Research Article

A New Iterative Numerical Continuation Technique for Approximating the Solutions of Scalar Nonlinear Equations

Grégory Antoni

IFSTTAR, Aix-Marseille Université, LBA UMR T24, 13016 Marseille, France

Correspondence should be addressed to Grégory Antoni; antoni.gregory@yahoo.fr

Received 30 June 2016; Accepted 24 October 2016; Published 16 January 2017

Academic Editor: Yurong Liu

Copyright © 2017 Grégory Antoni. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The present study concerns the development of a new iterative method applied to a numerical continuation procedure for parameterized scalar nonlinear equations. Combining both a modified Newton’s technique and a stationary-type numerical procedure, the proposed method is able to provide suitable approximate solutions associated with scalar nonlinear equations. A numerical analysis of predictive capabilities of this new iterative algorithm is addressed, assessed, and discussed on some specific examples.

1. Introduction

The resolution of scalar nonlinear equations is an issue frequently encountered in many branches of physical sciences such as mechanics [1–5]. Although, in the literature, the most used numerical methods are either the classical Newton’s technique [3, 4, 6] or modified Newton-type procedures [7–10], they suffer from the main disadvantage of being held in check in the presence of critical points [11]. In order to overcome this deficiency, we propose to develop a new iterative algorithm applied to a numerical continuation procedure [5] for providing the approximate solutions associated with parameterized scalar nonlinear equations. The presented algorithm is based on a modified Newton-type method coupled with a stationary numerical technique. This study is organized in the following manner: (i) in Section 2, the standard numerical continuation procedure is briefly recalled including some classical algorithms; (ii) in Section 3, the new proposed iterative numerical method is presented in detail; (iii) in Section 4, the predictive abilities associated with this new iterative algorithm are tested and evaluated on some examples.

2. Standard Numerical Continuation Methods

2.1. Problem Statement. We consider the parameterized scalar nonlinear equation \( \mathcal{E} : (X, L) \in \mathbb{R}^2 \to \mathbb{R} \) in the following form:

\[
\mathcal{E}(X, L) = 0,
\]

where \( X \) denotes the real-valued "solution" variable associated with the nonlinear problem under consideration and \( L \) is the real-valued scalar "parameter" variable. It is important to emphasize the following: (i) the parametrized scalar nonlinear equation \( \mathcal{E} \) (see (1)) may include critical points \((U_{cr}, \lambda_{cr})\) (see Figure 1); (ii) the couple \((X, L)\) can depend on another parameter \( p \in \mathbb{R} \) such as (1) reading \( \mathcal{E}(X(p), L(p)) = 0 \); (iii) in the framework of solid mechanics, (1) represents the mechanical equilibrium equation and the "solution" variable \( X \) and the scalar "parameter" variable \( L \) denote the displacement and the mechanical load, respectively (i.e., \( X \equiv U \) and \( L \equiv \lambda \)). Within this context, the natural parameter \( p \) is the physical time \( t \); that is, (1) can be written as follows:

\[
\mathcal{E}(U(t), \lambda(t)) = 0.
\]
\( E(U, \lambda) = 0 \) which presents only one critical point \((U_{cr}, \lambda_{cr})\) in four different situations ((a), (b), (c), and (d)).

2.2. Some Commonly Used Algorithms

2.2.1. Classical Newton and Newton-Type Procedures. By placing in the context of solid mechanics and considering that the physical time \( t = [0, T] \) (with \( T \in \mathbb{R}^+ \)) is divided into \((N + 1)\)-subintervals \([t_n, t_{n+1}]\) (i.e., \( T = (N + 1)\Delta t \) with \( \Delta t = \Delta t_{n+1} = t_{n+1} - t_n \) and \( t_{n+1} = (n + 1)\Delta t, \forall n = 0, \ldots, N \) with \( N \in \mathbb{N} \)), we have the displacement \( U(t) \) and the mechanical load \( \lambda(t) \) at the incremental time \( t_n \) (resp., \( t_{n+1} \)) that are written as follows: \( U(t_n) \equiv U_n \) and \( \lambda(t_n) \equiv \lambda_n \) (resp., \( U(t_{n+1}) \equiv U_{n+1} \) and \( \lambda(t_{n+1}) \equiv \lambda_{n+1} \)).

For solving numerically (2), we consider the discrete-time interval \([t_n, t_{n+1}]\) and we perform a Taylor series expansion of the function \( E \) (representing here the mechanical equilibrium of solid) in the first order at point \((U^{k+1}_{n+1}, \lambda_{n+1})\) (with \( \lambda_{n+1} \) being fixed and constant):

\[
E(U^{k+1}_{n+1}, \lambda_{n+1}) = E(U^k_{n+1}, \lambda_{n+1}) + \left[ \frac{\partial E}{\partial U}(U^k_{n+1}, \lambda_{n+1}) \right] (U^{k+1}_{n+1} - U^k_{n+1}) + o \left( \left\| (U^{k+1}_{n+1} - U^k_{n+1}) \right\| \right) = 0 \quad \forall k = 0, \ldots, K \forall n = 0, \ldots, N,
\]
Figure 2: Schematic diagram of some numerical continuation algorithms: (a) classical Newton’s; (b) modified Newton-type; (c) arc-length; (d) pseudo-arc-length.

where \( o(\|U_{n+1}^{k+1} - U_{n+1}^k\|) \) denotes the higher-order terms with the Landau notation \( o(\cdot) \) associated with the asymptotic behavior of the function \( \mathcal{E} \) (considering only the variable quantity \( U \)), \( \| \cdot \| \) is Euclidean norm associated with the quantity \( \cdot \) (here, Euclidean distance reduces to the absolute value \( | \cdot | \) since that there is only one-variable \( U \), i.e., \( \|U_{n+1}^{k+1} - U_{n+1}^k\| = |U_{n+1}^{k+1} - U_{n+1}^k| \)) and \( U \) \( (k+1) \)th iterative solution associated with the variable \( \lambda \) at the incremental time \( t_{n+1} \) (with \( K \in \mathbb{N} \)), and \( \frac{\partial \mathcal{E}}{\partial U}(U_{n+1}^k; \lambda_{n+1}) \equiv \frac{\partial \mathcal{E}}{\partial U}(U_{n+1}^k, \lambda)|_{\lambda=\lambda_{n+1}} \) is the first-order partial derivative operator associated with the function \( \mathcal{E} \) with respect to \( U \) (at point \( U_{n+1}^k \) which is \( k \)th iterative displacement of the incremental time \( t_{n+1} \)) for fixed and constant mechanical load \( \lambda \) (at point \( \lambda_{n+1} \) which is the value of parameter of the incremental time \( t_{n+1} \)). It should be underscored that the variable \( \lambda \) (resp., \( \lambda_{n+1} \)) without the exponent \( k \) or \( (k + 1) \) represents a converged (resp., known) quantity at the incremental time \( t_n \) (resp., \( t_{n+1} \)).

In line with (3), we can define the following:

(i) classical Newton’s algorithm (see case (a) of Figure 2; cf. [3, 4, 7]):

\[
U_{n+1}^{k+1} = U_{n+1}^k - \frac{\mathcal{E}(U_{n+1}^k, \lambda_{n+1})}{\frac{\partial \mathcal{E}}{\partial U}(U_{n+1}^k; \lambda_{n+1})} \quad \forall k = 0, \ldots, K \quad \forall n = 0, \ldots, N;
\]
ii) Newton-type algorithm (see case (b) of Figure 2; cf. [3, 4, 7]):

\[ U_{n+1}^{k+1} = U_n^{k+1} - \frac{\partial \mathcal{E}(U_n^{k+1}, \lambda_n^{k+1})}{\partial U} \]

\[ \forall k \neq 0 \forall k = 0, \ldots, K \forall n = 0, \ldots, N, \]

where \( \kappa \in \mathbb{R}^* \) denotes a coefficient which checks \( \kappa = \text{sgn}(\partial \mathcal{E}/\partial U)(U_n^{k+1}; \lambda_n^{k+1}) \) with \( |\partial \mathcal{E}/\partial U|(U_n^{k+1}; \lambda_n^{k+1}) \leq \theta < +\infty \), \( \text{sgn}(\cdot) \) is the sign function (such as \( \text{sgn}(\varphi) = -1 \) when \( \varphi < 0 \), \( \text{sgn}(\varphi) = 1 \) when \( \varphi > 0 \), and \( \text{sgn}(\varphi) = 0 \) when \( \varphi = 0 \)), and \( |\varphi| \) is the absolute-value function (such as \( |\varphi| = -\varphi \) when \( \varphi < 0 \) and \( |\varphi| = \varphi \) when \( \varphi \geq 0 \)). It is worth noting that if \( \kappa = (\partial \mathcal{E}/\partial U)(U_n^{k+1}; \lambda_n^{k+1}) \) then this is the classical Newton’s algorithm and if \( \kappa = C \) (with \( C \) being the constant coefficient which does not depend on the point \((U_n^{k+1}; \lambda_n^{k+1})\) and checking \( C = \text{sgn}(\partial \mathcal{E}/\partial U)(U_n^{k+1}; \lambda_n^{k+1}) \)) then this is the modified Newton’s algorithm.

2.2.2. Standard Arclength and Pseudo-Arclength Procedures. Similar to previous approach (see Section 2.2.1), we perform a Taylor series expansion of the function \( \mathcal{E} \) (representing the mechanical equilibrium of solid) in the first order at point \((U_n^{k+1}, \lambda_n^{k+1})\):

\[ \mathcal{E}(U_n^{k+1}, \lambda_n^{k+1}) \]

\[ = \mathcal{E}(U_n^{k+1}, \lambda_n^{k+1}) + \left[ \frac{\partial \mathcal{E}}{\partial U}(U_n^{k+1}; \lambda_n^{k+1}) \right] (U_{n+1}^{k+1} - U_n^{k+1}) + \left[ \frac{\partial \mathcal{E}}{\partial \lambda}(\lambda_n^{k+1}; U_n^{k+1}) \right] (\lambda_{n+1}^{k+1} - \lambda_n^{k+1}) + o \left( \left\| (U_{n+1}^{k+1} - U_n^{k+1}), (\lambda_{n+1}^{k+1} - \lambda_n^{k+1}) \right\| \right) = 0 \]

\[ \forall k = 0, \ldots, K \forall n = 0, \ldots, N, \]

where \( o(\left\| (U_{n+1}^{k+1} - U_n^{k+1}), (\lambda_{n+1}^{k+1} - \lambda_n^{k+1}) \right\|) \) denotes the higher-order terms with the Landau notation \( o(\cdot) \) associated with the asymptotic behaviour of the function \( \mathcal{E} \) (considering the variable quantities \((U, \lambda)\) and \( \partial \mathcal{E}/\partial \lambda)(\lambda_n^{k+1}; U_n^{k+1}) = (\partial \mathcal{E}/\partial \lambda)(\lambda_n^{k+1}) \) (resp., \( \partial \mathcal{E}/\partial U)(U_n^{k+1}; \lambda_n^{k+1}) = (\partial \mathcal{E}/\partial U)(U_n^{k+1}; \lambda_n^{k+1}) \)) is the first-order partial derivative operator associated with the function \( \mathcal{E} \) with respect to \( \lambda \) (resp., \( U \)) at point \( \lambda_n^{k+1} \) (resp., \( U_n^{k+1} \)) for fixed and constant displacement \( U \) (resp., mechanical load \( \lambda \)) at point \( U_n^{k+1} \) (resp., \( \lambda_n^{k+1} \)) which is \( k \)th iterative variable of incremental time \( t_n^{k+1} \).

Combined with (5), we introduce another scalar equation \( \mathcal{E} = 0 \) (so-called "constraint condition"); that is;

(i) the equation \( \mathcal{E} \) is nonlinear type (second order) in the case of an arclength algorithm (see case (c) of Figure 2) (e.g., Crisfield procedure [5, 12, 13]);

(ii) the equation \( \mathcal{E} \) is linear type in the case of a pseudo-arclength algorithm (see case (d) of Figure 2) (e.g., Riks [14–17], Ramm [18, 19], or Wempner [20] procedures).

It may be stressed that there exist many other methods used for numerical continuation procedures; one of them, which is not present here, is called “normal flow algorithm” or “Davidenko’s flow algorithm” (see [21, 22] for more details); the mechanical equilibrium equation of solid \( \mathcal{E} \) associated with the Davidenko’s flow reads \( \mathcal{E}(U, \lambda) = \chi \), where \( \chi \) denotes the perturbation parameter.

3. A New Iterative Numerical Continuation Method

3.1. Proposed Algorithm. In this section, we present a new iterative numerical continuation procedure for approximating the solutions associated with any parameterized scalar nonlinear equations. The proposed iterative algorithm belongs to the family of predictor-corrector methods, and it uses both a modified Newton’s method and a stationary-type numerical technique. The stationary procedure allows reducing the considered scalar nonlinear equation (see (I)) to only one explicit equation such as

\[ \mathcal{E}(\bullet; \bullet) \equiv \mathcal{E}(\bullet, \bullet) \ast, \]

where \( \ast \) is the considered variable and \( \ast \) is the fixed and constant parameter. It should be stressed that the derivative of the function \( \mathcal{E} \) (see (I)) checks that

\[ D\mathcal{E}(X, L) = D\mathcal{E}(X; L) + D\mathcal{E}(L; X), \]

where \( D \) denotes the first-order total derivative operator (with \( (D\mathcal{E}/D\lambda)(\bullet; \bullet) \equiv (D\mathcal{E}/D\lambda)(\bullet; \bullet) \))

When considering both a discrete-time interval \([t_n, t_{n+1}]\) and an orthonormal basis \([\vec{e}_x, \vec{e}_y]\), the direction vector \( \vec{T}_n \) associated with the tangent straight line \( \mathcal{T}(U; \lambda_n) \) at point \((U_n, \lambda_n)\) can be written as follows (with the stationary procedure (7)):

\[ \vec{T}_n = \vec{T}_n^{\perp} \vec{e}_x + \vec{T}_n^{\parallel} \vec{e}_y \]

with

\[ \begin{bmatrix} \vec{T}_n^{\perp} \\ \vec{T}_n^{\parallel} \end{bmatrix} = \begin{bmatrix} 1 \\ \frac{\partial \mathcal{E}}{\partial U}(U_n; \lambda_n) \end{bmatrix} \vec{e}_x \]

\[ \forall \frac{\partial \mathcal{E}}{\partial U}(U_n; \lambda_n) \neq 0, \]

\[ \mathcal{T}(U; \lambda_n) = \frac{\partial \mathcal{E}}{\partial U}(U_n; \lambda_n) (U - U_n) + \mathcal{E}(U_n; \lambda_n) \]

\[ \forall n = 0, \ldots, N, \]

where \( \vec{e}_x \) denotes the unit vector of the basis such as \( \|\vec{e}_x\| = 1 \) \((\forall i = U, \lambda, \text{ and } \vec{T}_n^{\parallel} \text{ are the components associated with the vector } \vec{T}_n \text{ in the orthonormal basis } [\vec{e}_x, \vec{e}_y]) \). It should be noted that, in (II), the term \( \mathcal{E}(U_n, \lambda_n) = \mathcal{E}(U_n, \lambda_n) = 0 \) since that \((U_n, \lambda_n)\) is the mechanical equilibrium point at the time \( t_n \).
For crossing more easily some critical points \((U_{cr}, \lambda_{cr})\) associated with the nonlinear function \(\mathcal{E}\), we introduce a new director vector \(\vec{H}_n\) associated with the straight line \(\mathcal{F}(U; \lambda_n)\) at point \((U_n, \lambda_n)\) at the time \(t_n\), such as

\[
\vec{H}_n = \frac{H_n}{H_n U} \hat{e}_U + \frac{1}{H_n \lambda} \hat{e}_\lambda
\]

with

\[
\begin{pmatrix}
H_n^U \\
H_n^\lambda
\end{pmatrix} = \begin{pmatrix}
\frac{\partial \mathcal{E}}{\partial U}(U_n; \lambda_n) + \frac{\beta}{(\partial \mathcal{E}/\partial U)(U_n; \lambda_n)} \hat{e}_U \hat{e}_\lambda \\
\end{pmatrix}
\]

where \(H_n\) (with \(i = U, \lambda\)) are the components associated with the vector \(\vec{H}_n\) in the orthonormal basis \(\{\hat{e}_U, \hat{e}_\lambda\}\) and \(\alpha\) and \(\beta\) are two parameters \((\alpha, \beta \in \mathbb{R})\).

Using (12) and (13), we define the equation of the straight line \(\mathcal{F}(U; \lambda_{n+1})\) passing through the point \((U_n, \lambda_{n+1})\) and with the director vector \(\vec{H}_n\) that must satisfy the following relation (see Figure 3):

\[
\mathcal{F}(U; \lambda_{n+1}) = \Theta \left[ U - U_{k+1} \right] + \mathcal{E}(U_{k+1}^n; \lambda_{n+1})
\]
For the first iterative step (i.e., $k = 1$), the iterative point $U_{n+1}^1$ must satisfy the following relation (with $\phi_{n+1} = \phi_n$, $\forall \phi = U, \lambda$):

$$
(U_{n+1}^1 - U_n)^2 + (\mathcal{H}(U_{n+1}^1; \lambda_n) - \mathcal{E}(U_n; \lambda_n))^2 = \gamma^2 \iff (U_{n+1}^1 - U_n)^2 \left[1 + \left(\frac{\alpha \frac{\partial \mathcal{E}}{\partial U}(U_n; \lambda_n) + \frac{\beta}{(\partial \mathcal{E}/\partial U)(U_n; \lambda_n)}}{\sqrt{1 + (\Theta_1)^2}}\right)^2\right] = \gamma^2 \quad \forall n = 0, \ldots, N,
$$

where $\gamma$ is a parameter ($\gamma \in \mathbb{R}^*_+)$.

In line with (16), the iterative point $U_{n+1}^1$ checks (see Figure 3)

$$
U_{n+1}^1 = U_n \pm \frac{\gamma}{\sqrt{1 + (\Theta_1)^2}} \quad (17)
$$

with

$$
\Theta_1 = \alpha \frac{\partial \mathcal{E}}{\partial U}(U_n; \lambda_n) + \frac{\beta}{(\partial \mathcal{E}/\partial U)(U_n; \lambda_n)} \quad (18)
$$

$\forall \partial \mathcal{E}/\partial U(U_n; \lambda_n) \neq 0 \forall n = 0, \ldots, N$.

Using (6), (7) and (8), we have (with $\mathcal{E}(U_{n+1}^1; \lambda_{n+1}) = 0$ and $\mathcal{E}(U_n; \lambda_n) = \mathcal{E}(\lambda_n; U_n) = \mathcal{E}(U_n; \lambda_n)$):

$$
\mathcal{E}(U_n; \lambda_n) + \left[\frac{\partial \mathcal{E}}{\partial U}(U_n; \lambda_n)\right]\left(U_{n+1}^1 - U_n\right) + \left[\frac{\partial \mathcal{E}}{\partial \lambda} (\lambda_n, U_n)\right]\left(\lambda_{n+1}^1 - \lambda_n\right) = 0 \quad (19)
$$
In line with (19) and considering (7) combined with the
updated iterative point $U_{n+1}^1$ obtained with (17) and (18), we
can approximate the new iterative point $\lambda_{n+1}$ as

$$
\lambda_{n+1} = \lambda_n - \frac{1}{(\partial E/\partial \lambda)(\lambda_n;U_{n+1}^1)} \left[ E(\lambda_n;U_{n+1}^1) \right.
+ \frac{1}{(\partial E/\partial \lambda)(\lambda_n;U_{n+1}^1)} (U_{n+1}^1 - U_n) \left. \right] \quad \forall \frac{1}{(\partial E/\partial \lambda)(\lambda_n;U_{n+1}^1)} \neq 0 \forall n = 0, \ldots, N.
$$

(20)

For the other iterative steps (i.e., $k > 1$), we introduce the straight line $\mathcal{W}(U;\lambda_i)$ passing through the point $(U_{n+1}^1, \mathcal{H}(U_{n+1}^1;\lambda_i))$ and with the director vector $\vec{W}_n$ must satisfy the following relations (see Figure 3):

$$
\vec{W}_n = \overrightarrow{W_n^U} \hat{e}_U + \overrightarrow{W_n^{\lambda}} \hat{e}_\lambda
$$

(21)

with

$$
\mathcal{W}(U;\lambda) = -\frac{1}{\Theta_1} (U - U_{n+1}^1) + \mathcal{H}(U_{n+1}^1;\lambda),
$$

(22)

where $\overrightarrow{W_n^i}$ (with $i = U, \lambda$) are the components associated with the vector $\vec{W}_n$ in the orthonormal basis $\{\hat{e}_U, \hat{e}_\lambda\}$.

**Figure 5:** Evolution of the approximate solutions $(U_{n+1}^k, \lambda_{n+1}^k)$ obtained with the new iterative numerical continuation algorithm (blue solid line with circles; see (26)–(30) with $\Theta = \Theta_1$) and compared with the exact solution (red solid line) for Example 2 ($E_2$) with the following parameters: $\alpha = 1000$, $\beta = 500$, and $\gamma = 2$. 
Figure 6: Evolution of the approximate solutions \((U_{n+1}^k, \lambda_{n+1}^k)\) obtained with the new iterative numerical continuation algorithm (blue solid line with circles; see (26)–(30) with \(\Theta = \Theta_1\)) and compared with the exact solution (red solid line) for Example 3 \((E_3)\) with the following parameters: \(\alpha = 10000\), \(\beta = 200\), and \(\gamma = 20\).

Figure 7: Evolution of the approximate solutions \((U_{n+1}^k, \lambda_{n+1}^k)\) obtained with the new iterative numerical continuation algorithm (see (26)–(30) with \(\Theta = \Theta_1\)) and compared with the exact solution (black solid line) for Example 1 \((E_1)\) with the following parameters: (i) \(\alpha = 50\), \(\beta = -100\), and \(\gamma = 1\) (blue solid line with circles); (ii) \(\alpha = 3000\), \(\beta = -1\), and \(\gamma = 1\) (green solid line with squares); (iii) \(\alpha = 30\), \(\beta = -200\), and \(\gamma = 1\) (magenta solid line with diamonds); (iv) \(\alpha = 80\), \(\beta = -600\), and \(\gamma = 1\) (red solid line with upward-pointing triangles); (v) \(\alpha = 100\), \(\beta = -8\), and \(\gamma = 1\) (cyan solid line with downward-pointing triangles).
Figure 8: Evolution of the residue error of the function $\mathcal{E}(U_{k+1}^{n+1}, \lambda_{k+1}^{n+1})$, the approximation error of displacement $U(U_{k+1}^{n+1} - U_k^{n+1})$, and mechanical load $\lambda(\lambda_{k+1}^{n+1} - \lambda_{k+1}^{n+1})$ obtained with the new iterative numerical continuation algorithm (see (26)–(30) with $\Theta = \Theta_1$) for Example 1 ($E_1$) with the following parameters: (i) $\alpha = 50$, $\beta = -100$, and $\gamma = 1$ (blue solid line with circles); (ii) $\alpha = 3000$, $\beta = -1$, and $\gamma = 1$ (green solid line with squares); (iii) $\alpha = 30$, $\beta = -200$, and $\gamma = 1$ (magenta solid line with diamonds); (iv) $\alpha = 80$, $\beta = -600$, and $\gamma = 1$ (red solid line with upward-pointing triangles); (v) $\alpha = 100$, $\beta = -8$, and $\gamma = 1$ (cyan solid line with downward-pointing triangles).

Figure 9: Evolution of the approximate solutions of first-order partial derivatives associated with the displacement $U((\partial \mathcal{E}/\partial U)(U_{k+1}^{n+1}, \lambda_{k+1}^{n+1}))$ and the mechanical load $\lambda((\partial \mathcal{E}/\partial \lambda)(\lambda_{k+1}^{n+1}, U_{k+1}^{n+1}))$ obtained with the new iterative numerical continuation algorithm (see (26)–(30) with $\Theta = \Theta_1$) for Example 1 ($E_1$) with the following parameters: (i) $\alpha = 50$, $\beta = -100$, and $\gamma = 1$ (blue solid line with circles); (ii) $\alpha = 3000$, $\beta = -1$, and $\gamma = 1$ (green solid line with squares); (iii) $\alpha = 30$, $\beta = -200$, and $\gamma = 1$ (magenta solid line with diamonds); (iv) $\alpha = 80$, $\beta = -600$, and $\gamma = 1$ (red solid line with upward-pointing triangles); (v) $\alpha = 100$, $\beta = -8$, and $\gamma = 1$ (cyan solid line with downward-pointing triangles).
The $(k+1)$th iterative solution is obtained when \( H(U; \lambda_{n+1}^k) = W(U; \lambda_{n+1}^k) \) with \( U = U_{n+1}^{k+1} \); that is, 

\[
H(U_{n+1}^{k+1}; \lambda_{n+1}^k) \iff \Theta \left[ U_{n+1}^{k+1} - U_n^k \right] + \varepsilon(U_n^k; \lambda_n^k) = - \frac{1}{\Theta_1} \left( U_n^{k+1} \right) 
- U_{n+1}^1 + \varepsilon(U_{n+1}^1; \lambda_n) \iff H(U_{n+1}^1; \lambda_n) \iff
\]

with \( \Theta = \Theta_1 = \alpha \frac{\partial \varepsilon}{\partial U}(U_n^k; \lambda_n) + \beta \frac{\partial \varepsilon}{\partial \lambda} (U_n^k; \lambda_n) \) or \( \Theta_2 = \alpha \frac{\partial \varepsilon}{\partial U}(U_{n+1}^k; \lambda_{n+1}^k) + \beta \frac{\partial \varepsilon}{\partial \lambda} (U_{n+1}^k; \lambda_{n+1}^k) \)

∀\( \frac{\partial \varepsilon}{\partial \lambda} (U_n^k; \lambda_n) \neq 0 \), ∀\( \frac{\partial \varepsilon}{\partial U}(U_{n+1}^k; \lambda_{n+1}^k) \neq 0 \) ∀\( k = 1, \ldots, K \) ∀\( n = 0, \ldots, N \).

In the same way with (20), the iterative solution \( \lambda_{n+1}^{k+1} \) can be determined as follows (with \( \varepsilon(U_{n+1}^{k+1}; \lambda_{n+1}^k) = 0 \):

\[
\lambda_{n+1}^{k+1} = \lambda_{n+1}^k - \frac{1}{\left( \frac{\partial \varepsilon}{\partial \lambda}(U_{n+1}^k; \lambda_{n+1}^k) \right)} \left[ \varepsilon(U_{n+1}^k; \lambda_{n+1}^k) \right] 
+ \frac{\partial \varepsilon}{\partial \lambda} \left( U_{n+1}^k; \lambda_{n+1}^k \right) \left( U_{n+1}^k - U_n^k \right) 
\forall \frac{\partial \varepsilon}{\partial \lambda} (U_{n+1}^k; \lambda_{n+1}^k) \neq 0 \text{ } \forall k = 1, \ldots, K \text{ } \forall n = 0, \ldots, N.
\]

3.2. Some Comments. It is important to emphasize the following:

(1) For the first iterative step at the first increment (i.e., \( k = 1 \) and \( n = 0 \)), the choice of sign “±” in (17) is arbitrary and depends only on the direction for beginning the numerical continuation procedure. Further in this study, we adopt the sign “+.” Nevertheless, for the other first iterative steps at other increments (i.e., \( k = 1 \) and \( n \geq 1 \)), we consider a certain continuity of path with respect to the previous converged increment and we adopt the following rule: (i) the sign “+” when
Figure 11: Evolution of the approximate solutions \((U^k_{n+1}, \lambda^k_{n+1})\) obtained with the new iterative numerical continuation algorithm (blue line; see (26)–(30) with \(\Theta = \Theta_1\)) and compared with the exact solution (black solid line) for Example 1 \((E_1)\) with the parameters \(\alpha = 50\) and \(\beta = -100\) and for different values of the parameter \(\gamma\): (i) \(\gamma = 1\) (blue solid line with upward-pointing triangles); (ii) \(\gamma = 1.4\) (blue dotted line with pentagrams); (iii) \(\gamma = 1.5\) (blue dash-dot line with plus signs); (iv) \(\gamma = 1.6\) (blue dashed line with crosses).

Figure 12: Evolution of the residue error of the function \(\mathcal{R}(|\mathcal{R}(U^k_{n+1}, \lambda^k_{n+1})|)\), the approximation error of displacement \(U (|U^{k+1}_{n+1} - U^k_{n+1}|)\), and mechanical load \(\lambda (|\lambda^{k+1}_{n+1} - \lambda^k_{n+1}|)\) obtained with the new iterative numerical continuation algorithm (blue line; see (26)–(30) with \(\Theta = \Theta_1\)) for Example 1 \((E_1)\) with the parameters \(\alpha = 50\) and \(\beta = -100\) and for different values of the parameter \(\gamma\): (i) \(\gamma = 1\) (blue solid line with upward-pointing triangles); (ii) \(\gamma = 1.4\) (blue dotted line with pentagrams); (iii) \(\gamma = 1.5\) (blue dash-dot line with plus signs); (iv) \(\gamma = 1.6\) (blue dashed line with crosses).
\( \gamma = 1.5 \)

(iii) pointing triangles); (ii) values of the parameter \( \gamma \) with the parameters \( \alpha = 50 \) and \( \beta = -100 \) and for different values of the parameter \( \gamma \): (i) \( \gamma = 1 \) (blue solid line with upward-pointing triangles); (ii) \( \gamma = 1.4 \) (blue dotted line with pentagrams); (iii) \( \gamma = 1.5 \) (blue dash-dot line with plus signs); (iv) \( \gamma = 1.6 \) (blue dashed line with crosses).

**Figure 13:** Evolution of the approximate solutions of first-order partial derivatives associated with the displacement \( U (\partial \bar{\mathcal{E}}/\partial U)(U_{k+1}^k; \lambda_{n+1}^k) \) and the mechanical load \( \lambda (\partial \bar{\mathcal{E}}/\partial \lambda)(\lambda_{n+1}^k; U_{n+1}^k) \) obtained with the new iterative numerical continuation algorithm (blue line; see (26)–(30) with \( \Theta = \Theta_1 \)) for Example 1 \((E_1)\) with the parameters \( \alpha = 50 \) and \( \beta = -100 \) and for different values of the parameter \( \gamma \): (i) \( \gamma = 1 \) (blue solid line with upward-pointing triangles); (ii) \( \gamma = 1.4 \) (blue dotted line with pentagrams); (iii) \( \gamma = 1.5 \) (blue dash-dot line with plus signs); (iv) \( \gamma = 1.6 \) (blue dashed line with crosses).

\[
(\partial \bar{\mathcal{E}}/\partial U)(U_{n+1}^k; \lambda_{n+1}^k) > 0 \text{ or } (\partial \bar{\mathcal{E}}/\partial U)(U_{n+1}^k; \lambda_{n+1}^k) < 0 \text{ and } (\partial \bar{\mathcal{E}}/\partial \lambda)(\lambda_{n+1}^k; U_{n+1}^k) > 0; \text{ (ii) the sign "-" when } (\partial \bar{\mathcal{E}}/\partial U)(U_{n+1}^k; \lambda_{n+1}^k) > 0 \text{ or } (\partial \bar{\mathcal{E}}/\partial U)(U_{n+1}^k; \lambda_{n+1}^k) < 0 \text{ and } (\partial \bar{\mathcal{E}}/\partial \lambda)(\lambda_{n+1}^k; U_{n+1}^k) < 0.
\]

(2) The new proposed iterative algorithm (see Section 3.1) uses the first-order partial derivative operator as classical Newton’s algorithm (see Section 2.2.1), but unlike this latter, the critical points can be passed without relatively strong difficulties. Based on a modified Newton’s procedure, the new algorithm considers iterative steps of predictions and corrections which depend on both the first-order partial derivative operator \((\partial \bar{\mathcal{E}}/\partial U)(U_{n+1}^k; \lambda_{n+1}^k), \forall k = 0, \ldots, K, \forall n = 0, \ldots, N\) and its inverse \((1/(\partial \bar{\mathcal{E}}/\partial U)(U_{n+1}^k; \lambda_{n+1}^k), \forall k = 0, \ldots, K, \forall n = 0, \ldots, N\) modulated by two parameters \( \alpha \) and \( \beta \) allowing to pass through the limit points for the first (see (17) and (18)) and other iterations (see (23) and (24)) during a discrete-time interval.

(3) The two values associated with the function \( \Theta \) (see (15) and (24)), which is a linear combination...
of the first-order partial derivative operator and its
inverse ($\Theta = \Theta_1$), the initial straight line previously converged
$H(U;\lambda_n)$ (i.e., $H(U;\lambda_{n+1}^k)$ with $k = 0$, \forall $n = 0, \ldots, N$); (ii) when \( \Theta = \Theta_2 \), the current iterative straight line is
$H(U;\lambda_{n+1}^k)$ (\forall $k = 0, \ldots, K$, \forall $n = 0, \ldots, N$).

(4) The new proposed algorithm is composed of three
parameters: \( \alpha \), \( \beta \), and \( \gamma \). The optimal values of these
parameters depend clearly on the nonlinear function
under consideration. Therefore, a sensibility analysis
must be conducted in order to obtain these optimal
values. An alternative approach consists, in the first
time, to choose the following values: (i) for \( \alpha \) given:
(i-a) \( \beta = \alpha \) and \( \gamma = 1 \); (i-b) \( \beta = 1/\alpha \) and \( \gamma = 1 \); and
(i-c) \( \beta = 0 \) and \( \gamma = 1 \); (ii) for \( \beta \) given: (ii-a) \( \alpha = 0 \) and \( \gamma = 1 \). On the other hand, in the case where \( \alpha = 1 \)
and \( \beta = 0 \), the new proposed algorithm reduces to a
pseudo-arclength type procedure (see Section 2.2.2).

4. Some Numerical Examples

4.1. Preliminary Remarks. In the current section, we propose
to test and evaluate the accuracy, efficiency, and robustness
associated with the developed iterative method associated
with the numerical continuation procedure in Section 3 on
some scalar nonlinear equations. Moreover, all the numerical
results of this section have been obtained with MATLAB
software (see [7]).

New Iterative Numerical Continuation Algorithm (See Figure 3)

(i) For the iterative solution $U_{n+1}^k (k \geq 1)$, one has the following:

(a) For the first iteration ($k = 1$),

$$U_{n+1}^1 = U_n + \frac{\gamma}{\sqrt{1 + (\Theta_1)^2}}$$

(26)
\begin{equation}
\Theta_1 = \alpha \frac{\partial \mathcal{E}}{\partial U} (U_n, \lambda_n) + \beta \left( \frac{\partial \mathcal{E}}{\partial U} (U_n, \lambda_n) \right)
\end{equation}
(27)

\begin{equation}
\forall \frac{\partial \mathcal{E}}{\partial U} (U_n, \lambda_n) \neq 0 \, \forall n = 0, \ldots, N.
\end{equation}

(b) For the other iterations \((k \geq 2)\),
\begin{equation}
U_{n+1}^k = \frac{1}{\Theta + 1/\Theta_1} \left( \Theta U_{n+1}^{k-1} + \frac{1}{\Theta_1} U_{n+1}^{k-1} \right)
+ \mathcal{K}^{1} (U_{n+1}, \lambda_n) - \mathcal{E} (U_{n+1}^{k-1}, \lambda_{n+1})
\end{equation}
(28)

with
Figure 20: Evolution of the residue error of the function $\mathcal{E}(U_{k,n+1}^i, \lambda_{k,n+1}^i)$, the approximation error of displacement $U_k(U_{k+1,n+1}^i - U_{k,n+1}^i)$, and mechanical load $\lambda$ $(\lambda_{k+1,n+1}^i - \lambda_{k,n+1}^i)$ obtained with the new iterative numerical continuation algorithm (cyan line; see (26)–(30) with $\Theta = \Theta_1$) for Example 2 $(E_2)$ with the parameters $\alpha = 3000$ and $\beta = 10$ and for different values of the parameter $\gamma$: (i) $\gamma = 2$ (cyan solid line with downward-pointing triangles); (ii) $\gamma = 4$ (cyan dotted line with pentagrams); (iii) $\gamma = 5$ (cyan dash-dot line with plus signs); (iv) $\gamma = 6.65$ (cyan dashed line with crosses).

Figure 21: Evolution of the approximate solutions of first-order partial derivatives associated with the displacement $U_k(U_{k+1,n+1}^i; U_{k,n+1}^i)$ and the mechanical load $\lambda$ $(\lambda_{k+1,n+1}^i; U_{k,n+1}^i)$ obtained with the new iterative numerical continuation algorithm (cyan line; see (26)–(30) with $\Theta = \Theta_1$) for Example 2 $(E_2)$ with the parameters $\alpha = 3000$ and $\beta = 10$ and for different values of the parameter $\gamma$: (i) $\gamma = 2$ (cyan solid line with downward-pointing triangles); (ii) $\gamma = 4$ (cyan dotted line with pentagrams); (iii) $\gamma = 5$ (cyan dash-dot line with plus signs); (iv) $\gamma = 6.65$ (cyan dashed line with crosses).
Figure 22: Evolution of the number of iterations \((k)\) for each increment \((n)\) obtained with the new iterative numerical continuation algorithm (cyan line; see (26)–(30) with \(\Theta = \Theta_1\) for Example 2 \((E_1)\) with the parameters \(\alpha = 3000\) and \(\beta = 10\) and for different values of the parameter \(\gamma\): (i) \(\gamma = 2\) (cyan solid line with downward-pointing triangles); (ii) \(\gamma = 4\) (cyan dotted line with pentagrams); (iii) \(\gamma = 5\) (cyan dash-dot line with plus signs); (iv) \(\gamma = 6.65\) (cyan dashed line with crosses).

\[
\Theta = \begin{cases}
\Theta_1 = \alpha \frac{\partial \mathcal{E}}{\partial U}(U_n; \lambda_n) + \frac{\beta}{\partial \mathcal{E}/\partial \lambda}(U_n; \lambda_n) \\
or \\
\Theta_2 = \alpha \frac{\partial \mathcal{E}}{\partial U}(U_{n+1}^k; \lambda_{n+1}^k) + \frac{\beta}{\partial \mathcal{E}/\partial \lambda}(U_{n+1}^k; \lambda_{n+1}^k)
\end{cases}
\]

\forall \frac{\partial \mathcal{E}}{\partial U}(U_n; \lambda_n) \neq 0, \forall \frac{\partial \mathcal{E}}{\partial U}(U_{n+1}^k; \lambda_{n+1}^k) \neq 0 \forall k = 2, \ldots, K \forall n = 0, \ldots, N.

(ii) For the iterative solution \(\lambda_{n+1}^k (k \geq 1)\),

\[
\lambda_{n+1}^k = \lambda_{n+1}^{k-1} - \frac{1}{(\partial \mathcal{E}/\partial \lambda)(\lambda_{n+1}^{k-1}; U_{n+1}^k)} \left[ \mathcal{E}(\lambda_{n+1}^{k-1}; U_{n+1}^k) \right]
\]

\[
+ \frac{\partial \mathcal{E}}{\partial U}(U_{n+1}^k; \lambda_{n+1}^{k-1})(U_{n+1}^k - U_{n+1}^{k-1})
\]

\forall \frac{\partial \mathcal{E}}{\partial \lambda}(\lambda_{n+1}^{k-1}; U_{n+1}^k) \neq 0 \forall k = 1, \ldots, K \forall n = 0, \ldots, N.

It is important to emphasize the following:

(i) On the one hand, we consider only the case where \(\Theta = \Theta_1\) for the new continuation algorithm (see (26)–(30)) used in this section.

(ii) On the other hand, we introduce four types of Convergence Criterion \(\text{CC}_l\) (with \(l = 1, \ldots, 4\)) in order to stop the iterative process associated with the new proposed algorithm:

\[
\text{CC}_1 : K \leq K_{\text{max}}
\]

\[
\text{CC}_2 : \mathcal{E}(U_{n+1}^k; \lambda_{n+1}^k) \leq \varepsilon_{\text{re}}
\]

\[
\text{CC}_3 : \left| U_{n+1}^{k+1} - U_{n+1}^k \right| \leq \varepsilon_{\text{ae1}}
\]

\[
\text{CC}_4 : \left| \lambda_{n+1}^{k+1} - \lambda_{n+1}^k \right| \leq \varepsilon_{\text{ae2}}
\]

\forall k = 0, \ldots, K \forall n = 0, \ldots, N,

where \(K_{\text{max}}\) represents the maximum number of iterations, \(\varepsilon_{\text{re}}\) and \(\varepsilon_{\text{ae1}}\) (resp., \(\varepsilon_{\text{ae2}}\)) are the tolerance parameters associated with the residue error of the function \(\mathcal{E}\) and approximation error criterion of displacement \(U\) (resp., mechanical load \(\lambda\)). In what follows, we consider the following values for each CC: \(K_{\text{max}} = 30, \varepsilon_{\text{re}} = 10^{-10}\), and \(\varepsilon_{\text{ae1}} = \varepsilon_{\text{ae2}} = 10^{-10}\).

4.2. Examples. We consider the following scalar nonlinear equations:

\[
(E_1) \quad \mathcal{E}(U, \lambda) = \lambda^2 + U^2 - \theta^2 = 0
\]

\[
(E_2) \quad \mathcal{E}(U, \lambda) = \lambda + \tau U^2 - \nu U = 0
\]

\[
(E_3) \quad \mathcal{E}(U, \lambda) = \lambda - \mu \sin(\eta U) = 0,
\]

where \(\theta, \tau, \nu, \mu, \) and \(\eta\) are five parameters chosen with these values: \(\theta = 10^2, \tau = 0.5, \nu = 10, \mu = 100, \) and \(\eta = 10^{-2}\).
Figure 23: Evolution of the approximate solutions \((U_{n+1}^k, \lambda_{n+1}^k)\) obtained with the new iterative numerical continuation algorithm (see (26)–(30) with \(\Theta = \Theta_1\)) and compared with the exact solution (black solid line) for Example 3 \((E_3)\) with the following parameters: (i) \(\alpha = 10000, \beta = 200,\) and \(\gamma = 20\) (blue solid line with circles); (ii) \(\alpha = 10000, \beta = 1000,\) and \(\gamma = 20\) (green solid line with squares); (iii) \(\alpha = 10000, \beta = 500,\) and \(\gamma = 20\) (magenta solid line with diamonds); (iv) \(\alpha = 1000, \beta = 5000,\) and \(\gamma = 20\) (red solid line with upward-pointing triangles); (v) \(\alpha = 100, \beta = 5000,\) and \(\gamma = 20\) (cyan solid line with downward-pointing triangles).

Figure 24: Evolution of the residue error of the function \(|\mathcal{E}(U_{n+1}^k, \lambda_{n+1}^k)|\), the approximation error of displacement \(|U_{n+1}^{k+1} - U_{n+1}^k|\), and mechanical load \(|\lambda_{n+1}^{k+1} - \lambda_{n+1}^k|\) obtained with the new iterative numerical continuation algorithm (see (26)–(30) with \(\Theta = \Theta_1\)) for Example 3 \((E_3)\) with the following parameters: (i) \(\alpha = 10000, \beta = 200,\) and \(\gamma = 20\) (blue solid line with circles); (ii) \(\alpha = 10000, \beta = 1000,\) and \(\gamma = 20\) (green solid line with squares); (iii) \(\alpha = 10000, \beta = 500,\) and \(\gamma = 20\) (magenta solid line with diamonds); (iv) \(\alpha = 1000, \beta = 5000,\) and \(\gamma = 20\) (red solid line with upward-pointing triangles); (v) \(\alpha = 100, \beta = 5000,\) and \(\gamma = 20\) (cyan solid line with downward-pointing triangles).
4.3. Results and Discussion

4.3.1. Predictive Numerical Capabilities. All the numerical results associated with scalar nonlinear functions \((E_1, E_2, E_3, E_4)\) (see Section 4.2) are presented in Figures 4–6. The obtained numerical results show fairly clearly that the new proposed iterative numerical algorithm (see Section 3) is capable of giving convenient approximate solutions associated with parameterized scalar nonlinear equations including critical points. Indeed, in Example 1 \((E_1)\), there are four critical points and we can see that the new iterative algorithm is able to pass all these points: (i) the first \((U_{cr_1}, \lambda_{cr_1}) = (100, 0)\) (representing case (c) of Figure 1); (ii) the second \((U_{cr_2}, \lambda_{cr_2}) = (0, -100)\) (representing case (b) of Figure 1); (iii) the third \((U_{cr_3}, \lambda_{cr_3}) = (-100, 0)\) (representing case (d) of Figure 1); and (iv) the fourth \((U_{cr_4}, \lambda_{cr_4}) = (0, 100)\) (representing case (a) of Figure 1). In Example 2 \((E_2)\), there is only one critical point \((U_{cr_5}, \lambda_{cr_5}) = (10, 50)\) (representing case (a) of Figure 1) and the new iterative algorithm passes without excessive difficulty this singularity. In the same way, in Example 3 \((E_3)\),
Figure 28: Evolution of the residue error of the function $\mathcal{E}(|E(U_{n+1}^{k},\lambda_{n+1}^{k})|)$, the approximation error of displacement $U(|U_{n+1}^{k+1} - U_{n+1}^{k}|)$, and mechanical load $\lambda(|\lambda_{n+1}^{k+1} - \lambda_{n+1}^{k}|)$ obtained with the new iterative numerical continuation algorithm (red line; see (26)–(30) with $\Theta = \Theta_1$) for Example 3 ($E_3$) with the parameters $\alpha = 1000$ and $\beta = 5000$ and for different values of the parameter $\gamma$: (i) $\gamma = 20$ (red solid line with upward-pointing triangles); (ii) $\gamma = 25$ (red dotted line with pentagrams); (iii) $\gamma = 50$ (red dash-dot line with plus signs); (iv) $\gamma = 33$ (red dashed line with crosses).

Figure 29: Evolution of the approximate solutions of first-order partial derivatives associated with the displacement $U((\partial E/\partial U)(U_{n+1}^{k+1},\lambda_{n+1}^{k}))$ and the mechanical load $\lambda((\partial E/\partial \lambda)(\lambda_{n+1}^{k+1},U_{n+1}^{k+1}))$ obtained with the new iterative numerical continuation algorithm (red line; see (26)–(30) with $\Theta = \Theta_1$) for Example 3 ($E_3$) with the parameters $\alpha = 1000$ and $\beta = 5000$ and for different values of the parameter $\gamma$: (i) $\gamma = 20$ (red solid line with upward-pointing triangles); (ii) $\gamma = 25$ (red dotted line with pentagrams); (iii) $\gamma = 50$ (red dash-dot line with plus signs); (iv) $\gamma = 33$ (red dashed line with crosses).
there are periodic critical points and we can see that the new algorithm passes again the first \((U_{cr}, \lambda_{cr}) = (157, 100)\) (representing case (a) of Figure 1) and the second \((U_{cr}, \lambda_{cr}) = (471, -100)\) (representing case (b) of Figure 1) singularity associated with the nonlinear function in the interval \([0, T_{per}]\) (where \(T_{per} = 2\pi/\eta \approx 628.31\) denotes the period of the sinusoidal function). In the light of all the numerical results obtained in Figures 4, 5, and 6, the new iterative numerical continuation algorithm enables passing through the critical points and predicting the approximate solutions \((U_{n+1}^k, \lambda_{n+1}^k)\) associated with parameterized scalar nonlinear equations.

4.3.2. Sensibility Analysis. In this section, we propose a sensibility analysis for evaluating the influence of different values assigned with the parameters (\(\alpha, \beta,\) and \(\gamma\)) used by the new iterative numerical continuation algorithm (see Section 3). All the numerical results associated with the scalar nonlinear functions \((E_1), (E_2),\) and \((E_3)\) (see Section 4.2) with different values of the parameters (\(\alpha, \beta,\) and \(\gamma\)) are presented in Figures 7–30.

In the light of all numerical results obtained in Sections 4.3.1 and 4.3.2, the new iterative numerical continuation algorithm is a relatively accurate, efficient, and robust method that allows passing specific critical points and providing suitable approximate solutions \((U_{n+1}^k, \lambda_{n+1}^k)\) associated with parameterized scalar nonlinear equations.

5. Conclusion

The present paper is devoted to a new iterative numerical continuation procedure for approximating the solutions associated with parameterized scalar nonlinear equations. Coupled with a modified Newton-type method and a stationary numerical technique, the presented algorithm is capable of providing satisfactory numerical solutions for scalar nonlinear equations using one control parameter. Through some illustrative examples, the predictive abilities of this new algorithm are tested, assessed, and discussed.

Competing Interests

The author declares that he has no competing interests.

References


Submit your manuscripts at
https://www.hindawi.com