STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **15**, February 2026, pp 1181–1194.
Published online in International Academic Press (www.IAPress.org)

Optimized Dual Algorithm with Automatic Penalty Parameter Selection for Elastoplastic Contact Analysis

Youssef BOUZID [1,*], EL Hassan BENKHIRA [2], Rachid FAKHAR [1], Youssef MANDYLY [3]

[1]*Laboratory LS3M, University Sultan Moulay Slimane, 25000 Khouribga, Morocco*
[2]*Laboratory MACS, University Moulay Ismaïl, Faculty of Sciences, ESTM, BP 3103, Toulal-Meknès, Morocco*
[3]*Laboratory AICSE, ENSAM of Casablanca, University Hassan II, of Casablanca, Casablanca 20000, Morocco*

Abstract In this article, We examine a mechanical contact problem involving an elastoplastic body and a rigid foundation. The behavior of the material is characterized by Hencky's nonlinear elastic constitutive law. We present an iterative method based on Kacanov's method, with an augmented Lagrangian formulation at each iteration. To improve the algorithm in the discrete case, we propose an alternative approach consisting of automatic and optimal selection of the penalty parameter, accompanied by an approximate algorithm. To this end, we eliminate two unknowns, the principal and the auxiliary, and thus formulate a purely dual algorithm, enabling us to study the convergence of our method in depth. Finally, numerical experiments on two-dimensional problems are conducted to demonstrate the performance of the proposed method.

Keywords Elasto-plasticity; Signorini conditions; Hencky's constitutive law; Kacanov's method; Augmented Lagrangian; Uzawa block.

AMS 2020 subject classifications 35J87, 74C05, 49J40, 47J25, 74S05, 65N55, 37M05.

DOI: 10.19139/soic-2310-5070-2580

1. Introduction

The study of interactions between elastoplastic materials and rigid foundations involves analyzing stress distributions and deformation behaviors under different loading conditions [1, 2, 3, 4, 5]. This research examines the mechanical responses of elastoplastic materials when in contact with rigid foundations, focusing on how these materials deform. Understanding these interactions is crucial to the design of durable structures and the prediction of material performance in engineering applications.
However, Problems related to contact mechanics are usually too complex to be solved analytically in dimensions higher than one. As a result, numerical methods are essential. Among these methods, the Finite Element Method (FEM) is particularly popular because of its strong connection to solid mechanics. FEM works by breaking down the body's geometry into smaller elements of fixed size. To handle contact and friction conditions, the Augmented Lagrangian method is commonly used [5, 6, 7, 8, 9].
This paper explores the challenges and advanced methods for solving these complex contact mechanics problems. Specifically, We tackle the nonlinear Hencky-type contact problem, considering Tresca's friction law and a rigid foundation,The existence and uniqueness of the solution have been established. We use the Alternating Direction Method of Multipliers (ADMM) and improve convergence by reducing variables to develop a fully dual algorithm, while also fine-tuning the choice of the penalty parameter.

*Correspondence to: Youssef BOUZID (Email: youssef.bouzid@usms.ac.ma). University Sultan Moulay Slimane, Laboratory LS2ME, 25000 Khouribga, Morocco.

ISSN 2310-5070 (online) ISSN 2311-004X (print)

2. The mechanical problem and Variational formulation

In this section, we present the model for a unilateral contact problem involving Tresca's friction law between an elastoplastic material and a rigid foundation.

2.1. The mechanical problem

Let us consider an elasto-plastic body occupying, in its reference configuration, a domain $D \subset \mathbb{R}^d$, $d = 2, 3$ that is bounded and open with a fairly regular boundary $\Sigma = \Sigma_d \cup \Sigma_n \cup \Sigma_c$, such that the three parts Σ_d, Σ_n and Σ_c are disjoint and $mes(\Sigma_c) > 0$. We assume that a volume force f_i acts in D. The body is supposed to be fixed on Σ_d, in part Σ_n by applying surface traction f_n. The body comes into contact with a rigid foundation in the part Σ_c.
Throughout this article, we denote: $\omega : D \to \mathbb{R}^d$ for the displacement field and $\tau : D \to \mathbb{S}^d$, $\tau = (\tau_{ij})$ for the stress tensor. Furthermore, we consider the hypothesis of small constraints so that the tensor of constraints is represented by $\varrho(\omega) = (\varrho_{ij}(\omega))$, the strain tensor given by $\varrho_{ij}(\omega) = \frac{1}{2}(\omega_{i,j} + \omega_{j,i})$, and "Div" denotes the divergence operator for tensors, *i.e.*, $\mathrm{Div}\tau = (\tau_{ij,j})$.
We denote the normal and tangential components of the displacement vector and the stress tensor by:

$$\omega_\nu = \omega \cdot \nu, \quad \omega_\tau = \omega - \omega_\nu \nu, \quad \tau_\nu = \tau \cdot \nu \cdot \nu, \quad \tau_\tau = \tau \cdot \nu - \tau_\nu \nu,$$

where ν is the unit outward normal vector on Σ. Assuming the system is static, the equilibrium equation is as follows:

$$\mathrm{Div}\ \tau + f_i \quad \text{in } D, \tag{1}$$

$$\omega = 0 \quad \text{on } \Sigma_d, \tag{2}$$

$$\tau\nu = f_n \quad \text{on } \Sigma_n, \tag{3}$$

$$(\omega_\nu - \rho) \le 0,\ \tau_\nu \le 0,\ \tau_\nu(\omega_\nu - \rho) = 0 \quad \text{on } \Sigma_c, \tag{4}$$

$$\left.\begin{array}{l} \|\tau_\tau\| \le \psi, \\ \|\tau_\tau\| < \psi \Rightarrow \omega_\tau = 0, \\ \|\tau_\tau\| = \psi \Rightarrow \exists \lambda \in \mathbb{R}^+ \text{ such that } \tau_\tau = -\lambda \omega_\tau, \end{array}\right\} \quad \text{on } \Sigma_c, \tag{5}$$

To describe the elasto-plastic material, we assume that the stress tensor and the strain tensor comply with Hencky's law, which can then be written as follows:

$$\tau = k_0\, \mathrm{tr}\varrho(\omega)\, \mathrm{I} + 2g(\|\overline{\varrho}(\omega)\|^2)\overline{\varrho}(\omega).$$

In equations (1)-(5), we have consolidated all the equilibrium, boundary, and contact conditions for the problem. The Signorini contact conditions (4) and Tresca's law of friction (5) are applied.

2.2. Variational formulation

To obtain a variational formulation, we consider the following real Hilbert functional spaces: $\mathcal{Q}(D) = L^2(D)^d$, $H_1 = H^1(D)^d$, $\mathcal{H}^1(D) = \{\tau = (\tau_{ij})\,;\, \tau_{ij} = \tau_{ji} \in L^2(D)\}$, $\mathcal{H}_1 = \{\tau \in \mathcal{H}^1(D)\,;\, \tau_{ij,j} \in \mathcal{Q}(D)\}$.
These spaces are equipped with the following inner products:

$$(\omega, \xi)_{\mathcal{Q}(D)} = \int_D \omega_i \xi_i\, dx, \quad (\tau, \mathscr{T})_{\mathcal{H}^1(D)} = \int_D \tau_{ij}\mathscr{T}_{ij}\, dx,$$

$$(\omega, \xi)_{H_1} = (\omega, \xi)_{\mathcal{Q}(D)} + (\nabla\omega, \nabla\xi)_{\mathcal{Q}(D)}, \quad (\tau, \mathscr{T})_{\mathcal{H}_1} = (\tau, \mathscr{T})_{\mathcal{H}^1(D)} + (\mathrm{div}\,\tau, \mathrm{div}\,\mathscr{T})_{\mathcal{Q}(D)}.$$

The corresponding norms are defined as:

$$\|\omega\|^2_{\mathcal{Q}(D)} = (\omega, \omega)_{\mathcal{Q}(D)}, \quad \|\tau\|^2_{\mathcal{H}^1(D)} = (\tau, \tau)_{\mathcal{H}^1(D)}, \quad \|\omega\|^2_{H_1} = (\omega, \omega)_{H_1}, \quad \|\tau\|^2_{\mathcal{H}_1} = (\tau, \tau)_{\mathcal{H}_1}.$$

Considering the boundary condition (2), we define the closed subspace of H_1 as follows:

$$\mathscr{V} = \{\, \xi \in H_1 \,;\; \xi = 0 \quad \text{on } \Sigma_d \},$$

Let Λ denote the set of admissible displacements:

$$\Lambda = \{\, \xi \in \mathscr{V} \,;\; (\xi_\nu - \rho) \leq 0 \quad \text{on } \Sigma_c \}.$$

Next, we introduce the functional $\ell : \mathscr{V} \to \mathbb{R}$ by:

$$\ell(\xi) = \int_{\Sigma_c} \psi \, \|\xi_\tau\| \, da.$$

Applying Riesz's representation theorem, we define $\mathcal{F} \in \mathscr{V}$ as

$$(\mathcal{F}, \xi)_{\mathscr{V}} = \int_D f_i \cdot \xi \, dx + \int_{\Sigma_n} f_n \cdot \xi \, da \quad \forall \xi \in \mathscr{V}.$$

These formulations establish the foundation for constructing the variational problem under study.

We thus arrive at the corresponding weak formulation.

Problem (PV). *Find a displacement field $\omega \in \Lambda$ such that :*

$$(\mathcal{A}\omega, \xi - u)_{\mathscr{V}} + \ell(\xi) - \ell(\omega) \geq (\mathcal{F}, \xi - \omega)_{\mathscr{V}} \quad \forall \xi \in \Lambda,$$

where the operator $\mathcal{A} : \mathscr{V} \to \mathscr{V}$ is defined by:

$$\begin{aligned} &(\mathcal{A}\omega, \xi)_{\mathscr{V}} \\ &= \int_D k_0 \operatorname{tr}(\rho(\omega)) \operatorname{tr}(\varrho(\xi)) + 2g(\|\overline{\varrho}(\omega)\|^2)(\overline{\varrho}(\omega) : \varrho(\xi)) dx. \end{aligned} \tag{6}$$

We are therefore in a position to state the existence and uniqueness result below:

Theorem 1
The problem **(PV)** has unique solution $\omega \in \Lambda$.

The proof of this theorem is given in [5].

3. Iteration method

3.1. Kacanov's iterative method

In this section, an iterative scheme based on Kacanov's method is proposed to solve the quasi-variational problem **(PV)**, addressing nonlinearities that arise from friction and material behavior. This method is considered as a sequence of linear contact problems with Tresca's law of friction.
This method consists of the following procedure. Let ω_n be the n-th approximation of the solution to **Problem (PV)**. We aim to find the weak solution ω_{n+1} of the following linear problem:

$$\begin{cases} \text{Given an initial guess } \omega_0, \text{ find } \omega_{n+1} \in \Lambda \text{ such that:} \\ \mathbb{B}\left(\omega_n; \omega_{n+1}, \xi - \omega_{n+1}\right) + j\left(\xi\right) - j\left(\omega_{n+1}\right) \\ \geq \left(\mathcal{F}, \xi - \omega_{n+1}\right)_{\mathscr{V}} \quad \forall \xi \in \Lambda, \end{cases} \tag{7}$$

where the operator $\mathbb{B} : \Lambda \times \mathscr{V} \times \mathscr{V} \longrightarrow \mathbb{R}$ is defined by:

$$\mathbb{B}(\omega; \xi, w) = \left(k_0 \operatorname{tr}(\varrho(\xi)) I + 2g\left(\|\overline{\varrho}(\omega)\|^2\right) \overline{\varrho}(\xi), \varrho(w)\right)_{\mathcal{H}}.$$

Remark 1
Given a fixed $\omega \in \Lambda$, $(\xi, w) \mapsto \mathbb{B}(\omega; \xi, w)$ is a bilinear, symmetric.

Theorem 2
Under assumption of Theorem 1, The iterative method in (7) is convergent:

$$\|\omega_n - \omega\|_V \to 0 \ \ as \ \ n \to +\infty.$$

The proof of this theorem is given in [5].

3.2. Augmented Lagrangian for the iterative problem

We formulate a constrained minimization problem corresponding to the equation (7), for which a numerical analysis has been performed. The proposed minimization problem can be expressed as follows:

$$\begin{cases} \text{Find } \omega \in \Lambda \text{ such that:} \\ Q_n(\omega) + \ell(\omega) \leq Q_n(\xi) + \ell(\xi) \quad \forall \xi \in \Lambda, \end{cases} \tag{8}$$

and Q_n is the energy functional resulting from non-frictional effects, expressed by:

$$Q_n(\xi) = \frac{1}{2}\mathbb{B}\left(\omega_n; \xi, \xi\right) - (\mathcal{F}, \xi)_{\mathscr{V}} \quad \forall \xi \in \mathscr{V}.$$

We define an augmented Lagrangian corresponding to the minimization problem (8). The augmented Lagrangian function $\mathscr{L}_r$ is given on $\mathscr{V} \times L^2\left(\Sigma_c\right)^2 \times L^2\left(\Sigma_c\right)^2$ by:

$$\begin{aligned} &\mathscr{L}_r(\xi, \phi; \theta) \\ &= Q_n(\xi) + j\left(\phi_f\right) + I_C\left(\phi_c\right) \\ &\quad + \left(\theta_c, \xi_\nu - \phi_c\right)_{L^2(\Sigma_c)} + \left(\theta_f, \xi_\tau - \phi_f\right)_{L^2(\Sigma_c)} \\ &\quad + \frac{r}{2}\left\|\xi_\nu - \phi_c\right\|^2_{L^2(\Sigma_c)} + \frac{r}{2}\left\|\xi_\tau - \phi_f\right\|^2_{L^2(\Sigma_c)}, \end{aligned} \tag{9}$$

where r is a constant penalty parameter with $r > 0$, and θ is defined as (θ_c, θ_f).

4. Discrete formulation

We consider a polyhedral domain, $D \subset \mathbb{R}^d$, which is therefore exactly triangularizable. We are considering a finite element triangulation $\mathcal{T}_h$ of D, which is piecewise linear and continuous, and compatible with the decomposition of its boundary Σ into the parts Σ_d, Σ_n, and Σ_c. Let n be the number of nodes in the triangulation. The dimension of the finite element subspace $\mathscr{V}_h \subset \mathscr{V}$ is given by $\dim \mathscr{V}_h = dn$. We use the notations introduced in [15, 19]. for the algebraic formulation.Finite element discretization produces several matrices and vectors that are crucial for formulating the problem. First of all, M_n and M_t are standard mass matrices of size $m \times m$, both symmetric and positive definite. Next, The vector $\mathcal{F} \in \mathbb{R}^{dn}$ represents the discrete external forces. $\rho \in \mathbb{R}^m$ is used to represent the discrete normalized gap. ϕ_n et ϕ_f sont des inconnues auxiliaires discrètes appartenant à $\mathbb{R}^m$.Enfin, λ_n et λ_f sont des multiplicateurs de Lagrange discrets appartenant à $\mathbb{R}^m$. . With the previous notations, the augmented Lagrangian (9) in the friction case is now formulated as follows:

$$\mathscr{L}_r(\omega, \phi; (\lambda_n, \lambda_f)) = Q_n(\omega) + \lambda_n^\top M_n(N\omega - \phi_n) + \lambda_f^\top M_\tau(T\omega - \phi_f) + \frac{r}{2}(N\omega - \phi_n)^\top M_n(N\omega - \phi_n) + \frac{r}{2}(T\omega - \phi_f)^\top M_\tau(T\omega - \phi_f),$$

where $Q_n(\omega) = \frac{1}{2}\omega^T B_n \omega - \mathcal{F}^T\omega$.

4.1. ADMM algorithm

We then present Algorithm 1 to handle the discrete case of friction contact.

Algorithm 1: Discrete ADMM Algorithm

Initialization: Set $k = 0$, and choose initial values ϕ^0 and λ^0.

For each iteration $k \geq 0$, compute the following steps:

Step 1: Update the primary variable ω_n^{k+1}
Solve for $\omega_n^{k+1} \in \mathbb{R}^{dn}$:

$$(B_n + rN^\top M_n N + rT^\top M_t T)\omega_n^{k+1} = \mathcal{F} + N^\top M_n\big(r\phi_n^k - \lambda_n^k\big) + T^\top M_t\big(r\phi_f^k - \lambda_f^k\big). \tag{10}$$

Step 2: Compute the auxiliary variable ϕ_n^{k+1}

$$\phi_n^{k+1} = \omega_n^{k+1} + \frac{1}{r}\left[\lambda^k - \left(\lambda^k + r\left(\omega_n^{k+1} - \rho\right)^+\right)\right], \tag{11}$$

$$\phi_f^{k+1} = \begin{cases} \dfrac{|\lambda_f^k + r\omega_\tau^{k+1}| - \psi}{r\,|\lambda_f^k + r\omega_\tau^{k+1}|}\left(\lambda_f^k + r\omega_\tau^{k+1}\right), & \text{si } |\lambda_f^k + r\omega_\tau^{k+1}| > \psi \\ 0, & \text{si } |\lambda_f^k + r\omega_\tau^{k+1}| \leq \psi \end{cases} \tag{12}$$

Step 3: Update the Lagrange multiplier λ^{k+1}

$$\lambda_n^{k+1} = \lambda_n^k + r\big(N\omega_n^{k+1} - \phi_n^{k+1}\big). \tag{13}$$

$$\lambda_f^{k+1} = \lambda_f^k + r\big(T\omega_n^{k+1} - \phi_f^{k+1}\big). \tag{14}$$

Now that we have the results of the previous subsections, we can introduce the algorithm 1, our Uzawa block relaxation technique. We'll continue the process until the relative error in ω_k, ϕ_n, and ϕ_f is sufficiently small. In particular, the cycle will continue until:

$$\frac{\|\omega_n^{k+1} - \omega_n^k\|^2_{L^2(\Omega)} + \|\phi_c^{k+1} - \phi_c^k\|^2_{L^2(\Gamma_3)} + \|\phi_f^{k+1} - \phi_f^k\|^2_{L^2(\Gamma_3)}}{\|\omega_n^{k+1}\|^2_{L^2(\Omega)} + \|\phi_c^{k+1}\|^2_{L^2(\Gamma_3)} + \|\phi_f^{k+1}\|^2_{L^2(\Gamma_3)}} < \epsilon^2, \tag{15}$$

where ϵ is a predefined tolerance.

remark. Next, we consider the frictionless case, which corresponds to working with a modified version of the above algorithm. In this setting, the first equation is updated as shown earlier, while equations (12) and (14) are removed. The goal is to enhance the handling of the normal contact, leading to an overall improvement of the algorithm.

$$(B_n + rN^\top M_n N)\omega_n^{k+1} = \mathcal{F} + N^\top M_n\big(r\phi_n^k - \lambda_n^k\big). \tag{16}$$

remark. The conditioning of the matrix $(B_n + rN^\top MN)$ is asymptotically proportional to r. Thus, as this parameter increases, the more difficult it becomes to solve the system (16).
The study of the influence of the penalization parameter r leads us to propose an elimination of primal and auxiliary variables, as we will explore in the following section.

To simplify the notation, we omit the subscript n indicating the normal contact; the tangential component has already been removed since we restrict ourselves to the frictionless contact case.

4.1.1. Pure dual version The pure dual algorithm can be obtained by eliminating the auxiliary ϕ and displacement vectors ω. Let $x = (x_1, x_2, ..., x_n)$ be an vector in $\mathbb{R}^n$, the positive part of x, denoted $max(x, 0)$, is a vector in

defined by its components:

$$x_i^+ = \begin{cases} x_i & \text{si } x_i > 0 \\ 0 & \text{si } x_i \leq 0 \end{cases},$$

we also note that $min(x,0) = x - max(x,0)$.
Furthermore, by substituting (11) into (13), we obtain:

$$\lambda^{k+1} = max(\lambda^k + r(N\omega^{k+1} - \rho), 0). \tag{17}$$

Using the formula (13) for the $k-1$ iteration, we obtain:

$$\lambda^k = \lambda^{k-1} + r(N\omega^k - \phi^k), \tag{18}$$

which gives

$$r\phi^k - \lambda^k = \lambda^{k-1} - 2\lambda^k + rN\omega^k \tag{19}$$

$$= \lambda^{k-1} + r(N\omega^k - \rho) + r\rho - 2\lambda^k. \tag{20}$$

To keep the calculations simple, we consider the following two sequences

$$\lambda_+^{k+1} = max(\lambda^k + r(N\omega^{k+1} - \rho), 0). \quad \lambda_-^{k+1} = min(\lambda^k + r(N\omega^{k+1} - \rho), 0). \tag{21}$$

we can then find the formula below which allows us to eliminate the auxiliary variable:

$$r\phi^k - \lambda^k = \lambda_-^k - \lambda_+^k + r\rho. \tag{22}$$

The pure dual algorithm can be obtained by eliminating the displacement vector ω. To do this, we define the matrix $B_{n,r}$ by :

$$B_{n,r} = (B_n + rN^\top MN), \tag{23}$$

this matrix is invertible by construction, so according to (16), we have:

$$\omega^{k+1} = B_{n,r}^{-1}(\mathcal{F} + rN^\top M\rho + N^\top M(\lambda_-^k - \lambda_+^k)). \tag{24}$$

We substitute the relation (24) into (22) which gives **Algorithm** 2.

Algorithm 2: Pure Dual ADMM

Initialization: Set $k = 0$, and initial values λ_+^0 and λ_-^0.

Set parameters: $B_{n,r}$, $a = B_{n,r}^{-1}(\mathcal{F} + rN^\top M\rho)$, $B_1 = \mathbb{I} - rNB_{n,r}^{-1}N^\top M$ and $B_2 = rNB_{n,r}^{-1}N^\top M$.

For each iteration $k > 0$, compute successively λ_+^{k+1} and λ_-^{k+1} as follows:

Step 1: Update λ_+^{k+1}

$$\lambda_+^{k+1} = \max\left(B_1\lambda_+^k + B_2\lambda_-^k + r(Na - \rho), 0\right). \tag{25}$$

Step 2: Update λ_-^{k+1}

$$\lambda_-^{k+1} = \min\left(B_1\lambda_+^k + B_2\lambda_-^k + r(Na - \rho), 0\right). \tag{26}$$

---

remark. Algorithm 2 is also of great interest for analyzing the influence of the parameter r on the convergence. It provides a dual-based approach that helps to better understand the algorithm's behavior, but it remains difficult to implement in practice.

4.2. Optimizing Penalty Parameter

It is clear that for all λ in $\mathbb{R}^n$ $\|\lambda_+\| \leq \|\lambda\|$ and $\|\lambda_-\| \leq \|\lambda\|$ according to relations (25) and (26), we obtain:

$$\|\lambda_{\pm}^{k+1}\| \leq \|\mathbb{B}\lambda_{\pm}^{k}\| + r\|(Na - \rho)\|_2, \tag{27}$$

such as :

$$\mathbb{B} = \begin{bmatrix} B_1 & B_2 \\ -B_1 & -B_2 \end{bmatrix}, \quad \lambda_{\pm}^{k} = \begin{bmatrix} \lambda_{+}^{k} \\ \lambda_{-}^{k} \end{bmatrix},$$

$$\|\lambda_{\pm}^{k}\| = max(\|\lambda_{-}^{k}\|_2, \|\lambda_{+}^{k}\|_2).$$

Hereafter, the convergence of the algorithm is dependent on the spectral radius of the matrix $\mathbb{B}$. which makes it possible to study the non-zero eigenvalues of this matrix.

remark. The properties of the sequence $\{\lambda_{\pm}^{k}\}_{k\geq 0}$, as specified in the theorem, directly follow from the convergence properties of the sequences $\{\omega_n^k\}_{k\geq 0}$ and $\{\phi_n^k\}_{k\geq 0}$, with the sequence $\{\lambda_{\pm}^{k}\}_{k\geq 0}$ depending linearly on these sequences.

So we have this proposition,

Proposition 4.1. *the non-zero eigenvalues of matrix $\mathbb{B}$ are those of $B_1 - B_2$ and conversely.*

.

Proof
Let $\boldsymbol{x} = [\boldsymbol{x}_1 \;\; \boldsymbol{x}_2]^\top$ be the eigenvector associated with a non-zero eigenvalue λ of the matrix $B_1 - B_2$
We have

$$\mathbb{B}\boldsymbol{x} = \lambda\boldsymbol{x},$$

$$\begin{cases} B_1\boldsymbol{x}_1 + B_2\boldsymbol{x}_2 = \lambda\boldsymbol{x}_1 \\ -B_1\boldsymbol{x}_1 - B_2\boldsymbol{x}_2 = \lambda\boldsymbol{x}_2 \end{cases} \Rightarrow \lambda\boldsymbol{x}_1 = -\lambda\boldsymbol{x}_2,$$

since λ is non-zero, so $\boldsymbol{x}_1 = -\boldsymbol{x}_2$ therefore,

$$(B_1 - B_2)\boldsymbol{x}_i = \lambda\boldsymbol{x}_i \quad \forall i = 1, 2.$$

Reciprocally, let λ be an eigenvalue associated with the eigenvector $\boldsymbol{x}$ of the matrix $B_1 - B_2$. It can be easily shown, through an algebraic manipulation, that these eigenvalues are also those of the matrix $\mathbb{B}$.

□

To study the eigenvectors of $\mathbb{B}$, we follow same of Glowinski's steps, for more details see the following reference [9].
We consider the sequence defined by:

$$\boldsymbol{\chi}_{k+1} = (B_1 - B_2)\boldsymbol{\chi}_k,$$

we have:

$$\boldsymbol{\chi}_{k+1} = \boldsymbol{\chi}_k - 2rNB_{n,r}^{-1}N^\top M\boldsymbol{\chi}_k, \tag{28}$$

Or :

$$B_{n,r}^{-1} = (I + rB_n^{-1}N^\top MN)^{-1}B_n^{-1}$$

we multiply (28) by $B_n^{-1}N^\top M$ what makes :

$$B_n^{-1}N^\top M\boldsymbol{\chi}_{k+1} = B_n^{-1}N^\top M\boldsymbol{\chi}_k - 2rB_n^{-1}N^\top MN(I + rB_n^{-1}N^\top MN)^{-1})\boldsymbol{\chi}_k,$$

we consider the following sequence

$$\boldsymbol{\xi}_k = B_n^{-1} N^\top M \boldsymbol{\chi}_k, \tag{29}$$

such a recursive relation on the sequence $\{\boldsymbol{\xi}_k\}_{k\geq 0}$ is important because it provides information on the convergence of the algorithm 2.
We can write (28) this way

$$\boldsymbol{\xi}_{k+1} = \mathcal{D}\boldsymbol{\xi}_k, \quad such \ \ that \ \ \mathcal{D} = I - 2rB_n^{-1} N^\top MN(I + rB_n^{-1} N^\top MN)^{-1}, \tag{30}$$

we have $\mathcal{D}$ is polynomial function of $\mathbb{B}_n^{-1} N^\top MN$, *i.e.*,

$$\mathcal{D} = p(B_n^{-1} N^\top MN) \quad where \quad p(t) = 1 - \frac{2rt}{1+rt}, \tag{31}$$

for (30),see, *e.g.*, [20] page 537 Example 7.3.8 for all λ_i eigenvalue associated with the eigenvector $\boldsymbol{x}_i$ of the matrix $B_n^{-1} N^\top MN$,then $\{p(\lambda_i); \boldsymbol{x}_i\}$are the eigenvalues and eigenvectors of the matrix $\mathcal{D}$ and conversely.
So $\dfrac{1-r\lambda_i}{1+r\lambda_i}$ such that λ_i are non-zero eigenvalues of $B_n^{-1} N^\top MN$ are non-zero eigenvalues of the matrix $\mathcal{D}$.

Proposition 4.2. *The eigenvalues of $B_n^{-1} N^\top MN$ are positive and the eigenvectors of two different eigenvalues are B_n-orthogonal, i.e. if $B_n^{-1} N^\top MN\boldsymbol{x}_i = \lambda_i \boldsymbol{x}_i$, $B_n^{-1} N^\top MN\boldsymbol{x}_j = \lambda_j \boldsymbol{x}_j$ with $\lambda_i \neq \lambda_j$, then $\boldsymbol{x}_j^\top B_n \boldsymbol{x}_i = 0$.*

Proof
Given that B_n^{-1} and M are positive definite and N is full rank, it follows that $B_n^{-1} N^\top MN$ is also positive definite. Let's get on with the rest, Let $\mathcal{A} = B_n^{-1} N'MN$. Assume that $\mathcal{A}$ has distinct eigenvalues λ_1 and λ_2 with corresponding eigenvectors $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, respectively.
By definition, we have:

$$\mathcal{A}\boldsymbol{x}_1 = \lambda_1 \boldsymbol{x}_1, \quad \mathcal{A}\boldsymbol{x}_2 = \lambda_2 \boldsymbol{x}_2.$$

To show B_n-orthogonality, we need to prove that:

$$\boldsymbol{x}_1^\top B_n \boldsymbol{x}_2 = 0.$$

Since $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are eigenvectors of $\mathcal{A}$ corresponding to distinct eigenvalues λ_1 and λ_2, we have:

$$\boldsymbol{x}_1^\top N^\top MN\boldsymbol{x}_2 = \lambda_2(\boldsymbol{x}_1^\top B_n \boldsymbol{x}_2) \quad \text{and} \quad \boldsymbol{x}_2^\top N^\top MN\boldsymbol{x}_1 = \lambda_1(\boldsymbol{x}_2^\top B_n \boldsymbol{x}_1),$$

since $N^\top MN$ is symmetric, which implies:

$$\boldsymbol{x}_1^\top N^\top MN\boldsymbol{x}_2 = \boldsymbol{x}_2^\top N^\top MN\boldsymbol{x}_1,$$

thus:

$$\lambda_2(\boldsymbol{x}_1^\top B_n \boldsymbol{x}_2) = \lambda_1(\boldsymbol{x}_2^\top B_n \boldsymbol{x}_1),$$

since $\lambda_1 \neq \lambda_2$, it follows that $\boldsymbol{x}_1^\top B_n \boldsymbol{x}_2 = 0$, proving that $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are B_n-orthogonal. □

It follows that,

Proposition 4.3. *If 0 is an eigenvalues of $B_n^{-1} N^\top MN$, then the corresponding eigen-subspace is $Ker(N)$. $Im(B_n^{-1} N^\top)$ and $Ker(N)$ are B_n-orthogonal and $Im(B_n^{-1} N^\top)$ is the eigen-subspace of non-zero eigenvalues of the matrix $B_n^{-1} N^\top MN$.*

From the relation(29), we have that the sequence $\{\boldsymbol{\xi}_n\}$ belongs to $Im(B_n^{-1} N^\top)$,
we can rewrite the sequence $\{\boldsymbol{\xi}_n\}$ in the eigenvector basis $\{\xi_i\}$ associated with the strictly positive eigenvalues $\{\lambda_i\}$ of the matrix $B_n^{-1} N^\top MN$, According to relation (31), this allows us to rewrite equation (30) in the following form:

$$\xi_{n+1}^i = \frac{1-r\lambda_i}{1+r\lambda_i}\xi_n^i \quad \forall\ i = 1, 2...., n_0 \ \ and \ \ n \geq 0,$$

from the above, we derive this convergence theorem

Theorem 3 (convergence)
Algorithm 2 converges for all values of $r > 0$.

Corollary 1 (Optimal step size and rate of convergence)
The optimal penalty parameter for r is determined by

$$r^* = \frac{1}{\sqrt{\lambda_{min}\lambda_{max}}}, \tag{32}$$

where λ_{min} and λ_{max} denote the smallest nonzero eigenvalue and the largest eigenvalue of $B_n^{-1}N^\top MN$, respectively.

Using the value for provided in (32), the convergence of Algorithm 1 is linear, with an asymptotic constant θ that satisfies:

$$\theta \leq \frac{1-\left(\widetilde{\lambda}\right)^{\frac{1}{2}}}{1+\left(\widetilde{\lambda}\right)^{\frac{1}{2}}}, \quad \text{with} \quad \widetilde{\lambda} = \frac{\lambda_{min}}{\lambda_{max}}. \tag{33}$$

Proof
The convergence parameter (32) is deduced from the behavior of the function $x \to \dfrac{1-x}{1+x}$. □

4.3. Penalty parameter approximation

We now focus on calculating λ_{min} and λ_{max}, the smallest nonzero eigenvalue and the largest eigenvalue of $B_n^{-1}N^\top MN$. To make this calculation easier, we arrange the indices so that the contact indices (C) and the interior indices (I) are grouped together in order. In this context, O^{II}, O^{IC}, and O^{Cl} are zero matrices, while M^{CC} is the diagonal mass matrix for the contact boundary.

$$B_n = \begin{pmatrix} B_n^{II} & B_n^{IC} \\ B_n^{Cl} & B_n^{CC} \end{pmatrix},$$

and

$$N^\top MN = \begin{pmatrix} O^{II} & O^{IC} \\ O^{CI} & M^{CC} \end{pmatrix},$$

where O^{II}, O^{IC} and O^{Cl} are matrices of zeros; and M^{CC} is the contact boundary (diagonal) mass matrix. Consequently, if we set

$$B_n^{-1} = \begin{pmatrix} \tilde{B}_n^{II} & \tilde{B}_n^{IC} \\ \tilde{B}_n^{CI} & \tilde{B}_n^{CC} \end{pmatrix},$$

it's easy to see that

$$B_n^{-1}N^\top MN = \begin{pmatrix} O^{II} & \tilde{B}_n^{IC}M^{CC} \\ O^{CI} & \tilde{B}_n^{CC}M^{CC} \end{pmatrix},$$

then the spectrum of $B_n^{-1}N^\top MN$ is the union of the spectrum of matrix $\tilde{B}_n^{CC}M^{CC}$ and the spectrum of matrix O^{II} (see [21] page 81 1.4p5).
To compute $\tilde{B}_n^{CC}$, we can apply block-Gaussian elimination to B_n with B^{II} as block-pivot

$$\tilde{B}_n^{CC} = \left(B^{CC} - B_n^{CI}(B_n^{II})^{-1}B_n^{IC}\right)^{-1},$$

it follows that $\tilde{B}_n^{CC}$ is the Schur complement of B_n^{II} in B_n. We now detail the practical steps for the approximation of the penalty parameter. Since computing $(B_n^{II})^{-1}$ is unpracticable even for medium size problem, we compute $\tilde{B}_n^{CC}$ by solving equivalent linear systems. We obtain Algorithm 3 .

Algorithm 3: Algorithm for computing the approximate penalty parameter r^*.

Input: B^{II}, B^{IC}, B^{CC}, B^{CI}, M^{CC}
Output: r^*
begin
Step 1. Solve for $\tilde{X}$ the system $B^{II}\tilde{X} = B^{IC}$;
Step 2. Compute the Schur complement of B^{II}: $S = B^{CC} - B^{CI}\tilde{X}$;
Step 3. Solve for X: $SX = M^{CC}$;
Step 4. Compute λ_{min} and λ_{max}, the extreme eigenvalues of X, and the optimal penalty parameter approximation (32);
end

5. Numerical experiments

We implemented the algorithms in MATLAB using piecewise linear finite elements and vectorized assembly functions. Our calculations were carried out on a Windows 11 system with a clock speed of 2.1 GHz and 16 GB RAM. The ADMM solver, following Algorithm 1, was used for our experiments. The test problems were chosen to illustrate the algorithm's behavior rather than to model real contact scenarios.

Example 1

The study investigates the deformation of a beam on a rigid foundation, considering specific loading and support conditions. It focuses on a beam of defined dimensions made from elastoplastic material, which is subjected to a surface force applied to one face.

Beam Dimensions We have chosen the academic example of a parallelepiped bar which has the following dimensions: the height $h = 1\,\mathrm{mm}$ and the length is equal to four times the height, with $\Sigma_d = \{0\} \times [0, 1]$, $\Sigma_n = [2, 4] \times \{1\}$ and $\Sigma_c = [0, 4] \times \{0\}$.

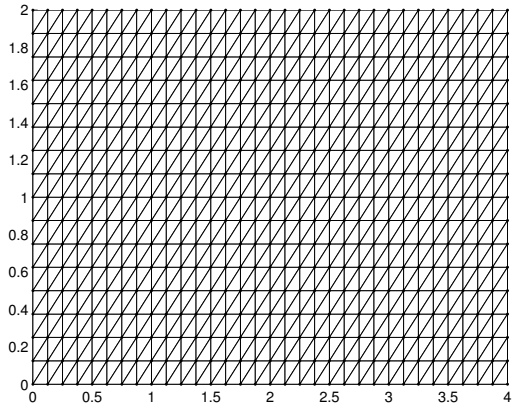


Figure 1. Undeformed configuration of 1/16 mesh size.

Loading and Support Conditions The body is subject to no volumetric force ($\mathcal{F} = 0$), a surface force of $[f_{n,x} = 0; f_{n,y} = -0.005]\,\mathrm{daN/mm}^2$ on Σ_n, and is fixed with zero displacement on Σ_d. It initially makes contact with a rigid foundation at Σ_c with no gap.

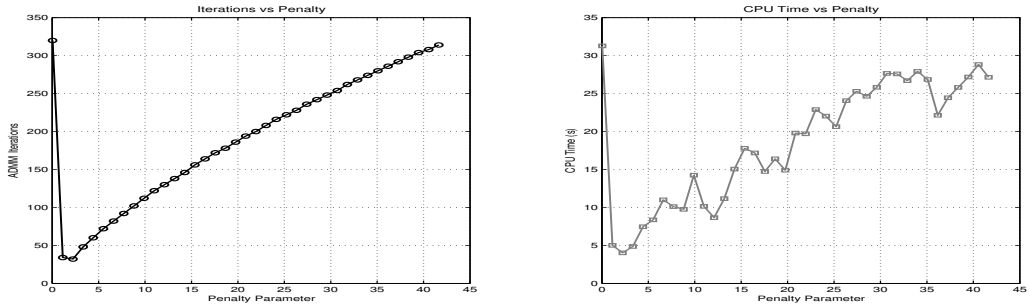
Material Properties Material properties include Young's modulus $E = 1\,\text{daN/mm}^2$, Poisson's ratio $\nu = 0.3$, and elastoplastic behavior with a defined yield strength. Suppose that the plasticity function g has the following form:

$$g(x) = \begin{cases} \mu & \text{if } x \leq x_0, \\ \mu \dfrac{x_0}{x} \left(\ln \left(\dfrac{x}{x_0} \right) + 1 \right) & \text{if } x \geq x_0. \end{cases} \tag{34}$$

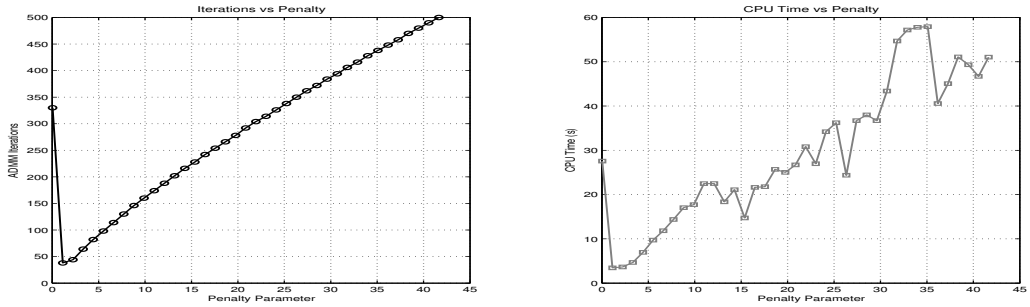
For the numerical simulations, we set $\epsilon = 10^{-5}$, as specified by the error term in equation 15.

remark. The matrix $B_n + rN^\top MN$ involved in the 23 algorithm is constant. Consequently, a Cholesky factorization is performed once during the initialization step of the 1 algorithm. In the following iterative process, the resolution of linear systems is reduced to the execution of forward and backward substitutions.

This section focuses on a case with a mesh resolution of $h = 1/16$, under two loading conditions: a traction of $0.05\,\text{daN/m}^2$, followed by $0.005\,\text{daN/m}^2$. Figures 2a and 2b demonstrate how the number of iterations and CPU time vary as a function of the penalty parameter. Figures 3 and 4 show the deformed configuration with the Von Mises effective stress distribution for the frictionless contact case and the Tresca friction case, respectively. Optimizing the penalty parameter significantly reduces both the iteration count and the computational time required for convergence. **Table 1** summarizes the optimal choices of the penalty parameter r_k^*, highlighting their impact on the global ADMM performance in terms of CPU time and iteration count.



(a) $0.05\,\text{daN/m}^2$



(b) $0.005\,\text{daN/m}^2$

Figure 2. Comparison of the number of iterations and CPU time versus the penalty parameter for mesh resolution $h = 1/16$: (a) $f_n = 0{,}05\,\text{daN/m}^2$, (b)$f_n = 0{,}005\,\text{daN/m}^2$.

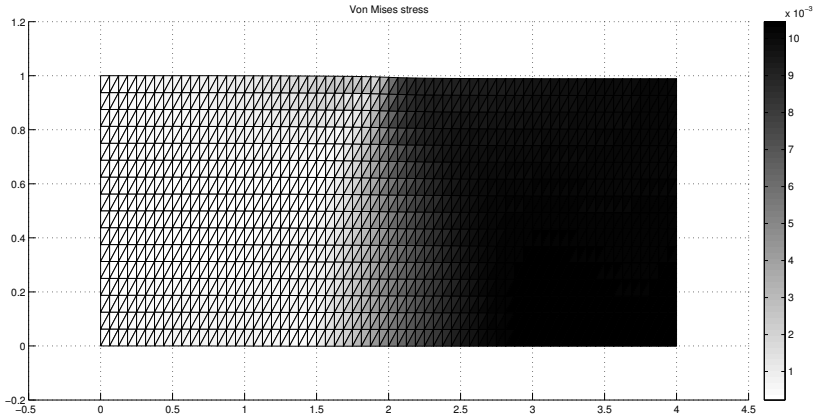


Figure 3. Deformed configuration with Von Mises effective stress distribution (Frictionless case).

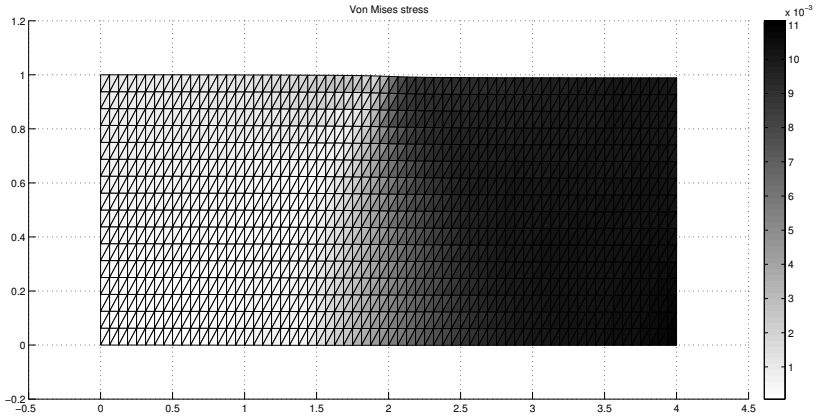
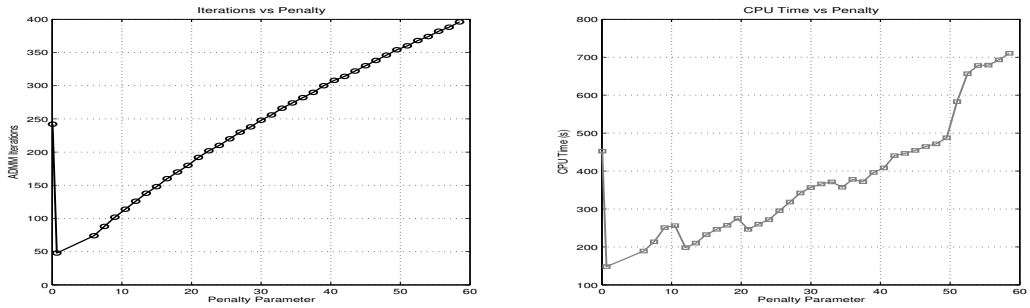


Figure 4. Deformed configuration with Von Mises effective stress distribution (Tresca friction case).

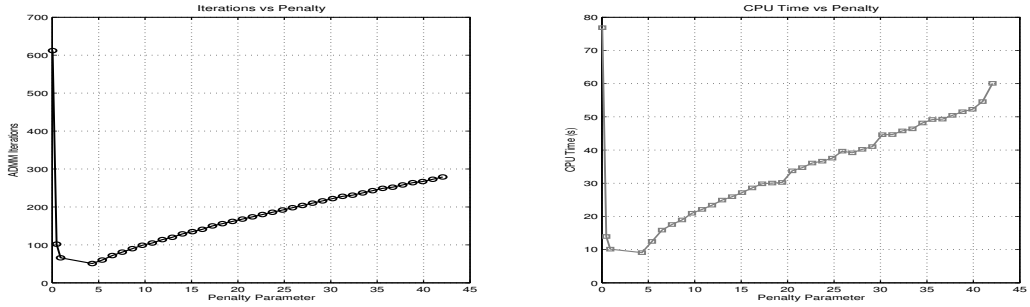
Table 1. Results for different values of f_n with optimal penalty parameters (mesh size $h = 1/16$)

f_n	r_1^*	r_2^*	CPU (s)	ADMM Iterations
0.005	4.3143	4.3143	7.3906	34
0.05	4.3143	4.3143	7.5469	34

Figures 5a illustrate the global number of ADMM iterations and the CPU time as functions of a fixed penalty parameter. In this case, the mesh has been refined to $h = 1/32$, and a normal force of $f_n = 0.05\,\text{daN/m}^2$ is applied. **Figure 5b presents the case of contact with Tresca friction, using a friction coefficient of 0.2, a coarser mesh resolution of $h = 1/16$, and the same normal load $f_n = 0.05$ daN/m^2.** On the other hand, **Table 2** presents the optimal choice of the penalty parameter, as well as the corresponding global number of ADMM iterations and CPU times, for both the mesh resolution $h = 1/32$ and the case with Tresca friction.



(a) Without friction ($h = 1/32$)



(b) With Tresca friction ($h = 1/16$)

Figure 5. Comparison of the number of iterations and CPU time versus the penalty parameter for mesh resolution $h = 1/32$ and normal load $f_n = 0{,}05\,\mathrm{daN/m^2}$: (a) without friction, (b) with Tresca friction.

Table 2. Optimal penalty parameters and performance results for different contact conditions

Case	r_1^*	r_2^*	CPU (s)	ADMM Iterations
Without friction, $h = 1/32$	5.9999	5.9999	159.7969	74
Tresca friction ($\mu = 0.2$), $h = 1/16$	4.3143	4.3143	8.1875	51

6. Conclusion

In this paper, we looked at how to automatically choose parameters for the Alternating Direction Method of Multipliers (ADMM) when solving the contact problem between an elastoplastic material and a rigid foundation. The material's behavior was modeled using Hencky's nonlinear elastic law. To show linear convergence for any penalty parameter, we created a fully dual version of the ADMM method. We demonstrated that the optimal penalty parameter can be found using the largest and smallest (nonzero) eigenvalues of a generalized symmetric eigenvalue problem. We also introduced an approximation method to determine the best penalty parameter. Numerical tests showed that our proposed method for selecting the parameter is a good alternative to the traditional sampling approach.

REFERENCES

1. Han, W. (2005). *A Posteriori Error Analysis via Duality Theory, with Applications in Modeling and Numerical Approximations*. New York, NY: Springer.
2. Han, W., Sofonea, M. (1999). Analysis and numerical approximation of an elastic frictional contact problem with normal compliance. *Appl. Math.*, **26**(4): 415-435.
3. Haslinger, J., Makinen, R. (1992). Shape optimization of elasto-plastic bodies under plane strains: sensitivity analysis and numerical implementation. *Struct. Opt.*, **4**(3-4): 133-141. Springer-Verlag.
4. Hlavacek, I., Necas, J. (1981). *Mathematical Theory of Elastic and Elasto-Plastic Bodies: An Introduction*. Studies in Applied Mechanics **3**. North Holland: Elsevier.
5. Benkhira, E. H., Fakhar, R., Mandyly, Y. (2019). Analysis and numerical approximation of a contact problem involving nonlinear Hencky-type materials with nonlocal Coulomb's friction law. *Numerical Functional Analysis and Optimization*, **40**(11): 1291-1314.
6. Fortin, M., Glowinski, R. (1983). *Augmented Lagrangian Methods: Applications to the Numerical Solution of Boundary-Value Problems*. Studies in Mathematics and Its Applications **15**. North Holland: Elsevier.
7. Glowinski, R. (2013). *Numerical Methods for Nonlinear Variational Problems*. Springer Science & Business Media.
8. Trémolieres, R., Lions, J. L., Glowinski, R. (2011). *Numerical Analysis of Variational Inequalities*. North Holland: Elsevier.
9. Glowinski, R., Le Tallec, P. (1989). *Augmented Lagrangian and Operator-Splitting Methods in Nonlinear Mechanics*. SIAM.
10. Han, W., Jensen, S., Shimansky, I. (1997). The Kačanov method for some nonlinear problems. *Applied Numerical Mathematics*, **24**(1): 57-79.
11. Duvaut, G., Lions, J.-L. (1976). *Inequalities in Mechanics and Physics*. Springer-Verlag.
12. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.*, **3**(1): 1–122.
13. He, B., Yuan, X. (2012). On the $O(1/n)$ convergence rate of the Douglas–Rachford alternating direction method. *SIAM J. Numer. Anal.*, **50**(2): 700–709.
14. Goldstein, T., Osher, S. (2009). The Split Bregman Method for L^1-Regularized Problems. *SIAM J. Imaging Sci.*, **2**(2): 323–343.
15. Haslinger, J., Kucera, R., Dostal, Z. (2004). An algorithm for the numerical realization of 3D contact problems with Coulomb friction. *J. Comput. Appl. Math.*, **164-165**: 387–408.
16. Koko, J. (2008). Uzawa block relaxation domain decomposition method for the two-body contact problem with Tresca friction. *Comput. Methods Appl. Mech. Engrg.*, **198**: 420–431.
17. Koko, J. (2009). Uzawa block relaxation domain decomposition method for the two-body frictionless contact problem. *Appl. Math. Lett.*, **22**: 1534–1538.
18. Koko, J. (2015). A MATLAB mesh generator for the two-dimensional finite element method. *Appl. Math. Comput.*, **250**: 650–664.
19. Koko, J. (2016). Fast MATLAB assembly of FEM matrices in 2D and 3D using cell array approach. *Int. J. Model. Simul. Sci. Comput.*, **7**: e1650010.
20. Meyer, C. D. (2023). *Matrix Analysis and Applied Linear Algebra*. SIAM.
21. Horn, R. A., Johnson, C. R. (2012). *Matrix Analysis*. Cambridge University Press.