf{z}_{2:n}\|. \quad (7-23)$$

By hypothesis $\mathbf{x}, \mathbf{z} \in \mathcal{C}^{SOC}$, hence

$$\begin{cases} \|\mathbf{x}_{2:n}\| \leq x_1 \\ \|\mathbf{z}_{2:n}\| \leq z_1 \end{cases} \implies \|\mathbf{x}_{2:n}\| \|\mathbf{z}_{2:n}\| \leq x_1 z_1. \quad (7-24)$$

Combining inequalities (7-23) and (7-24) results in

$$\mathbf{x}_{2:n}^T \mathbf{z}_{2:n} \geq -\|\mathbf{x}_{2:n}\| \|\mathbf{z}_{2:n}\| \geq -x_1 z_1. \quad (7-25)$$

On the other hand, the first equality in (2-17) requires that $\mathbf{x}_{2:n}^T \mathbf{z}_{2:n} = -x_1 z_1$; thus, the inequalities in (7-25) must be satisfied as equalities. The Cauchy-Schwarz inequality holds as an equality if and only if the two vectors are linearly dependent, hence it follows that $\mathbf{x}_{2:n} = -\alpha \mathbf{z}_{2:n}$ with $\alpha \geq 0$. Owing to the inequalities induced by the hypothesis that $\mathbf{x}, \mathbf{z} \in \mathcal{C}^{SOC}$, the second inequality in (7-25) is clearly satisfied as an equality if and only if $x_1 = \|\mathbf{x}_{2:n}\|$ and $z_1 = \|\mathbf{z}_{2:n}\|$; otherwise, it would result in a strict inequality. Therefore, we must have $x_1 = \|\mathbf{x}_{2:n}\| = \alpha \|\mathbf{z}_{2:n}\| = \alpha z_1$, i.e.,

$$\begin{aligned} \alpha = \frac{x_1}{z_1} &\implies \mathbf{x}_{2:n} + \frac{x_1}{z_1} \mathbf{z}_{2:n} = \mathbf{0} \\ &\implies z_1 \mathbf{x}_{2:n} + x_1 \mathbf{z}_{2:n} = \mathbf{0}. \end{aligned} \quad (7-26)$$

■

Appendix B

Proof of Lemma 2-26

Proof. Given that \mathbf{Z} is symmetric positive semidefinite, it may be decomposed as

$$\mathbf{Z} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T \quad (7-27)$$

where \mathbf{U} is an orthogonal matrix that collects the eigenvectors of \mathbf{Z} in a column-wise manner, and $\mathbf{\Lambda} = \text{diag}(\lambda_1(\mathbf{Z}), \dots, \lambda_6(\mathbf{Z}))$ with $\lambda_k(\mathbf{Z}) \geq 0 \quad \forall k$. Let $\mathbf{W} = \mathbf{U}^T \mathbf{X} \mathbf{U}$. Then, it follows that $\mathbf{W} \succeq 0$, and consequently, the diagonal entries of \mathbf{W} are nonnegative, i.e., $\mathbf{W}_{ii} \geq 0 \quad \forall i$. Clearly $\langle \mathbf{W}, \mathbf{\Lambda} \rangle = \langle \mathbf{X}, \mathbf{Z} \rangle = 0$. Given that $\mathbf{\Lambda}$ is a diagonal matrix, it follows that $\sum_i \mathbf{\Lambda}_{ii} \mathbf{W}_{ii} = 0$. Because the diagonal elements of both matrices are nonnegative it follows that each summand must be zero, i.e., $\mathbf{\Lambda}_{ii} \mathbf{W}_{ii} = 0 \quad \forall i$. Therefore, one of the following two alternatives must hold.

$$\begin{aligned} (i) \quad & \mathbf{\Lambda}_{ii} > 0, \quad \mathbf{W}_{ii} = 0 \\ (ii) \quad & \mathbf{W}_{ii} > 0, \quad \mathbf{\Lambda}_{ii} = 0 \end{aligned} \quad (7-28)$$

Consider an element of the product of $\mathbf{W}\mathbf{\Lambda}$, i.e., $(\mathbf{W}\mathbf{\Lambda})_{ij} = \mathbf{W}_{ij}\mathbf{\Lambda}_{jj}$. If alternative (i) holds, then $\mathbf{W}_{jj} = 0$. Consequently, given that $\mathbf{W}_{jj} = 0$ is positive semidefinite, the j -th row and column contain only 0, which implies that $(\mathbf{W}\mathbf{\Lambda})_{ij} = 0$. On the other hand, if alternative (ii) holds, i.e., $\mathbf{\Lambda}_{ii} > 0$, it follows that $(\mathbf{W}\mathbf{\Lambda})_{ij} = 0$. Therefore $\mathbf{W}\mathbf{\Lambda} = \mathbf{0}$ and, clearly, $\mathbf{X}\mathbf{Z} = \mathbf{0}$. ■

Appendix C

A minimal example

In this example, the elastoplastic analysis of a perforated circular plate under internal pressure is conducted. The metal plate has inner and outer diameters of $d_i = 20\text{cm}$ and $d_o = 20\text{cm}$, respectively. The finite-element model employed in the elastoplastic analysis is depicted in Figure 7.1. The mesh consists of 22 nodes and 10 four-noded quadrilateral elements. Owing to the geometry and boundary conditions, an axisymmetric analysis is conducted. The metal of the plate is modeled as a von Mises perfectly plastic material with an associative flow rule, elastic modulus $E = 210\text{GPa}$, Poisson's ratio $\nu = 0.3$, and yield limit $\sigma_y = 0.16\text{MPa}$. The analysis is performed by increasing the internal pressure until the limit of 12.5MPa . The input for this problem is given in Listing 7.1.

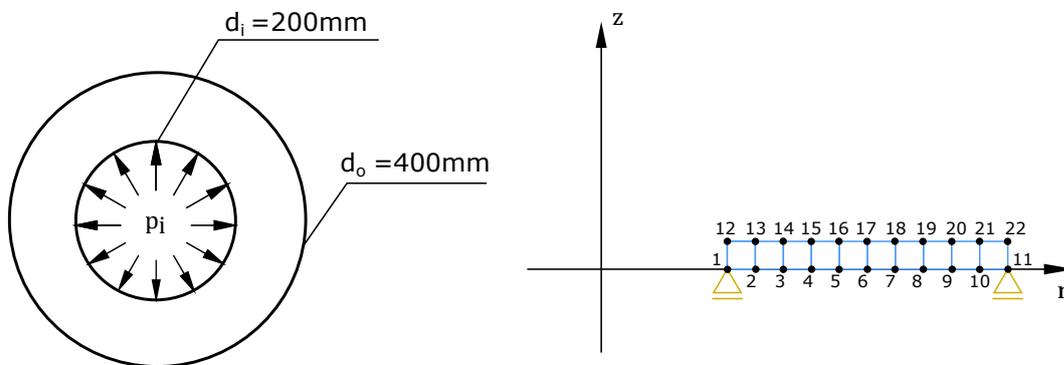


Figure 7.1: Perforated circular plate under internal pressure.

```
%-----%
%-----MESH-----%
%-----%
coords = [10 0; %pt1
          11 0; %pt2
          12 0; %pt3
          13 0; %pt4
          14 0; %pt5
          15 0; %pt6
          16 0; %pt7
```

```

17 0; %pt8
18 0; %pt9
19 0; %pt10
20 0; %pt11
10 1; %pt12
11 1; %pt13
12 1; %pt14
13 1; %pt15
14 1; %pt16
15 1; %pt17
16 1; %pt18
17 1; %pt19
18 1; %pt20
19 1; %pt21
20 1];%pt22

elem_ids = [1 2 13 12;
            2 3 14 13;
            3 4 15 14;
            4 5 16 15;
            5 6 17 16;
            6 7 18 17;
            7 8 19 18;
            8 9 20 19;
            9 10 21 20;
            10 11 22 21];

numElems = 10;
connect = cell(numElems,1);
for elem = 1:numElems
connect{elem} = Q4(3); % Q4 axisymmetric element
connect{elem}.nodes = elem_ids(elem,:); % Assign nodes ids
end
elems_prop = ones(size(connect,1),1); %Elems property ids
%-----%
%-----END OF MESH-----%
%-----%
%-----%
%-----%
%-----BOUNDARY CONDITIONS-----%
%-----%
% Displacements
dbc = [1 0 1 0 0 %Node 1 vertical constraint
      11 0 1 0 0]; %Node 11 vertical constraint

%Forces
p = 1.25e-2; %Internal pressure

```

```

ri = 10.0;    %Inner radius
fx_nodal = p*2*pi*ri*10; %Equivalent nodal force

fbc = [1  fx_nodal/2  0;
       12 fx_nodal/2  0];

%-----%
%-----END OF BOUNDARY CONDITIONS-----%
%-----%

%-----%
%-----MATERIAL PROPERTIES-----%
%-----%

props{1}.E = 210; %Elastic modulus
props{1}.v = 0.3; %Poisson coefficient
props{1}.t = 1;   %Thickness
props{1}.sy = 0.16; %Yield limit
props{1}.H = 0.05; %Hardening modulus
%Incremental elastoplastic function
props{1}.constitutive_model = @nl_vonmises_axisymmetric;
%-----%
%-----END OF MATERIAL PROPERTIES-----%
%-----%

%-----%
%-----ANALYSIS-----%
%-----%

params.max_it = 50; %Maximum iterations

params.step = [0.1;0.2;0.3;0.4;0.5;0.75;...
              0.90;0.95;0.96;0.97;0.98;0.99;1.00]; %Load steps
params.tol = fx_nodal*1e-5; %Numerical tolerance

%Main procedure
[output] = fem_nonlinear_elastoplastic_load_ctrl(coords,...
                                                  connect,...
                                                  dbc,...
                                                  fbc,...
                                                  elems_prop,...
                                                  props,...
                                                  params);
%-----%
%-----END OF ANALYSIS-----%
%-----%

```

Listing 7.1: Circular perforated plate example input.

The `@nl_vonmises_axisymmetric` function handle is referenced to the script of Listing 7.2 which consists in the computational implementation of the

state-updated procedure for the associative von Mises model by means of the methodology presented in Chapter 5.

```
function [elem_data,Cep] = ...
    nl_vonmises_axisymmetric(dlt_strain,...
        elem_data,...
        elem_prop )

%Load decomposition L'*L = P of VM criterion
L = [ 0 0 1 0;
      -sqrt(3)/3 sqrt(3)/6 0 sqrt(3)/6;
      0 1/2 0 -1/2];

%Load material properties
E = elem_prop.E;
v = elem_prop.v;
H = elem_prop.H;
sy = elem_prop.sy;

%Linear Plane Strain constitutive matrix
C = E/(1+v)/(1-2*v)*[1-v v 0 v;
                    v 1-v 0 v;
                    0 0 (1-2*v)/2 0;
                    v v 0 1-v];

%Calculate trial values
stress_npl_trial = elem_data.sigma + C*dlt_strain;
alpha_npl_trial = elem_data.alpha;

%Trial step
sVM_npl_trial = sqrt(3)*norm(L*stress_npl_trial);
if(sVM_npl_trial <= sy + H*alpha_npl_trial)
elem_data.sigma = stress_npl_trial;
elem_data.alpha = alpha_npl_trial;
Cep = C;
return;
end

%Compliance matrix
C_inv = 1/E*[ 1 -v 0 -v;
             -v 1 0 -v;
             0 0 2*(v+1) 0;
             -v -v 0 1];

%Cholesky compliance matrix
L_C_inv = chol(C_inv);
```

```
%Setup the optimization problem
% addpath('C:\Program Files\Mosek\8\toolbox\r2014a');
[~, res] = mosekopt('symbcon echo(0)');

%Number of cones
num_cones = 3; %two cones for restating the problem...
              %in the conic form plus the von Mises cone

%Variables dimensions
sigma_dim = 4;
s_dim = 3;
alpha_dim = 1;
a1_dim = 4;
a2_dim = 1;
y1_dim = 1;
y2_dim = 1;
y11_dim = 1;
y12_dim = 1;
y21_dim = 1;
y22_dim = 1;
kappa_dim = 1;

vars_count = 0;
%Assigning the variables ids
sigma_vars = (1:sigma_dim)';
vars_count = sigma_dim + vars_count;
s_vars = (1:s_dim)' + vars_count;
vars_count = s_dim + vars_count;
alpha_vars = (1:alpha_dim)' + vars_count;
vars_count = alpha_dim + vars_count;
a1_vars = (1:a1_dim)' + vars_count;
vars_count = a1_dim + vars_count;
a2_vars = (1:a2_dim)' + vars_count;
vars_count = a2_dim + vars_count;
y1_vars = (1:y1_dim)' + vars_count;
vars_count = y1_dim + vars_count;
y2_vars = (1:y2_dim)' + vars_count;
vars_count = y2_dim + vars_count;
y11_vars = (1:y11_dim)' + vars_count;
vars_count = y11_dim + vars_count;
y12_vars = (1:y12_dim)' + vars_count;
vars_count = y12_dim + vars_count;
y21_vars = (1:y21_dim)' + vars_count;
vars_count = y21_dim + vars_count;
y22_vars = (1:y22_dim)' + vars_count;
vars_count = y22_dim + vars_count;
kappa_vars = (1:kappa_dim)' + vars_count;
```

```

vars_count = kappa_dim + vars_count;

%Specify the cones' types
prob.cones.type(1:num_cones) = res.symbcon.MSK_CT_QUAD;

%Specify the cones' variables
prob.cones.sub = [y11_vars;y12_vars;a1_vars; %y1 cone
y21_vars;y22_vars;a2_vars; %y2 cone
kappa_vars;s_vars]; %kappa cone

%Assign the keys ids
key1_ptr = 1;
key2_ptr = y11_dim + y12_dim + a1_dim + key1_ptr;
key3_ptr = y21_dim + y22_dim + a2_dim + key2_ptr;

%Specify the cones' keys
prob.cones.subptr = [key1_ptr;key2_ptr;key3_ptr];

%Specify lower and upper bounds
prob.blx = -Inf*ones(vars_count,1);
prob.bux = +Inf*ones(vars_count,1);

prob.blx(kappa_vars) = zeros(kappa_dim);

%Specify objective function
prob.c = zeros(vars_count,1);
prob.c(y1_vars) = 1;
prob.c(y2_vars) = 1;

%Specify constraints
num_constraints = a1_dim + a2_dim + y11_dim + ...
                 y12_dim + y21_dim + y22_dim + ...
                 s_dim + kappa_dim;
prob.a = sparse(num_constraints,vars_count);
prob.blc = zeros(num_constraints,1);
prob.buc = zeros(num_constraints,1);

row_count = 0;
% First row
r1 = 1:a1_dim;
row_count = a1_dim + row_count;
prob.a(r1,a1_vars) = eye(a1_dim);
prob.a(r1,sigma_vars) = 1/sqrt(2)*L_C_inv;
prob.blc(r1) = 1/sqrt(2)*L_C_inv*stress_np1_trial;
prob.buc(r1) = 1/sqrt(2)*L_C_inv*stress_np1_trial;

% Second row

```

```

r2 = (1:a2_dim) + row_count;
row_count = a2_dim + row_count;
prob.a(r2,a2_vars) = eye(a2_dim);
prob.a(r2,alpha_vars) = 1/sqrt(2)*sqrt(H)*eye(alpha_dim);
prob.blc(r2) = 1/sqrt(2)*sqrt(H)*alpha_np1_trial;
prob.buc(r2) = 1/sqrt(2)*sqrt(H)*alpha_np1_trial;

% Third row
r3 = (1:y11_dim) + row_count;
row_count = y11_dim + row_count;
prob.a(r3,y11_vars) = eye(y11_dim);
prob.a(r3,y1_vars) = -1/2*eye(y1_dim);
prob.blc(r3) = 0.5;
prob.buc(r3) = 0.5;

% Fourth row
r4 = (1:y12_dim) + row_count;
row_count = y12_dim + row_count;
prob.a(r4,y12_vars) = eye(y12_dim);
prob.a(r4,y1_vars) = -1/2*eye(y1_dim);
prob.blc(r4) = -0.5;
prob.buc(r4) = -0.5;

% Fifth row
r5 = (1:y21_dim) + row_count;
row_count = y21_dim + row_count;
prob.a(r5,y21_vars) = eye(y21_dim);
prob.a(r5,y2_vars) = -1/2*eye(y2_dim);
prob.blc(r5) = 0.5;
prob.buc(r5) = 0.5;

% Sixth row
r6 = (1:y22_dim) + row_count;
row_count = y22_dim + row_count;
prob.a(r6,y22_vars) = eye(y22_dim);
prob.a(r6,y2_vars) = -1/2*eye(y2_dim);
prob.blc(r6) = -0.5;
prob.buc(r6) = -0.5;

% Seventh row
r7 = (1:s_dim) + row_count;
row_count = s_dim + row_count;
prob.a(r7,s_vars) = eye(s_dim);
prob.a(r7,sigma_vars) = -L;
prob.blc(r7) = 0;
prob.buc(r7) = 0;

```

```

% Eighth row
r8 = (1:kappa_dim) + row_count;
row_count = kappa_dim + row_count;
prob.a(r8,kappa_vars) = eye(kappa_dim);
prob.a(r8,alpha_vars) = -1/sqrt(3)*H*eye(alpha_dim);
prob.blc(r8) = 1/sqrt(3)*sy;
prob.buc(r8) = 1/sqrt(3)*sy;

% Set accuracy parameters

% Primal feasibility tolerance for the primal solution
param.MSK_DPAR_INTPNT_CO_TOL_PFEAS = 1.0e-16;

% Solve SOCP problem
[~,res] = mosekopt('minimize echo(0)',prob,param);

elem_data.sigma = res.sol.itr.xx(sigma_vars);
elem_data.alpha = res.sol.itr.xx(alpha_vars);
elem_data.ep = elem_data.ep + dlt_strain ...
    + C_inv*(stress_npl_trial - elem_data.sigma);

% -----Consistent Tangent Elasto-Plastic Operator-----%

sub = prob.cones.sub;
subptr = prob.cones.subptr;

x = res.sol.itr.xx;
X = zeros(length(x));

for i = 1:length(subptr)-1
    cone_vars = sub(subptr(i):subptr(i+1)-1);
    for j = 1:length(cone_vars)
        X(cone_vars(j),cone_vars(j)) = x(cone_vars(1));
    end
    for j = 2:length(cone_vars)
        X(cone_vars(j),cone_vars(1)) = x(cone_vars(j));
        X(cone_vars(1),cone_vars(j)) = x(cone_vars(j));
    end
end

cone_vars = sub(subptr(end):end);
for j = 1:length(cone_vars)
    X(cone_vars(j),cone_vars(j)) = x(cone_vars(1));
end
for j = 2:length(cone_vars)
    X(cone_vars(j),cone_vars(1)) = x(cone_vars(j));
    X(cone_vars(1),cone_vars(j)) = x(cone_vars(j));
end

```

```

end

R_vars = setdiff(1:length(x), sub);
for i = 1:length(R_vars)
X(R_vars(i), R_vars(i)) = x(R_vars(i));
end

s = res.sol.itr.snx;
S = zeros(length(s));

for i = 1:length(subptr)-1
cone_vars = sub(subptr(i):subptr(i+1)-1);
for j = 1:length(cone_vars)
S(cone_vars(j), cone_vars(j)) = s(cone_vars(1));
end
for j = 2:length(cone_vars)
S(cone_vars(j), cone_vars(1)) = s(cone_vars(j));
S(cone_vars(1), cone_vars(j)) = s(cone_vars(j));
end
end

cone_vars = sub(subptr(end):end);
for j = 1:length(cone_vars)
S(cone_vars(j), cone_vars(j)) = s(cone_vars(1));
end
for j = 2:length(cone_vars)
S(cone_vars(j), cone_vars(1)) = s(cone_vars(j));
S(cone_vars(1), cone_vars(j)) = s(cone_vars(j));
end

R_vars = setdiff(1:length(x), sub);
for i = 1:length(R_vars)
S(R_vars(i), R_vars(i)) = s(R_vars(i));
end

A = prob.a;

H = [zeros(vars_count) A'          eye(vars_count);
A      zeros(row_count)          zeros(row_count, vars_count);
S      zeros(vars_count, row_count) X];

dcdx = zeros(vars_count, 4);
dbdx = zeros(row_count, 4);
dbdx(r1, :) = 1/sqrt(2)*L_C_inv*C;

q = [dcdx;dbdx;zeros(vars_count, 4)];

```

```

dpda = H\q;

Cep = dpda(1:4,1:4);

end
% -----Consistent Tangent Elasto-Plastic Operator-----%

```

Listing 7.2: State-update procedure for the von Mises model using conic optimization.

The load-displacement curve may be plotted by accessing the appropriate fields of the `output` data structure. For instance, the following piece of code:

```

num_step = length(params.step);
plot_pts = zeros(num_step+1,2);
for step = 1:num_step
    plot_output = output{step};
    P = plot_output.F(1,1) + plot_output.F(12,1);
    u = plot_output.u(1,1);
    plot_pts(step+1,:) = [u,P];
end

figure
plot(plot_pts(:,1),plot_pts(:,2),'-sb');

```

leads to the load-displacement related to the displacement of the inner layer of the perforated plate against the equivalent force due to the inner pressure, as depicted in Figure 7.2.

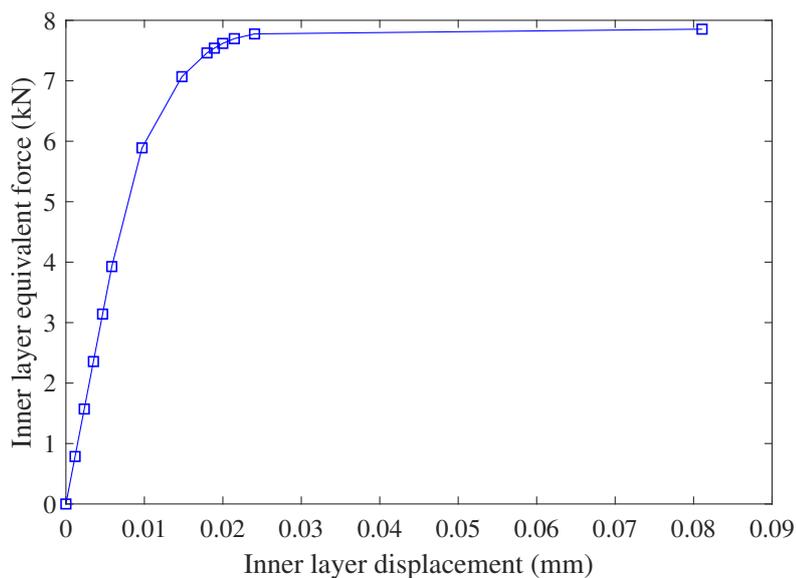


Figure 7.2: Perforated circular plate example load-displacement curve.