



Computational simulation of manufacturing processes

Identification of inelastic parameters based on deep drawing forming operations using a global–local hybrid Particle Swarm approach



Miguel Vaz Jr.^{a,*}, Marco A. Luersen^b, Pablo A. Muñoz-Rojas^a,
Robson G. Trentin^c

^a Department of Mechanical Engineering, State University of Santa Catarina, Campus Universitario, 89219-710 Joinville, Brazil

^b Department of Mechanical Engineering, Federal University of Technology – Parana, Av. Sete de Setembro, 80230-901 Curitiba, Brazil

^c Department of Mechanical Engineering, Federal University of Technology – Parana, Via do Conhecimento, 85503-390 Pato Branco, Brazil

ARTICLE INFO

Article history:

Received 8 June 2015

Accepted 15 July 2015

Available online 23 February 2016

Keywords:

Deep drawing

Parameter identification

SQP

PSO

Nelder–Mead

ABSTRACT

Application of optimization techniques to the identification of inelastic material parameters has substantially increased in recent years. The complex stress–strain paths and high nonlinearity, typical of this class of problems, require the development of robust and efficient techniques for inverse problems able to account for an irregular topography of the fitness surface. Within this framework, this work investigates the application of the gradient-based Sequential Quadratic Programming method, of the Nelder–Mead downhill simplex algorithm, of Particle Swarm Optimization (PSO), and of a global–local PSO–Nelder–Mead hybrid scheme to the identification of inelastic parameters based on a deep drawing operation. The hybrid technique has shown to be the best strategy by combining the good PSO performance to approach the global minimum basin of attraction with the efficiency demonstrated by the Nelder–Mead algorithm to obtain the minimum itself.

© 2016 Académie des sciences. Published by Elsevier Masson SAS. All rights reserved.

1. Introduction

In the last two decades, the development of robust computational models has made it possible to efficiently simulate a wide range of metal-forming operations, such as forging, extrusion, rolling, and deep drawing, amongst many others. However, the success of the simulations depends upon the capacity of the constitutive models and respective material parameters to accurately reproduce the mechanical behaviour. In general, the industry obtains material parameters by means of mechanical tests described in technical standards, most of which based on the assumption of uniform stress states. However, the use of such parameters in the simulation of metal-forming processes (which involve large inelastic strains) may compromise the accuracy of the numerical predictions. In order to determine more realistic inelastic parameters, it has been proposed to use mechanical tests able to provide stress–strain paths similar to those of the actual forming operation. Furthermore, the identification of material parameters based upon optimization strategies has also been suggested, owing to a greater flexibility allowing one to handle complex deformation paths.

* Corresponding author.

E-mail addresses: Miguel.Vaz@udesc.br (M. Vaz Jr.), Luersen@utfpr.edu.br (M.A. Luersen), Pablo.Munoz@udesc.br (P.A. Muñoz-Rojas), RobsonTrentin@utfpr.edu.br (R.G. Trentin).

<http://dx.doi.org/10.1016/j.crme.2015.07.015>

1631-0721/© 2016 Académie des sciences. Published by Elsevier Masson SAS. All rights reserved.

Within this framework, the identification of inelastic parameters based on a *deep drawing* operation is addressed here by four different optimization strategies: (I) gradient-based Sequential Quadratic Programming combined with a sensitivity analysis computed using an enhanced semi-analytical finite difference scheme, (II) the Nelder–Mead downhill simplex algorithm, (III) Particle Swarm Optimization (PSO), and (IV) a global–local hybrid approach combining PSO and the Nelder–Mead algorithm. Emphasis is placed upon studies (III) and (IV) due to the robustness and efficiency of the techniques. This paper is structured as follows: Section 2 provides a brief description of the elastic–plastic approximation; Section 3 presents key principles and implementation characteristics of the aforementioned optimization methods; Section 4 discusses the deep drawing application, and Section 5 summarises some relevant recommendations.

2. Mechanical incremental boundary value problem

Deep drawing features a forming operation that requires an elastic–plastic formulation at finite strains. This work makes use of the von Mises constitutive modelling in conjunction with nonlinear isotropic hardening. The governing equations of the mechanical problem comprise the linear and angular momentum equations, respectively,

$$\begin{cases} \operatorname{div}[\boldsymbol{\sigma}] + \rho \mathbf{b} = \mathbf{0} \\ \boldsymbol{\sigma} = \boldsymbol{\sigma}^T \end{cases} \quad \text{with} \quad \begin{cases} \mathbf{u} = \mathbf{u}^a \text{ for } \mathbf{x} \in \partial\Omega^u \\ \boldsymbol{\sigma} \mathbf{n} = \mathbf{t}^a \text{ for } \mathbf{x} \in \partial\Omega^t \end{cases} \quad \text{and} \quad \begin{cases} \partial\Omega = \partial\Omega^u \cup \partial\Omega^t \\ \partial\Omega^u \cap \partial\Omega^t = \emptyset \end{cases} \quad (1)$$

where $\boldsymbol{\sigma}$ is the Cauchy stress tensor, ρ is the specific mass, \mathbf{b} indicates body forces and \mathbf{u} are displacements. The boundary conditions are defined so that \mathbf{n} is the outward normal unit vector and \mathbf{u}^a and \mathbf{t}^a are prescribed displacements and loads, respectively.

The finite element method based on an updated Lagrangian formulation is used to approach the mechanical boundary value problem. The first step is the application of the principle of virtual work to Equation (1), so that

$$\delta W_i = \delta W_e \quad \text{where} \quad \begin{cases} \delta W_i = \int_{\Omega} \boldsymbol{\sigma} : \frac{\partial \delta \mathbf{u}}{\partial \mathbf{x}} \, d\Omega \\ \delta W_e = \int_{\Omega} \rho \mathbf{b} \cdot \delta \mathbf{u} \, d\Omega + \int_{\partial\Omega^t} \mathbf{t} \cdot \delta \mathbf{u} \, da \end{cases} \quad (2)$$

in which $\delta \mathbf{u}$ is the virtual displacement, and δW_i and δW_e represent the virtual work related to the internal and external forces, respectively. In addition to the finite element discretisation procedure, the Newton–Raphson iterative method is used to solve the material and geometrically nonlinear problem, which, written for displacements \mathbf{u}_{n+1} at time t_{n+1} , can be summarised as

$$\mathbf{R}(\mathbf{u}_{n+1}) = \mathbf{F}_{n+1}^{\text{Int}} - \mathbf{F}_{n+1}^{\text{Ext}} \quad (3)$$

where \mathbf{R} is the residual of the nonlinear problem, $\mathbf{F}_{n+1}^{\text{Int}}$ and $\mathbf{F}_{n+1}^{\text{Ext}}$ are the internal and external global force vectors, respectively, and the subscript $n + 1$ indicates the current time step (solution increment). Linearization is performed by expanding (3) in a Taylor series and neglecting the higher-order terms,

$$\mathbf{R}(\mathbf{u}_{n+1}^{(k+1)}) = \mathbf{R}(\mathbf{u}_{n+1}^{(k)} + \delta \mathbf{u}^{(k+1)}) \cong \mathbf{R}(\mathbf{u}_{n+1}^{(k)}) + \left. \frac{\partial \mathbf{R}}{\partial \mathbf{u}_{n+1}} \right|_{\mathbf{u}_{n+1}^{(k)}} \delta \mathbf{u}^{(k+1)} \quad (4)$$

and requiring the residual at the $(k + 1)$ Newton–Raphson iteration, $\mathbf{R}(\mathbf{u}_{n+1}^{(k+1)})$, to be zero, so that

$$\mathbf{K}_T \delta \mathbf{u}^{(k+1)} = -\mathbf{R}(\mathbf{u}_{n+1}^{(k)}), \quad \text{and} \quad \begin{cases} \mathbf{u}_{n+1}^{(k+1)} = \mathbf{u}_{n+1}^{(k)} + \delta \mathbf{u}^{(k+1)} \text{ or} \\ \mathbf{u}_{n+1}^{(k+1)} = \mathbf{u}_n + \Delta \mathbf{u}^{(k+1)} \end{cases} \quad (5)$$

where \mathbf{u}_n is the known displacements of the solution increment n , $\Delta \mathbf{u}^{(k+1)} = \Delta \mathbf{u}^{(k)} + \delta \mathbf{u}^{(k+1)}$ is the total displacement increment of time step $n + 1$, and $\mathbf{K}_T = \left[\frac{\partial \mathbf{R}(\mathbf{u}_{n+1}^{(k)})}{\partial \mathbf{u}_{n+1}} \right]$ is the tangent stiffness matrix, which represents the derivative of the residual with respect to displacements for the iterative step (k) of the current solution increment $n + 1$. For the sake of objectivity, this work will not discuss the plasticity model. The reader is referred to Souza Neto et al. [1] for further insights, including a detailed description of the method used to obtain the tangent stiffness matrix.

3. Parameter identification and the optimization problem

3.1. The optimization problem

Parameter identification in elastic–plastic problems consists in finding elastic and/or inelastic material parameters of the constitutive model using experimental data from mechanical tests. The approach adopted in the present work uses inverse problem techniques based on unconstrained optimization, which is generally defined as

$$\min_{\mathbf{p}} g(\mathbf{p}), \quad \mathbf{p} \in R^{n_d} \tag{6}$$

under side constraints $p_i^{\text{inf}} \leq p_i \leq p_i^{\text{sup}}$, $i = 1, \dots, n_d$, where p_i^{sup} and p_i^{inf} are the upper and lower bounds of the design variable p_i (material parameter), respectively, n_d is the number of design variables, $g(\mathbf{p})$ is the objective function (or fitness), and $\mathbf{p} = [p_1, \dots, p_i, \dots, p_{n_d}]^T$ is the design vector.

The objective function to be minimised represents the quadratic mean relative difference between experimental measures and corresponding computed responses, and is written as

$$g(\mathbf{p}) = \sqrt{\frac{1}{N} \sum_{j=1}^N \xi_j \left(\frac{F_j^{\text{exp}} - F_j^{\text{FE}}}{F_j^{\text{exp}}} \right)^2} \tag{7}$$

where N is the number of experimental points, F^{exp} and F^{FE} are experimental and computed forming loads, respectively, and ξ is a weight function.

In this work, the solution to the optimization problem stated in Equation (6) is accomplished by using the gradient-based Sequential Quadratic Programming (SQP) algorithm (Section 3.2), the Nelder–Mead downhill simplex algorithm (Section 3.3), Particle Swarm Optimization (Section 3.4), and a global–local hybrid scheme (Section 3.5).

3.2. Sequential Quadratic Programming algorithm

Gradient-based optimization strategies were the first choice to approach parameter identification in the last decade due to their higher convergence rate when compared to direct search algorithms. The literature shows several applications: for instance, Springmann and Kuna [2] used the Levenberg–Marquardt method to identify inelastic parameters in damaged materials, and Frař et al. [3] adopted the conjugate gradient method to determine viscoplastic parameters of high-strength steels (see Muřoz-Rojas et al. [4] and references therein for further details on the application of gradient-based schemes to the identification of inelastic parameters). Nevertheless, the existence of local minima has been an important hindrance to solve this class of problems, as discussed by Ponthot and Kleinermann [5]. The authors used in cascade six different gradient-based algorithms in order to overcome convergence issues when addressing the identification of material parameters for a classical von Mises constitutive model.

A quick search in scientific databases shows that SQP-based optimization methods have been widely used in general engineering applications. More importantly, their success in elastoplastic [6] and structural mechanics [7] problems has recommended this strategy to the present parameter identification application. This work uses Schittkowski's SQP implementation for parallel computing [8], so that, for iteration (q), the quadratic problem is written as

$$\begin{aligned} \min_{\mathbf{p}} \quad & \frac{1}{2} \mathbf{d}^{(q)T} \mathbf{B}(\mathbf{p}^{(q)}) \mathbf{d}^{(q)} + \nabla g(\mathbf{p}^{(q)})^T \mathbf{d}^{(q)}, \quad \mathbf{p}, \mathbf{d} \in R^{n_d} \\ \text{s.t.} \quad & \nabla f_j(\mathbf{p}^{(q)}) \mathbf{d}^{(q)T} + f_j(\mathbf{p}^{(q)}) \geq 0 \\ & j = 1, \dots, m \end{aligned} \tag{8}$$

under side constraints $p_i^{\text{inf}} \leq p_i \leq p_i^{\text{sup}}$, $i = 1, \dots, n_d$, in which \mathbf{d} is the search direction and $\mathbf{B}(\mathbf{p}) \in R^{n_d \times n_d}$ is the Hessian of the Lagrangian function $L(\mathbf{p}) = g(\mathbf{p}) + \lambda^T \mathbf{f}(\mathbf{p})$, where λ are Lagrangian multipliers. The bounds of parameters \mathbf{p} are included as general inequality constraints $\mathbf{f}(\mathbf{p})$. The Hessian \mathbf{B} is evaluated based on the BFGS method [9], thus requiring only first-order derivatives. The parameters for iteration ($q + 1$) are, therefore, determined as $\mathbf{p}^{(q+1)} = \mathbf{p}^{(q)} + \alpha^{(q)} \mathbf{d}^{(q)}$, where $\alpha^{(q)}$ is a steplength parameter computed by minimising the unidimensional function $\phi = g(\mathbf{p}) + \lambda^T \min[0, \mathbf{f}(\mathbf{p})]$.

In addition, gradient-based schemes require a sensitivity analysis, i.e. computation of the derivatives of the objective function with respect to the design variables. In the present work, this task is accomplished by using an enhanced semi-analytical finite difference method [10].

The derivative of the objective function $g(\mathbf{p})$ (Equation (7)) with respect to the design variable p_i is obtained as

$$\frac{\partial g(\mathbf{p})}{\partial p_i} = \frac{1}{2} \left[\frac{1}{N} \sum_{j=1}^N \xi_j \left(\frac{F_j^{\text{FE}} - F_j^{\text{exp}}}{F_j^{\text{exp}}} \right)^2 \right]^{-1/2} \frac{1}{N} \sum_{j=1}^N 2\xi_j \left(\frac{F_j^{\text{FE}} - F_j^{\text{exp}}}{F_j^{\text{exp}}} \right) \frac{1}{F_j^{\text{exp}}} \frac{dF_j^{\text{FE}}}{dp_i} \tag{9}$$

in which dF_j^{FE}/dp_i represents the sensitivity of the forming load with respect to the material parameters.

The sensitivity procedure requires solutions for the original, p_i , and perturbed, $p_i + \delta p_i$, design variables. In the present method, both problems are solved in sequence and, at the end of every step, the results of the original problem are stored and the first perturbed problem is loaded for the iterative solution within the same solution increment. The converged solution to the original problem is used as initial estimate of the perturbed problem (which is also solved iteratively until convergence is reached). Moreover, the tangent stiffness matrix, \mathbf{K}_T , of the last iteration of the original problem is used to solve the perturbed problem. This procedure is applied for each perturbed variable and repeated for each load increment, from which the sensitivity of each design variable is computed using finite differences.

From Equation (5), the displacement increment of the original problem is given by

$$\mathbf{u}_{n+1}^{(k+1)} = \mathbf{u}_{n+1}^{(k)} - \mathbf{K}_T^{-1} \mathbf{R}(\mathbf{u}_{n+1}^{(k)}; \mathbf{p}) \quad (10)$$

where \mathbf{R} is the residual of the equilibrium equation at the solution step $n + 1$ and k -th iteration, and \mathbf{K}_T is the tangent stiffness matrix. When the converged solution to the original problem is used as an initial estimate for the perturbed problem of a forward finite difference scheme, it is important to subtract the converged residual of the perturbed problem from the residual of the original problem, as

$$\mathbf{u}_{n+1}^{(k+1), \Delta i} = \mathbf{u}_{n+1}^{(k), \Delta i} - \mathbf{K}_T^{-1} [\mathbf{R}(\mathbf{u}_{n+1}^{(k), \Delta i}; \mathbf{p} + \delta \mathbf{p}) - \mathbf{R}(\mathbf{u}_{n+1}; \mathbf{p})] \quad (11)$$

where the superscript Δi indicates the perturbed solution, $\mathbf{u}_{n+1}^{(0), \Delta i} = \mathbf{u}_{n+1}$ and $\delta \mathbf{p}$ contains the perturbation of the i -th design variable, δp_i . By using forward finite differences, the sensitivity of the internal force with respect to the design variable p_i is

$$\frac{d\mathbf{F}_{n+1}}{dp_i} \approx \frac{\Delta \mathbf{F}_{n+1}}{\Delta p_i} = \frac{\mathbf{F}_{n+1}^{\Delta i} - \mathbf{F}_{n+1}}{\delta p_i} \quad (12)$$

where $\mathbf{F}_{n+1}^{\Delta i}$ and \mathbf{F}_{n+1} are the forming loads of the perturbed and unperturbed problems for time step $n + 1$.

It is important to mention that the preceding technique has proven to be quite flexible and, contrary to the classical finite difference scheme, the present sensitivity analysis has been successfully used for problems involving *remeshing* in conjunction with von Mises elastic–plastic materials [10]. It should also be remarked that a similar sensitivity analysis procedure was developed by Kleinermann [11] based on central finite differences (without provision for remeshing).

3.3. The downhill simplex method – the Nelder–Mead algorithm

The gradient-free downhill simplex method, also known as *Nelder–Mead* algorithm [12], is based upon an earlier work of Spendley et al. [13] on simplex design in optimization. The original technique defines a regular polytope of $n_d + 1$ vertices (in a n_d -dimensional design space) which moves towards the optimum by replacing the worst vertex by its reflection around the centroid of the hyper-plane formed by the remaining vertices. Nelder and Mead [12] introduced additional operations to make it possible that the simplex “better adapts to the local landscape by elongating down along inclined planes, changing direction on encountering a valley at an angle, and contracting in the neighbourhood of a minimum.”

The good performance of the downhill simplex method, referred to from this point on as *Nelder–Mead* (NM) algorithm, has instigated a large number of researchers to propose variants to the original technique and to investigate its applicability to several different areas. The convergence assessment and a NM variant discussed by Lagarias et al. [14] have prompted a new surge of interest in the Nelder–Mead optimization method in recent years. However, in spite of its widespread use in many different fields, it is noteworthy that there are few studies reporting the application of the NM scheme to the identification of inelastic material parameters. For instance, Banabic et al. [15] used the NM method in the identification of inelastic parameters based on biaxial tensile tests. The same strategy was adopted by Pannier et al. [16] to determine elastic–plastic constitutive parameters using the Virtual Fields Method associated with tensile tests. Shanpo et al. [17] also described the application of the NM method in order to obtain the inelastic parameters of geotechnical materials. The availability of the NM scheme in some commercial simulators has encouraged further applications (most of which using the NM routine as a black box). Nevertheless, the lack of flexibility of some commercial codes in allowing one to modify the NM routine and the corresponding control parameters may render the identification process inefficient or even unsuccessful, as reported by Helfenstein [18].

The Nelder–Mead algorithm contains three important elements: (i) creation of the initial simplex, (ii) search along a given direction and formation of a new polytope, and (iii) shrinkage of the polytope towards the best vertex.

There are several strategies to create the initial simplex from an initial estimate. The most common approaches define the initial geometry from placing the estimate either in a vertex [13] or in the centroid of the polytope [19]. Aiming at maintaining a regular shape of the simplex, the present implementation uses a parameterised version of the latter. Initially, the first vertex, \mathbf{p}_1 , is defined from the hyper-coordinates of the initial estimate, \mathbf{p}_d ,

$$[p_i]_1 = [p_i]_d - \frac{a_i + (n_d - 1) \times b_i}{n_d + 1} \quad (13)$$

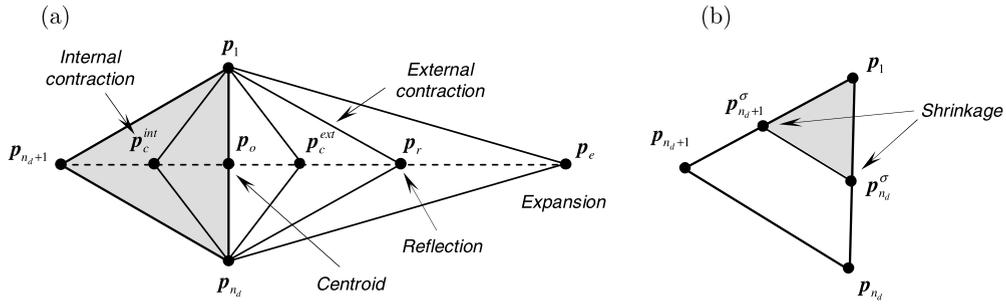


Fig. 1. Conceptual illustration of the Nelder–Mead method for a two-dimensional design space [20]: (a) search direction and (b) shrinkage of the polytope.

where $[p_i]_1$ and $[p_i]_d$ represent a design variable (material parameter) of vertex 1 and initial estimate, respectively, $i = 1, \dots, n_d$, and a_i and b_i are computed as

$$a_i = d_i \times \frac{\sqrt{n_d + 1} + (n_d - 1)}{n_d \sqrt{2}} \quad \text{and} \quad b_i = d_i \times \frac{\sqrt{n_d + 1} - 1}{n_d \sqrt{2}} \quad (14)$$

in which $d_i = h \times (p_i^{\text{sup}} - p_i^{\text{inf}})$ is the edge length, defined as a fraction of the search space for the design variable p_i , where $h \in [0, 1]$ is the initial simplex edge factor. The remaining vertices of the initial polytope, $\mathbf{p}_2, \dots, \mathbf{p}_m, \dots, \mathbf{p}_{n_d+1}$, are determined from the first vertex, \mathbf{p}_1 , as

$$[p_i]_m = [p_i]_1 + \begin{cases} a_i & \text{if } m = i - 1 \\ b_i & \text{if } m \neq i - 1 \end{cases} \quad (15)$$

where $\mathbf{p}_m = [p_1, \dots, p_i, \dots, p_{n_d}]^T$ and $m = 2, \dots, n_d + 1$.

The second element of the NM scheme is the search along a direction defined by the worst vertex, \mathbf{p}_{n_d+1} , and the centroid, \mathbf{p}_o , of the hyper-plane formed by the remaining vertices, $\mathbf{p}_1, \dots, \mathbf{p}_{n_d}$, as illustrated in Fig. 1a for a two-dimensional design space [20]. Initially, a reflection, \mathbf{p}_r , of the worst vertex around the centroid, \mathbf{p}_o , is performed. The worst vertex is then replaced by the best point selected along the search direction after the following possible operations: expansion, \mathbf{p}_e , internal contraction, $\mathbf{p}_c^{\text{int}}$, external contraction, $\mathbf{p}_c^{\text{ext}}$, or the reflected point itself. Noticeably, after reflection or expansion, the prospective new vertex is tested against the minimum and maximum limits of each design variable and a projection is performed if necessary, as proposed by Luersen and Le Riche [21].

Finally, if the best vertex is sufficiently close to the minimum, the shrinkage operation is performed, as depicted in Fig. 1b. Each of the aforementioned operations can be individually tuned by control coefficients, which have been originally defined as follows: reflection $\rho = 1$, expansion $\gamma = 2$, contraction $\beta = 0.5$, and shrinkage $\sigma = 0.5$. Table 1 presents a detailed description of the Nelder–Mead scheme implemented in the present work (using the considerations proposed by Lagarias et al. [14]).

The stopping criterion is established by either the fitness of the best vertex or the relative difference between the objective function of the worst and best vertices of the polytope, as

$$g(\mathbf{p}_1^{(k)}) \leq \text{TO}_{L\phi_g} \quad \text{and} \quad \phi_{\text{NM}} = \frac{g(\mathbf{p}_{n_d+1}^{(k)}) - g(\mathbf{p}_1^{(k)})}{g(\mathbf{p}_{n_d+1}^{(0)}) - g(\mathbf{p}_1^{(0)})} \leq \text{TO}_{L\phi_{\text{NM}}} \quad (16)$$

in which $\text{TO}_{L\phi_g}$ and $\text{TO}_{L\phi_{\text{NM}}}$ are the assumed convergence limits.

3.4. Particle Swarm Optimization

Particle Swarm Optimization was proposed by Eberhart and Kennedy [22,23] based on the concept of social behaviour of populations. The method accounts for a combination of social interactions and individual cognitive abilities by attributing to each particle (i) inertia, (ii) personal history, and (iii) neighbourhood effects:

- (i) *inertia effect*: the inertia increases the search capacity by leading the particle to follow along its previous direction;
- (ii) *cognitive effect*: the personal history aims at increasing its cognitive ability by adding a velocity component which is associated with its best position;
- (iii) *social effect*: the neighborhood (or social) effects account for location(s) of the best particle(s) of the swarm.

The potential of the method to avoid local minima has dictated its successful application to fields as diverse as image and sound analysis, visualisation and computer graphics, robotics, traffic management, maintenance planning, etc. [24]. In recent

Table 1
Nelder–Mead algorithm.

(i) Set $k = 0$ and generate the initial simplex, $\mathbf{p}^{(0)}$
 $\mathbf{p}^{(0)} = \{\mathbf{p}_1^{(0)}, \dots, \mathbf{p}_m^{(0)}, \dots, \mathbf{p}_{n_d+1}^{(0)}\}$ where $\mathbf{p}_m^{(0)} = [p_1^{(0)}, \dots, p_i^{(0)}, \dots, p_{n_d}^{(0)}]^T$
 in which $n_d + 1$ is the number of vertices
 WHILE ($g(\mathbf{p}^{(k)}) > \text{TOL}_{\phi_g}$) or ($\phi_{\text{NM}} > \text{TOL}_{\phi_{\text{NM}}}$) DO

(ii) Compute the objective function if necessary and sort simplex vertices
 $g(\mathbf{p}_1^{(k)}) < g(\mathbf{p}_2^{(k)}) < \dots < g(\mathbf{p}_m^{(k)}) < \dots < g(\mathbf{p}_{n_d+1}^{(k)})$

(iii) Compute *reflexion*^(*) of the worst vertex and corresponding objective function
 $\mathbf{p}_r^{(k)} \leftarrow \mathbf{p}_o^{(k)} + \rho(\mathbf{p}_o^{(k)} - \mathbf{p}_{n_d+1}^{(k)})$ where $\mathbf{p}_o^{(k)} = \frac{1}{n_d} \sum_{m=1}^{n_d} \mathbf{p}_m^{(k)}$ and $g(\mathbf{p}_r^{(k)})$

(iv) Compare $g(\mathbf{p}_r^{(k)})$ with the best, second worst and worst vertices
 IF $g(\mathbf{p}_r^{(k)}) \leq g(\mathbf{p}_1^{(k)})$ THEN Perform *expansion*^(*) of the polytope
 $\mathbf{p}_e^{(k)} \leftarrow \mathbf{p}_o^{(k)} + \gamma(\mathbf{p}_r^{(k)} - \mathbf{p}_o^{(k)})$ and $g(\mathbf{p}_e^{(k)})$
 IF $g(\mathbf{p}_e^{(k)}) < g(\mathbf{p}_r^{(k)})$ THEN $\mathbf{p}_{n_d+1}^{(k+1)} \leftarrow \mathbf{p}_e^{(k)}$
 ELSE $\mathbf{p}_{n_d+1}^{(k+1)} \leftarrow \mathbf{p}_r^{(k)}$
 END IF
 ELSE IF $g(\mathbf{p}_r^{(k)}) \leq g(\mathbf{p}_{n_d}^{(k)})$ THEN $\mathbf{p}_{n_d+1}^{(k+1)} \leftarrow \mathbf{p}_r^{(k)}$
 ELSE Perform *contraction* of the polytope
 IF $g(\mathbf{p}_r^{(k)}) < g(\mathbf{p}_{n_d+1}^{(k)})$ THEN $\mathbf{p}_{n_d+1}^{(k+1)} \leftarrow \mathbf{p}_r^{(k)}$
 $\mathbf{p}_c^{(k)} \leftarrow \mathbf{p}_o^{(k)} + \beta(\mathbf{p}_{n_d+1}^{(k)} - \mathbf{p}_o^{(k)})$ and $g(\mathbf{p}_c^{(k)})$
 IF $g(\mathbf{p}_c^{(k)}) > g(\mathbf{p}_{n_d+1}^{(k)})$ THEN Perform *shrinkage* of the polytope
 $\mathbf{p}_m^{(k+1)} \leftarrow \mathbf{p}_1^{(k)} + \sigma(\mathbf{p}_m^{(k)} - \mathbf{p}_1^{(k)})$
 $m = 2, \dots, n_d + 1$
 ELSE
 $\mathbf{p}_{n_d+1}^{(k+1)} \leftarrow \mathbf{p}_c^{(k)}$
 END IF
 END IF
 END IF
 $k \leftarrow k + 1$
 END WHILE
 (v) $\mathbf{p}^{\text{end}} \leftarrow \mathbf{p}_1^{(k)}$

(*) The new vertex location is verified against the side constraints, $p_i \in [p_i^{\text{inf}}, p_i^{\text{sup}}]$
 IF $p_i^{(k)} > p_i^{\text{sup}}$ THEN $p_i^{(k)} \leftarrow p_i^{\text{sup}}$ or IF $p_i^{(k)} < p_i^{\text{inf}}$ THEN $p_i^{(k)} \leftarrow p_i^{\text{inf}}$

years, several variants have been proposed aiming at improving neighbourhood conditions and particle interaction rules amongst many other features. The extensive survey presented by Sedighizadeh and Masehian [24], Schutte and Groenwold [25], and Blum and Li [26] demonstrate the flexibility, efficiency, and robustness of the PSO technique.

Despite its ample spectrum of applications, there are relatively few investigations on the PSO suitability to the identification of *thermal* or *mechanical constitutive parameters*. The literature shows some recent advancements in PSO applications to inverse *thermal* problems (e.g., [27–29]); however, the numerical simulation of *mechanical* problems makes use of more complex constitutive relations and requires more elaborate mathematical formulations and computational modelling strategies. Nevertheless, a brief survey shows some new applications to structural mechanics, such as optimization of pressure vessels [30], truss [31], and composite [32] structures. There are even fewer studies on the application of PSO-based methods to the identification of *mechanical material parameters*. For instance, the identification of visco-elastic parameters in rock mass modelling and visco-elastic visco-plastic parameters for polymeric trusses were described by Feng et al. [33] and Carniel et al. [34], respectively. Applications of PSO-based techniques to composite materials and functionally graded materials (FGM) were addressed by Hornig and Flowers [35] and Fereidoon et al. [36], respectively. A discussion on the PSO performance in the identification of inelastic parameters was introduced by Vaz Jr. et al. [37] within the framework of tensile tests of cylindrical specimens. The comparative studies featuring PSO and Genetic Algorithms indicate that, for a classical von Mises material, the latter was not capable to converge with the same rate and accuracy level exhibited by the PSO method.

The outline of the PSO algorithm used in the present work is shown in Table 2 and its general concept is illustrated in Fig. 2 for a two-dimensional problem. The first step consists in generating a random population, $\mathbf{p}^{(0)}$, of size n_p , and corresponding velocities, $\mathbf{v}^{(0)}$, with particles defined in a design space of n_d dimensions. For the current population, the individual, \mathbf{p}_{ib} , and global, \mathbf{p}_{gb} , best values are determined, respectively, as the best location of the particle along its history and the best particle of the previous step. The new velocities are computed encompassing inertia, cognitive and social effects tuned by the control weights w , φ_1 and φ_2 , respectively. The stochastic character is included in the random operator $\mathbf{U}(0, \varphi_b)$, in which $0 \leq \mathbf{U}(0, \varphi_b) \leq \varphi_b$ and $b = 1, 2$. The maximum velocity is imposed as a fraction, w_i , of the difference between the upper, p_i^{sup} , and lower, p_i^{inf} limits of the design variable. The next location of particles is evaluated and verified against the side constraints, and the projection of the design variable, p_i , is imposed if necessary.

Table 2
PSO algorithm.

-
- (i) Set $k = 0$ and generate the initial population and corresponding velocities
 $\mathbf{p}^{(0)} = \{\mathbf{p}_1^{(0)}, \dots, \mathbf{p}_m^{(0)}, \dots, \mathbf{p}_{n_p}^{(0)}\}$ and $\mathbf{v}^{(0)} = \{\mathbf{v}_1^{(0)}, \dots, \mathbf{v}_m^{(0)}, \dots, \mathbf{v}_{n_p}^{(0)}\}$
 in which n_p is the number of particles, n_d is the number of design variables, and
 $\mathbf{p}_m^{(0)} = [p_1^{(0)}, \dots, p_i^{(0)}, \dots, p_{n_d}^{(0)}]^T$ and $\mathbf{v}_m^{(0)} = [v_1^{(0)}, \dots, v_i^{(0)}, \dots, v_{n_d}^{(0)}]^T$
 WHILE ($g(\mathbf{p}^{(k)}) > \text{TOL}_{\phi_g}$) or ($\phi_{\text{PSO}} > \text{TOL}_{\phi_{\text{PSO}}}$) DO
 - (ii) Evaluate individual and global best particles, $\mathbf{p}_{ib}^{(k)}$, and $\mathbf{p}_{gb}^{(k)}$,
 IF $g(\mathbf{p}_m^{(k)}) < g(\mathbf{p}_{ib}^{(k)})$ THEN $\mathbf{p}_{ib}^{(k)} \leftarrow \mathbf{p}_m^{(k)}$ and
 IF $g(\mathbf{p}_m^{(k)}) < g(\mathbf{p}_{gb}^{(k)})$ THEN $\mathbf{p}_{gb}^{(k)} \leftarrow \mathbf{p}_m^{(k)}$
 - (iii) Compute new velocities, $\mathbf{v}^{(k+1)}$
 $\mathbf{v}^{(k+1)} \leftarrow w\mathbf{v}^{(k)} + \mathbf{U}(0, \varphi_1) \otimes (\mathbf{p}_{ib}^{(k)} - \mathbf{p}^{(k)}) + \mathbf{U}(0, \varphi_2) \otimes (\mathbf{p}_{gb}^{(k)} - \mathbf{p}^{(k)})$
 - (iv) Verify velocities against maximum limits of each design variable
 $v_i^{\text{max}} \leftarrow w_i(p_i^{\text{sup}} - p_i^{\text{inf}})$ where $w_i \in [0, 0.5]$
 IF $|v_i^{(k)}| > v_i^{\text{max}}$ THEN $|v_i^{(k)}| \leftarrow v_i^{\text{max}}$
 - (v) Compute new location^(*) of all particles, $\mathbf{p}^{(k+1)}$, (design vectors)
 $\mathbf{p}^{(k+1)} \leftarrow \mathbf{p}^{(k)} + \mathbf{v}^{(k+1)}$
 $k \leftarrow k + 1$
 END WHILE
 - (vi) $\mathbf{p}^{\text{end}} \leftarrow \mathbf{p}_{gb}^{(k)}$
-

^(*) The location is verified against the side constraints, $p_i \in [p_i^{\text{inf}}, p_i^{\text{sup}}]$
 IF $p_i^{(k)} > p_i^{\text{sup}}$ THEN $p_i^{(k)} \leftarrow p_i^{\text{sup}}$ or IF $p_i^{(k)} < p_i^{\text{inf}}$ THEN $p_i^{(k)} \leftarrow p_i^{\text{inf}}$

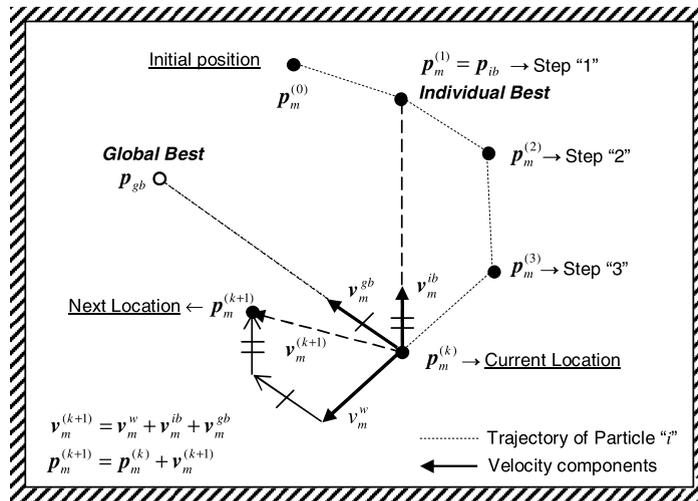


Fig. 2. Conceptual illustration of Particle Swarm Optimization for a two-dimensional design space [37].

The stopping criteria adopted for the PSO method are similar to those used for the NM scheme, i.e. the objective function of the best particle, $g(\mathbf{p}_{gb}^{(k)}) \leq \text{TOL}_{\phi_g}$, and the relative difference between the maximum and minimum values of the objective function of the n_s best particles, $\phi_{\text{PSO}} = [g(\mathbf{p}_{n_s}^{(k)}) - g(\mathbf{p}_{gb}^{(k)})] / [g(\mathbf{p}_{n_s}^{(0)}) - g(\mathbf{p}_{gb}^{(0)})] \leq \text{TOL}_{\phi_{\text{PSO}}}$.

3.5. Global–local hybrid schemes

The concept of using *global–local* hybrid schemes in general optimization problems is not new. The present work adopts a strategy in which the *global* search aims at reducing the search space, whereas the *local* search at improving a trade-off ratio between accuracy and convergence rate. When using such schemes, the potential to avoid local minima is the prime requirement to select the *global* search method, i.e. the application of the *global* algorithm to the target problem should be robust in detecting the global minimum basin of attraction. In addition, the *local* algorithm should be able to obtain the minimum at higher convergence rate from an initial estimate located at a given distance of the optimum. Therefore, the general characteristics of the present scheme are:

- (i) *global search*: the global search is performed by Particle Swarm Optimization – the initial search space is defined in order to allow ample range for each material parameter – the exploration stage. Global – local transition is determined by a predefined criterion;

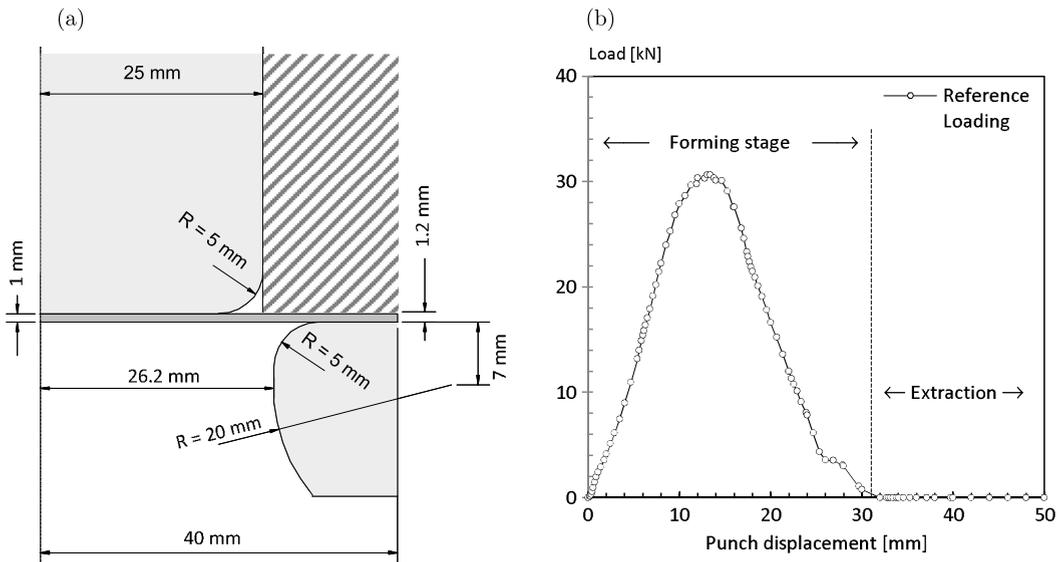


Fig. 3. (a) Initial geometry (axisymmetrical model) [20] and (b) reference loading curve.

(ii) *local search*: the local search is accomplished by the Nelder–Mead method using the design variables (material parameters) of the best particle as initial estimate, i.e. the second stage aims at increasing the exploitation capacity at reduced computational cost.

It is noteworthy that, in recent years, *global–local* hybrid techniques have been successfully applied to the identification of inelastic parameters. Chaparro et al. [38], studying anisotropic materials, performed *global* search using Genetic Algorithms and *local* search via the Levenberg–Marquardt gradient-based method. Following such strategy, a combination of Genetic Algorithms and of the Globally Convergent Method of Moving Asymptotes (GCMMA) was proposed by Muñoz-Rojas et al. [39] to determine material parameters for damaged materials. Similarly, De Carvalho et al. [40] also used Genetic Algorithms associated with the Levenberg–Marquardt gradient-based method to obtain inelastic and hyperelastic material parameters.

The literature shows that PSO–NM *global–local* algorithms have already been applied to other classes of optimization problems. For instance, Kumar and Singh [41] discussed the application of the PSO–NM scheme to energy distribution systems. Kayhan et al. [42] proposed a ‘general’ optimizer based on the PSO–NM *global–local* hybrid algorithm and presented an application to pressure vessel design. The multidisciplinary character of the PSO–NM algorithm can be observed in the work of Barzinpour et al. [43], who discussed the application to economic design of control charts in manufacturing processes. Application of the PSO–NM technique to the identification of inelastic parameters based on tensile tests has been recently reported by Vaz Jr. et al. [44]. It is noteworthy that preliminary issues on the application of the PSO–NM hybrid scheme to deep drawing operations were discussed by the authors in an early work [45], and its insertion in a broader context of parameter identification problems was briefly discussed in [20].

4. Numerical examples and discussion

4.1. The deep drawing operation

Deep drawing is a metal forming process by which a punch is used to force a sheet metal blank, usually held by a blank holder, to flow between the surfaces of a punch and die [46], as depicted in Fig. 3a. The process comprises two steps: (i) forming and (ii) extraction stages. In the present case, the forming stage takes place for a punch displacement up to 30 mm, followed by extraction in the remaining 20 mm, as shows the forming loaded indicated in Fig. 3b. The identification uses only the first step, since no load associated with plastic deformation is generally observed during extraction, i.e. the weight functions assigned to Equation (6) for stages (i) and (ii) are $\xi_j = 1$ and $\xi_j = 10^{-6}$, respectively.

The suitability of the SQP, NM, PSO and PSO–NM schemes is investigated in this work by defining a demanding inverse problem: to recover the set of reference hardening parameters of the Ramberg–Osgood model given in Table 3. The reference load–displacement curve is defined by applying a small random perturbation to the forming load determined by the reference parameters, as shown in Fig. 3b. It is important to emphasize that the proposed testing is even more exigent than the parameter identification problem discussed in Vaz Jr. et al. [37], i.e., in the present case, the individual use of the well-established optimization schemes presented low performance. Therefore, even though it looks apparently simple, the small number of design variables highlights even more the degree of difficulty of the proposed identification problem.

Table 3
Deep drawing: elastic and reference hardening parameters.

Description	Symbol	Value
Young's modulus	E	70 GPa
Poisson's ratio	ν	0.3
Yield stress	σ_0^{ref}	150 MPa
$\sigma_Y = \sigma_0(1 + k\bar{\epsilon}_p)^n$	k^{ref}	200
	n^{ref}	0.25

$\bar{\epsilon}_p$ is the equivalent plastic strain.
 σ_Y is the yield stress.

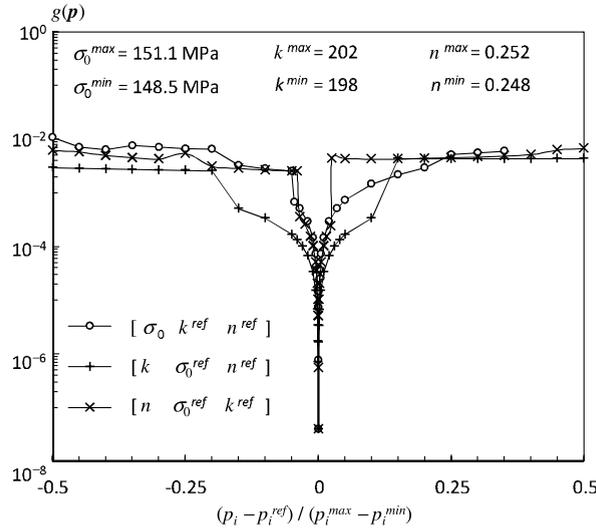


Fig. 4. Objective function in the vicinity of the minimum.

It is observed in the present deep drawing operation that small variations in the hardening parameters can cause changes in the load–displacement curve, which in turn leads to fitness variations, sometimes in unexpected ways. For instance, by computing the objective function for three sets of hardening parameters taken along a predefined search direction towards the minimum, (A) $\sigma_0 = 148.8$ MPa, (B) $\sigma_0 = 149.1$ MPa and (C) $\sigma_0 = 149.4$ MPa ($k = 200$ and $n = 0.25$), one obtains $g(\mathbf{p}_A) = 6.381 \times 10^{-3}$, $g(\mathbf{p}_B) = 7.066 \times 10^{-3}$ and $g(\mathbf{p}_C) = 6.519 \times 10^{-3}$. The non-monotonic convergence of $g(\mathbf{p})$ indicates an irregular topography of the objective function, which, in this example, clearly shows that sensitivity computation (in gradient-based methods) and simplex operations (for the Nelder–Mead technique) around point B would be compromised.

The existence of flat regions constitutes also a convergence hindrance for most optimization techniques. The present problem exhibits irregular flat surfaces in the space of design variables followed by a steep decrease close to the minimum. Fig. 4 illustrates this effect by assigning the reference values for two of the hardening parameters and performing an exhaustive search on the third one. The process is repeated for each design variable, shown in Fig. 4 for a parametric measure of the corresponding hardening parameter, $(p_i - p_i^{ref}) / (p_i^{max} - p_i^{min})$, in the vicinity of the optimal point. It can be observed that a significant decrease in the objective function takes place only for values very close to the minimum. A combination of such effects indeed imposes relevant difficulties to the gradient-based SQP method and the Nelder–Mead optimization technique, as discussed in Sections 4.2 and 4.3.

4.2. Parameter identification using the SQP algorithm

Gradient-based algorithms are designed for convex optimization and unquestionably yield the highest convergence rates for this class of problems. Nevertheless, the successful application of the SQP algorithm to elastoplastic and solid mechanics problems [6,7] combined to the robustness exhibited by the semi-analytical sensitivity analysis and enhanced finite difference scheme [10] have encouraged investigation on their use in the present deep drawing operation. In this test, 25 random initial estimates are used for the SQP method in order to evaluate the performance of the algorithm in the present scenario. No success was obtained for all trials. The SQP algorithm was also tested as the local stage of the hybrid scheme without success. Convergence could be achieved only for initial estimates very close to the reference point, which renders the scheme inefficient for the present identification problem. The literature rarely discusses failures; however, the authors have found relevant to report such unsuccessful attempt owing to the fact that the gradient-based Sequential Quadratic Programming

Table 4
Side constraints for the hardening parameters.

Parameter p_i	Lower limit p_i^{inf}	Upper limit p_i^{sup}
σ_0	100 MPa	200 MPa
k	100	300
n	0.10	0.4

Table 5
Initial and final parameters for runs 16 and 23.

Run		σ_0 [MPa]	k	n	$g(\mathbf{p})$	$ \mathbf{d} $
16	Initial	154.408889	242.924376	0.253838919	0.2639	0.219
	Final	148.209226	222.257589	0.246582765	3.921×10^{-3}	0.113
23	Initial	102.863597	157.339679	0.165459624	0.3804	0.589
	Final	150.000011	199.999959	0.250000009	1.259×10^{-8}	2.32×10^{-7}

algorithm has proved to be robust, being among the optimization methods for nonlinear applications that are the most referred to.

Remark (i): Despite the use of an enhanced semi-analytical sensitivity analysis approach and the small number of variables to be determined, the irregular topography of the fitness surface prevented the technique from computing the derivatives accurately and, consequently, from obtaining the minimum.

4.3. Parameter identification using the Nelder–Mead method

As in the preceding section, attempts to obtain the global minimum using the Nelder–Mead technique are performed for 25 random initial estimates. The NM control coefficients used in the simulations are as follows: reflection, $\rho = 1.0$, expansion, $\gamma = 2.0$, contraction, $\beta = 0.5$, and shrinkage, $\sigma = 0.5$. The initial simplex edge factor is $h = 0.2$. The lateral bounds (side constraints) used in the identification process are shown in Table 4. The identification process is successful when the objective function of the best vertex reaches $g(\mathbf{p}_1) \leq 1.7 \times 10^{-8}$, whilst final convergence is assumed when $\phi_{\text{NM}} \leq 1 \times 10^{-8}$. Based on the fact that the Nelder–Mead algorithm was originally conceived for convex optimization, its exclusive application to the present identification problem yielded a low rate of success, i.e. only one run was able to reach the global minimum.

It is interesting to mention that the Euclidean parametric distance,

$$|\mathbf{d}| = \sqrt{\sum_{i=1}^{n_d} \left[(p_i^{(0)} - p_i^{\text{ref}}) / (p_i^{\text{sup}} - p_i^{\text{inf}}) \right]^2} \quad (17)$$

between the reference parameters, \mathbf{p}^{ref} , and initial estimate, $\mathbf{p}^{(0)}$, of the successful run (Run 23) does not correspond to the closest initial estimate of all trials. Conversely, the closest initial estimate (Run 16) was not able to attain the global minimum (see Table 5). The results highlight the fact that the fitness landscape hinders the convergence path and gives a high degree of unpredictability to the NM method in the present case.

Remark (ii): Computation of the search direction by the Nelder–Mead scheme was compromised by the irregular topography of the fitness surface, thereby hampering convergence to the optimal point in most NM attempts.

4.4. Parameter identification using the PSO algorithm

This section presents an assessment of the application of PSO to the proposed deep drawing operation. The geometry of the problem and of the loading curve are presented in Figs. 3a and 3b, respectively. The lower and upper limits, p_i^{inf} and p_i^{sup} , used in the preceding examples are also adopted in the present identification process (see Table 4). Convergence is attained when $\phi_{\text{PSO}} \leq 1 \times 10^{-8}$, whereas the identification process is assumed successful when the objective function of the best particle reaches $g(\mathbf{p}_{\text{gb}}) \leq 1.7 \times 10^{-8}$ (the same criteria used for the NM algorithm). The success and the efficiency of the PSO algorithm strongly depend upon the proper choice of population size, inertia, cognitive, and social control weights. Based on previous studies on identification of inelastic parameters [37], the following ranges of the PSO parameters are investigated: population size, $n_p \geq 60$, inertia, $0.1 \leq w \leq 0.9$, and cognitive and social weights, $\varphi_1, \varphi_2 \in [0.5, 2.0]$.

4.4.1. Effects of the population size and randomness

The PSO method is stochastic in its essence owing to the initial random dispersion of the population and influence of $\mathbf{U}(0, \varphi_1)$ and $\mathbf{U}(0, \varphi_2)$ upon particle velocities (see task (iii) of Table 2). Two aspects are summarised in this section: the PSO performance for (i) population sizes $n_p = 60, 90, 120, 150, 180$ and 300, and (ii) five random populations for $n_p = 120$.

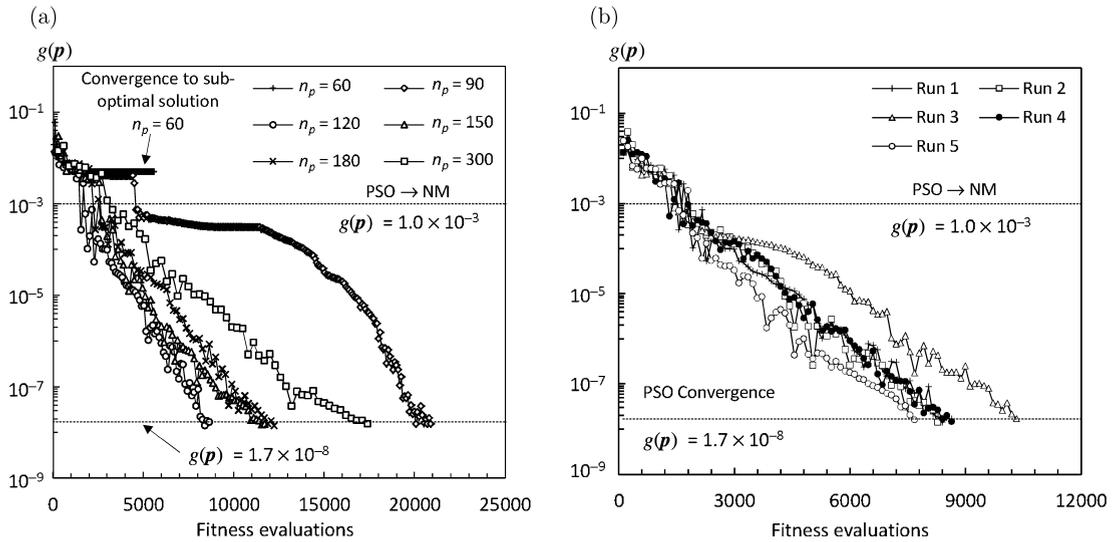


Fig. 5. (a) Effect of the number of particles, n_p , [20] and (b) randomness of the population $n_p = 120$ (PSO control weights: $w = 0.5$ and $\varphi_1 = \varphi_2 = 1.0$).

The ideal population size should combine robustness and efficiency. The former represents the ability of the algorithm to obtain the global minimum within a given tolerance on changes of its control parameters and randomness effects. In addition, efficiency of an algorithm is associated with the number of fitness computations required to achieve the global optimum. Contrasting to Genetic Algorithms, there are relatively few studies on the influence of the population size in mechanical problems. Most works approach the issue by assessing test functions, such as those by Shi and Eberhart [47] and by Coelho and Oliveira [48]. Aiming at identifying hardening parameters, Vaz Jr. et al. [37] reported successful results for population sizes $n_p = 120, 150, 200,$ and 250 .

Fig. 5(a) shows the evolution of the identification process for population sizes $n_p = 60, 90, 120, 150, 180,$ and 300 (the PSO control weights are $w = 0.5$ and $\varphi_1 = \varphi_2 = 1.0$) [20]. The characteristics of the fitness surface discussed in Section 4.1 rendered the procedure unsuccessful for smaller populations. Nevertheless, in spite of greater robustness, larger populations do not improve the convergence rate, i.e. the global minimum was obtained for equally larger numbers of fitness evaluations.

Despite the fact that randomness is implied when one defines different population sizes, it is worthy to investigate the evolution of the identification process for different populations of a given size. This example shows five different random initial populations of size $n_p = 120$. Fig. 5b highlights the stochastic character of the method by portraying markedly distinct evolution paths for each random population. Success was achieved in all trials, but for different numbers of evaluations of the objective function.

4.4.2. Effects of the inertia, cognitive and social weights

The success of the PSO algorithm is intrinsically associated with its control parameters. The inertia, cognitive and social weights are problem-dependent and must reflect an ideal balance between exploration and exploitation. Therefore, the recommended values for the PSO control weights can vary according to the application, oftentimes showing significant differences, as summarised by Sedighizadeh and Masehian [24]. Two scenarios are addressed in the present example: (i) the inertia effects, assuming $\varphi_1 = \varphi_2 = 1.0$ and (ii) cognitive and social effects for an inertia control weight $w = 0.5$. In both cases, a population size $n_p = 120$ is adopted.

The inertia parameter affects the search capacity of the algorithm, i.e. smaller values of w lead the PSO algorithm to premature convergence, whereas an excessively large inertia weight may cause the particles to frequently collide against the lateral bounds, thereby decreasing the convergence rate, or even precluding convergence itself. The simulations show that $w < 0.3$ causes the algorithm to converge to spurious minima and $w > 0.7$ renders the optimization procedure unstable (see Table 6). Noticeably, the results presented in Table 6 for $0.3 \leq w \leq 0.7$ show differences in the 7th–8th significant digit, which are associated with the convergence limit established for the examples. Therefore, a statistical treatment of the converged material parameters would not apply in this case.

The cognitive and social weights define the learning capacity of the algorithm to find the optimal point within the design space. Sedighizadeh and Masehian’s survey indicates that most applications adopt cognitive and social weights $\varphi_1 = \varphi_2 = 2.0$ associated with an inertia parameter such that $0.2 \leq w \leq 0.9$ [24]. However, the present identification problem has shown that $\varphi_1, \varphi_2 > 1.9$ causes the particles to move erratically around the minimum, demonstrating diminished local search capacity. On the other hand, it was observed that smaller values of φ_1 and φ_2 lead to spurious minima. Table 7 shows that successful convergence was achieved for $\varphi_1, \varphi_2 \in [0.7, 1.9]$ with best performance (small number of fitness computations) for $0.9 \leq \varphi_1, \varphi_2 \leq 1.1$.

Table 6Influence of the inertia weight for $\varphi_1 = \varphi_2 = 1.0$.

w	σ_0 [MPa]	k	n	Fitness evaluations	Convergence
0.1	149.993176	200.065446	0.249991092	5880	Spurious
0.3	149.999997	200.000043	0.250000000	6600	Global minimum
0.4	150.000007	200.000048	0.249999985	10560	
0.5	150.000005	199.999988	0.249999996	9120	
0.6	150.000025	199.999867	0.250000019	19680	
0.7	150.000020	199.999836	0.250000034	14760	
0.9	–	–	–	–	Unstable

Table 7Influence of the cognitive and social parameters for $w = 0.5$.

$\varphi_1 = \varphi_2$	σ_0 [MPa]	k	n	Fitness evaluations	Convergence
0.5	157.237632	144.154304	0.258825102	21600	Spurious
0.7	150.000015	199.999836	0.250000027	10200	Global minimum
0.9	150.000002	200.000014	0.250000011	7920	
1.0	150.000005	199.999988	0.249999996	9120	
1.1	150.000019	199.999925	0.249999996	7560	
1.3	150.000000	200.000066	0.249999991	10560	
1.5	150.000013	199.999992	0.249999996	13800	
1.7	150.000026	199.999821	0.250000024	18600	
1.9	150.000019	199.999916	0.249999996	32760	
2.0	–	–	–	–	Unstable

Remark (iii): In spite of the demanding character of the present deep drawing operation, the PSO method has proven to be robust, yielding a high success rate for the range of population sizes and control parameters investigated. The recommended set of PSO parameters for the present optimization process is as follows: the values $n_p = 120$, $w = 0.5$, and $\varphi_1 = \varphi_2 = 1.0$ are significantly different from those reported by Sedighizadeh and Masehian [24]. Therefore, despite the success of the PSO method, why should one pursue further investigation on hybrid schemes? The answer lies in the high computational cost exhibited by the PSO technique in the present application – the average number of fitness computations of the objective function using the recommended control parameters is 9300.

4.5. The PSO–NM hybrid identification strategy

In spite of the high success rate, the relatively high computational cost of the PSO method has motivated further investigation on *global–local* hybrid search approaches. This work combines the good PSO performance to approach the neighbourhood of the optimal point with the high convergence rate demonstrated by the Nelder–Mead technique. The hybrid strategy splits the identification problem into two stages: (A) the PSO technique is applied until a transition criterion is attained, followed by (B) application of the NM method using the best PSO particle as initial estimate.

The following aspects are relevant to the success of the Nelder–Mead stage: a proper choice of the PSO→NM transition criterion and geometry of the initial polytope. The choice of the transition criterion should balance efficiency and robustness, i.e. the earliest possible PSO→NM transition point able to ensure a high success rate of the NM stage. In the present work, transition is accomplished when the objective function of the best particle reaches $g(\mathbf{p}) \leq 10^{-3}$.

As discussed in Section 3.3, the NM initial estimate corresponds to the centroid of the initial simplex. This strategy requires a single parameter, the edge factor h , to define the initial size of the simplex. The simulations show that, for the second stage of the hybrid identification procedure, a range $0.01 \leq h \leq 0.05$ represents a compromise between efficiency and high success rate.

An investigation on the Nelder–Mead control parameters has indicated successful trials for the following intervals: reflection, $0.3 \leq \rho \leq 0.7$, expansion, $0.5 \leq \gamma \leq 10.0$, contraction, $0.3 \leq \beta \leq 0.9$, and shrinkage, $0.1 \leq \sigma \leq 0.9$. For the present application, it was not observed substantial variation in the number of fitness evaluations for the preceding ranges. Therefore, the following examples use the standard NM control parameters: $\rho = 1.0$, $\gamma = 2.0$, $\beta = 0.5$, and $\sigma = 0.5$.

The hybrid technique was applied to all successful runs reported in Section 4.4 following analysis of the (i) random initial populations, (ii) population size, (iii) inertia weight, and (iv) social and cognitive weights. The NM initial estimates of cases (i)–(iv) are presented in Table 8, which shows the hardening parameters of the best PSO particle and corresponding number of fitness evaluations required to achieve transition. The first column of Table 8 indicates the case number in order to facilitate further reference.

Fig. 6a presents the PSO and PSO–NM evolution for two limiting cases [20]. Case 11 ($n_p = 120$, $w = 0.3$, $\varphi_1 = \varphi_2 = 1.0$) and Case 14 ($n_p = 120$, $w = 0.6$, $\varphi_1 = \varphi_2 = 1.0$) provide the best and worst combined performance of the trials. It is noteworthy that the total number of fitness evaluations is mostly affected by the PSO stage of the identification process.

Table 8
Initial parameters for the second stage: PSO → NM transition ($g(\mathbf{p}_{gb}) \leq 10^{-3}$).

	Case	σ_0 [MPa]	k	n	Fitness evaluations		
1	(i)	Run ^(*)	1	150.471625	198.775415	0.249539890	1800
2			2	150.524674	199.514531	0.249270467	1440
3			3	150.559362	198.199834	0.249594225	1560
4			4	150.672116	195.420749	0.250248172	1320
5			5	150.410668	198.769248	0.249808870	1560
6	(ii)	n_p ^(†)	90	150.724907	193.570903	0.251043166	4590
7			120	150.471625	198.775415	0.249539890	1800
8			150	150.245809	199.883120	0.249568615	2700
9			180	150.224593	197.622301	0.250291768	2340
10			300	150.236905	200.499732	0.249588827	3300
11	(iii)	w ^(‡)	0.3	150.328936	198.615766	0.249772752	1200
12			0.4	149.719611	202.291381	0.249652357	1440
13			0.5	150.471625	198.775415	0.249539890	1800
14			0.6	150.565260	196.843793	0.250194050	5760
15			0.7	150.380380	197.412698	0.250255644	2280
16	(iv)	φ_1, φ_2 ^(§)	0.7	150.514433	198.224474	0.249886334	1440
17			0.9	150.382016	197.505476	0.250457272	1560
18			1.0	150.471625	198.775415	0.249539890	1800
19			1.1	149.947609	200.863582	0.249748910	2040
20			1.3	150.163744	199.551390	0.249892780	2040
21			1.5	150.461387	198.403398	0.249866949	2880
22			1.7	150.243440	198.038179	0.250227928	3360
23			1.9	150.306339	199.897177	0.249727207	5280

(*) Random populations: $n_p = 120$, $w = 0.5$ and $\varphi_1 = \varphi_2 = 1.0$.
 (†) Population size: $w = 0.5$ and $\varphi_1 = \varphi_2 = 1.0$.
 (‡) Inertia weight: $n_p = 120$ and $\varphi_1 = \varphi_2 = 1.0$.
 (§) Cognitive and social weights: $n_p = 120$ and $w = 0.5$.

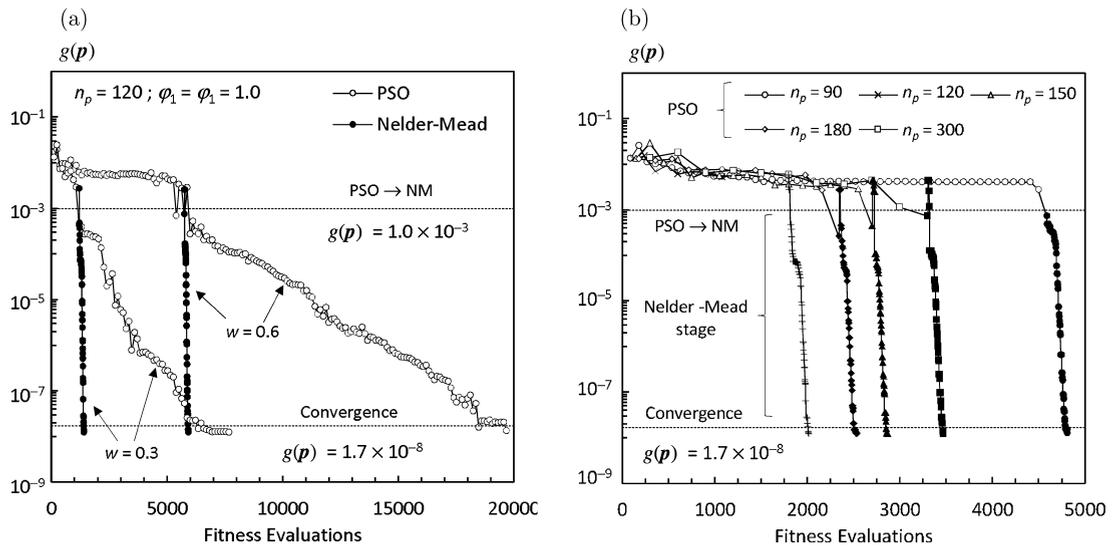


Fig. 6. (a) Full PSO and Nelder–Mead evolution for limiting cases: inertia parameters $w = 0.3$ and $w = 0.6$ (Cases 11 and 14) [20], and (b) combined PSO–Nelder–Mead evolution for different population sizes (Cases 6–10).

Fig. 6b shows the combined PSO–NM evolution for PSO original population sizes $n_p = 90, 120, 150, 180$ and 300 (see also Table 9). The Nelder–Mead initial simplex utilised in the present work causes the first NM steps to exhibit objective functions somewhat higher than the transitional PSO value (see curves for $n_p = 150, 180$ and 300 of Fig. 6b). Since the best PSO particle is located in the centroid of the NM polytope, the initial simplex construction algorithm may place its vertices slightly farther from the minimum. This small difference has imposed no significant effect on the NM convergence rate.

The final parameters and corresponding numbers of fitness evaluations for the individual and combined PSO–NM stages are presented in Table 9 for Cases 1 to 23. Convergence to the global minimum was obtained for all trials. As discussed previously, the differences in the converged parameters are also in the 7th–8th significant digits, which are associated with

Table 9
Final parameters after the Nelder–Mead stage.

Case	σ_0 [MPa]	k	n	NM	PSO	Total
1	150.000007	199.999988	0.249999997	186	1800	1986
2	150.000020	199.999862	0.250000021	148	1440	1588
3	150.000015	199.999899	0.250000008	193	1560	1753
4	150.000023	199.999844	0.250000034	165	1320	1485
5	150.000000	200.000069	0.249999988	184	1560	1744
6	150.000007	199.999961	0.250000014	222	4590	4812
7	150.000007	199.999988	0.249999997	186	1800	1986
8	150.000008	199.999989	0.250000003	168	2700	2868
9	150.000007	199.999992	0.250000002	194	2340	2534
10	150.000012	199.999988	0.250000001	170	3300	3470
11	150.000008	199.999987	0.250000000	217	1200	1417
12	150.000022	199.999899	0.250000007	218	1440	1658
13	150.000007	199.999988	0.249999997	186	1800	1986
14	150.000009	199.999989	0.250000002	172	5760	5932
15	150.000021	199.999901	0.250000012	193	2280	2473
16	150.000019	199.999905	0.250000010	200	1440	1640
17	150.000008	199.999985	0.249999999	170	1560	1730
18	150.000007	199.999988	0.249999997	186	1800	1986
19	150.000012	199.999992	0.249999996	197	2040	2237
20	149.999998	200.000069	0.249999991	184	2040	2224
21	150.000021	199.999904	0.250000009	152	2880	3032
22	150.000010	199.999962	0.250000008	169	3360	3529
23	150.000008	199.999960	0.250000009	179	5280	5459

the convergence limit established for the examples. A global assessment shows that the PSO stage required 93% of the total number of fitness computations. The average number of evaluations of the objective function for the PSO–NM hybrid scheme was approximately 2600.

Remark (iv): The simulations show that the PSO–NM hybrid scheme has increased the efficiency of the identification process without loss of robustness. However, it can always be argued the possibility to use independently the PSO and Nelder–Mead techniques. Nevertheless, as discussed in Section 4.3, the Nelder–Mead exhibited a low rate of success, whereas Section 4.4.2 shows that the exclusive application of the PSO has yielded a substantially high computational cost.

5. Concluding remarks

Identification of inelastic parameters based upon deep drawing process is addressed using the following well-established optimization strategies: a gradient-based Sequential Quadratic Programming (SQP), Nelder–Mead (NM) downhill simplex algorithm and Particle Swarm Optimization (PSO). The low performance exhibited by the aforementioned methods when used independently (SQP and Nelder–Mead algorithms were originally designed for convex optimization problems and the PSO method yielded a high computational cost) has encouraged investigation of global–local hybrid schemes.

Despite the robustness of the sensitivity analysis and SQP algorithm demonstrated in other demanding applications, all attempts to use the identification procedure have failed owing to the irregular fitness landscape. The Nelder–Mead technique has also rendered a low success rate due to similar effects.

Particle Swarm Optimization has shown high tolerance to changes of the control weights without compromising its high success rate for $n_p \geq 90$, $w \in [0.3, 0.7]$ and $\varphi_1, \varphi_2 \in [0.7, 1.9]$ (the recommended set is $n_p = 120$, $w = 0.5$ and $\varphi_1, \varphi_2 = 1.0$). However, the high computational cost demonstrated in the present case constitutes a clear disadvantage of the exclusive application of this algorithm. Nevertheless, in line with recent publications featuring other applications [41–44], the global–local PSO–NM hybrid strategy has demonstrated to be the most efficient approach by combining the PSO robust global search capacity with the NM high convergence rate (provided the initial estimate is sufficiently close to the minimum). The original NM control parameters, $\rho = 1$, $\gamma = 2$, $\beta = 0.5$, and $\sigma = 0.5$, and initial edge factor, $0.01 \leq h \leq 0.05$, were found appropriate for the second stage of the hybrid scheme.

Acknowledgements

The SQP source code used in this work was kindly made available by Prof. K. Schittkowski for academic purposes. The authors acknowledge the financial support provided by the Brazilian funding agency CNPq (National Council for Scientific and Technological Development) under grant Nos. 301601/2012-7 and 470327/2012-0.

References

- [1] E.A. de Souza Neto, D. Peric, D.R.J. Owen, *Computational Methods for Plasticity: Theory and Applications*, Wiley, Chichester, UK, 2008.
- [2] M. Springmann, M. Kuna, Identification of material parameters of the Gurson–Tvergaard–Needleman model by combined experimental and numerical techniques, *Comput. Mater. Sci.* 33 (4) (2005) 501–509.
- [3] T. Fraś, Z. Nowak, P. Perzyna, R.B. Peçherski, Identification of the model describing viscoplastic behaviour of high strength metals, *Inverse Probl. Sci. Eng.* 19 (1) (2011) 17–30.
- [4] P.A. Muñoz-Rojas, L.A.B. Cunda, E.L. Cardoso, M. Vaz Jr., G.J. Creus, A mixed optimization approach for parameter identification applied to the Gurson damage model, in: M. Vaz Jr., E.A. de Souza Neto, P.A. Muñoz-Rojas (Eds.), *Advanced Computational Materials Modeling: From Classical to Multi-Scale Techniques*, Wiley–VCH, Weinheim, Germany, 2011, pp. 165–204.
- [5] J.P. Ponthot, J.P. Kleinermann, A cascade optimization methodology for automatic parameter identification and shape/process optimization in metal forming simulation, *Comput. Methods Appl. Mech. Eng.* 195 (41–43) (2006) 5472–5508.
- [6] A. Bilotta, L. Leonetti, G. Garcea, An algorithm for incremental elastoplastic analysis using equality constrained sequential quadratic programming, *Comput. Struct.* 102–103 (2012) 97–107.
- [7] T.Y. Chen, J.H. Huang, An efficient and practical approach to obtain a better optimum solution for structural optimization, *Eng. Optim.* 45 (8) (2013) 1005–1026.
- [8] K. Schittkowski, NLPQLP: a new Fortran implementation of a sequential quadratic programming algorithm for parallel computing, Technical report, University of Bayreuth, Bayreuth, Germany, 2001.
- [9] S.S. Rao, *Engineering Optimization. Theory and Practice*, Wiley, Hoboken, NJ, USA, 2009.
- [10] P.A. Muñoz-Rojas, J.S.O. Fonseca, G.J. Creus, A modified finite difference sensitivity analysis method allowing remeshing in large strain path-dependent problems, *Int. J. Numer. Methods Eng.* 61 (7) (2004) 1049–1071.
- [11] J.P. Kleinermann, *Identification paramétrique et optimisation des procédés de mise à forme par problèmes inverses*, Ph.D. thesis, University of Liège, Liège, Belgium, 2000.
- [12] J.A. Nelder, R. Mead, A simplex method for function minimization, *Comput. J.* 7 (4) (1965) 308–313.
- [13] W. Spendley, G.R. Hext, F.R. Himsforth, Sequential application of simplex designs in optimisation and evolutionary operation, *Technometrics* 4 (4) (1962) 441–461.
- [14] J.C. Lagarias, J.A. Reeds, M.H. Wright, P.E. Wright, Convergence properties of the Nelder–Mead simplex method in low dimensions, *SIAM J. Optim.* 9 (1) (1998) 112–147.
- [15] D. Banabic, T. Kuwabara, T. Balan, D.S. Comsa, D. Julean, Non-quadratic yield criterion for orthotropic sheet metals under plane-stress conditions, *Int. J. Mech. Sci.* 45 (4) (2003) 797–811.
- [16] Y. Pannier, S. Avril, R. Rotinat, F. Pierron, Identification of elasto-plastic constitutive parameters from statically undetermined tests using the virtual fields method, *Exp. Mech.* 46 (6) (2006) 735–755.
- [17] J. Shanpo, Z. Youqing, Z. Chensong, Numerical solution to identification problems of material parameters in geotechnical engineering, *Proc. Eng.* 28 (2012) 61–65.
- [18] J. Helfenstein, *Continuum mechanical investigations of the intervertebral disc*, Ph.D. thesis, Diss. ETH Nr. 19545, ETH Zurich, Zurich, Switzerland, 2011.
- [19] J.J. Tomick, *On convergence of the Nelder–Mead simplex algorithm for unconstrained stochastic optimization*, Ph.D. thesis, The Pennsylvania State University, University Park, PA, USA, 1995.
- [20] M. Vaz Jr., E.L. Cardoso, P.A. Muñoz-Rojas, T.A. Carniel, M.A. Luersen, M. Tomiyama, J.O. da Silva, J. Stahlschmidt, R.G. Trentin, Identification of constitutive parameters – optimization strategies and applications, *Mat.-wiss. Werkstofftech.* 46 (4–5) (2015) 477–491.
- [21] M.A. Luersen, R. Le Riche, Globalized Nelder–Mead method for engineering optimization, *Comput. Struct.* 82 (23–26) (2004) 2251–2260.
- [22] R. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, in: *Proceedings of the 6th International Symposium on Micro Machine and Human Science*, IEEE Press, Piscataway, NJ, USA, 1995, pp. 39–43.
- [23] J. Kennedy, R. Eberhart, Particle swarm optimization, in: *Proceedings of the IEEE International Conference on Neural Networks*, IEEE Press, Piscataway, NJ, USA, 1995, pp. 1942–1948.
- [24] D. Sedighzadeh, E. Masehian, Particle swarm optimization methods, taxonomy and applications, *Int. J. Comput. Theory Eng.* 1 (5) (2009) 486–502.
- [25] J.F. Schutte, A.A. Groenwold, A study of global optimization using particle swarms, *J. Glob. Optim.* 31 (1) (2005) 93–108.
- [26] C. Blum, X. Li, *Swarm intelligence in optimization*, in: C. Blum, D. Merkle (Eds.), *Swarm Intelligence – Introduction and Applications*, Springer, Heidelberg, Germany, 2008, pp. 43–85.
- [27] N. Tian, J. Sun, W. Xu, C.H. Lai, Quantum-behaved particle swarm optimization with ring topology and its application in estimating temperature-dependent thermal conductivity, *Numer. Heat Transf., Part B, Fundam.* 60 (2) (2011) 73–95.
- [28] F.B. Liu, Particle swarm optimization-based algorithms for solving inverse heat conduction problems of estimating surface heat flux, *Int. J. Heat Mass Transf.* 55 (7–8) (2012) 2062–2068.
- [29] S. Vakili, M.S. Gadala, Low cost surrogate model based evolutionary optimization solvers for inverse heat conduction problem, *Int. J. Heat Mass Transf.* 56 (1–2) (2013) 263–273.
- [30] A. Kaveh, S. Talatahari, Hybrid charged system search and particle swarm optimization for engineering design problems, *Eng. Comput.* 28 (4) (2011) 423–440.
- [31] B. Nanda, D. Maity, D.K. Maity, Modal parameter based inverse approach for structural joint damage assessment using unified particle swarm optimization, *Appl. Math. Comput.* 242 (2014) 407–422.
- [32] H. Li, K. Chandrashekhara, Particle swarm-based structural optimization of laminated composite hydrokinetic turbine blades, *Eng. Optim.* 47 (9) (2015) 1191–1207.
- [33] X.T. Feng, B.R. Chen, C. Yang, H. Zhou, X. Ding, Identification of visco-elastic models for rocks using genetic programming coupled with the modified particle swarm optimization algorithm, *Int. J. Rock Mech. Min. Sci.* 43 (5) (2006) 789–801.
- [34] T.A. Carniel, P.A. Muñoz-Rojas, M. Vaz Jr., A viscoelastic viscoplastic constitutive model including mechanical degradation: uniaxial transient finite element formulation at finite strains and application to space truss structures, *Appl. Math. Model.* 39 (5–6) (2015) 1725–1739.
- [35] K.H. Hornig, G.T. Flowers, Performance of heuristic optimisation methods in the characterisation of the dynamic properties of sandwich composite materials, *Int. J. Acoust. Vib.* 12 (1) (2007) 60–68.
- [36] A. Fereidoon, F. Sadri, H. Hemmatian, Functionally graded materials optimization using particle-swarm based algorithms, *J. Therm. Stresses* 35 (4) (2012) 377–392.
- [37] M. Vaz Jr., E.L. Cardoso, J. Stahlschmidt, Particle swarm optimization and identification of inelastic material parameters, *Eng. Comput.* 30 (7) (2013) 936–960.
- [38] B.M. Chaparro, S. Thuillier, L.F. Menezes, P.Y. Manach, J.V. Fernandes, Material parameters identification: gradient-based, genetic and hybrid optimization algorithms, *Comput. Mater. Sci.* 44 (2) (2008) 339–346.
- [39] P.A. Muñoz-Rojas, E.L. Cardoso, M. Vaz Jr., Parameter identification of damage models using genetic algorithms, *Exp. Mech.* 50 (5) (2010) 627–634.
- [40] R. De-Carvalho, R.A.F. Valente, A. Andrade-Campos, Optimization strategies for non-linear material parameters identification in metal forming problems, *Comput. Struct.* 89 (1–2) (2011) 246–255.

- [41] P. Kumar, A.K. Singh, A Nelder–Mead PSO based approach to optimal capacitor placement in radial distribution system, in: B.K. Panigrahi, P.N. Suganthan, S. Das, S.C. Satapathy (Eds.), *Swarm, Evolutionary, and Memetic Computing*, in: *Lecture Notes in Computer Science*, vol. 7076, Springer-Verlag, Heidelberg, Germany, 2011, pp. 143–150.
- [42] A.H. Kayhan, H. Ceylan, M.T. Ayvaz, G. Gurarslan, PSOLVER: a new hybrid particle swarm optimization algorithm for solving continuous optimization problems, *Expert Syst. Appl.* 37 (10) (2010) 6798–6808.
- [43] F. Barzinpour, R. Noorossana, S.T.A. Niaki, M.J. Ershadi, A hybrid Nelder–Mead simplex and PSO approach on economic and economic-statistical designs of MEWMA control charts, *Int. J. Adv. Manuf. Technol.* 65 (9–12) (2013) 1339–1348.
- [44] M. Vaz Jr., P.A. Muñoz-Rojas, E.L. Cardoso, M. Tomiyama, Considerations on parameter identification and material response for Gurson-type and Lemaitre-type constitutive models, *Int. J. Mech. Sci.* 106 (2016) 254–265.
- [45] M. Vaz Jr., M.A. Luersen, P.A. Muñoz-Rojas, E. Bertoti, R.G. Trentin, A benchmark study on identification of inelastic parameters based on deep drawing processes using PSO Nelder Mead hybrid approach, in: E. Oñate, D.R.J. Owen, D. Peric, B. Suárez (Eds.), *Computational Plasticity XII – Fundamentals and Applications*, CIMNE, Barcelona, Spain, 2013, pp. 153–163.
- [46] V. Boljanovic, *Sheet Metal Forming Process and Die Design*, Industrial Press, New York, NY, USA, 2004.
- [47] Y. Shi, C. Eberhart, Empirical study of particle swarm optimization, in: *Proceedings of the IEEE Congress on Evolutionary Computation*, IEEE Press, Piscataway, NJ, USA, 1999, pp. 1945–1950.
- [48] A.L.V. Coelho, D.G. de Oliveira, Dynamically tuning the population size in particle swarm optimization, in: *Proceedings of the ACM Symposium on Applied Computing*, ACM Press, New York, NY, USA, 2008, pp. 1782–1787.