

Article

Modified Potra–Pták Multi-step Schemes with Accelerated Order of Convergence for Solving Systems of Nonlinear Equations

Himani Arora ^{1,*}, Juan R. Torregrosa ²  and Alicia Cordero ² 

¹ Department of Mathematics, D.A.V. University, Sarmastpur, 144012 Jalandhar, India

² Instituto de Matemática Multidisciplinar, Universitat Politècnica de València, 46022 Valencia, Spain; jrortre@mat.upv.es (J.R.T.); acordero@mat.upv.es (A.C.)

* Correspondence: arorahimani362@gmail.com

Received: 29 November 2018; Accepted: 23 December 2018; Published: 27 December 2018

Abstract: In this study, an iterative scheme of sixth order of convergence for solving systems of nonlinear equations is presented. The scheme is composed of three steps, of which the first two steps are that of third order Potra–Pták method and last is weighted-Newton step. Furthermore, we generalize our work to derive a family of multi-step iterative methods with order of convergence $3r + 6, r = 0, 1, 2, \dots$. The sixth order method is the special case of this multi-step scheme for $r = 0$. The family gives a four-step ninth order method for $r = 1$. As much higher order methods are not used in practice, so we study sixth and ninth order methods in detail. Numerical examples are included to confirm theoretical results and to compare the methods with some existing ones. Different numerical tests, containing academical functions and systems resulting from the discretization of boundary problems, are introduced to show the efficiency and reliability of the proposed methods.

Keywords: systems of nonlinear equations; iterative methods; Newton’s method; order of convergence; computational efficiency; basin of attraction

1. Introduction

Many applied problems in Science and Engineering [1–3] are reduced to solve nonlinear systems $F(x) = 0$ numerically, that is, for a given nonlinear function $F(x) : D \subset R^m \rightarrow R^m$, where $F(x) = (f_1(x), f_2(x), \dots, f_m(x))^T$ and $x = (x_1, x_2, \dots, x_m)^T$, to find a vector $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_m)^T$ such that $F(\alpha) = 0$. The most widely used method for this purpose is the classical Newton’s method [3,4], which converges quadratically under the conditions that the function F is continuously differentiable and a good initial approximation $x^{(0)}$ is given. It is defined by

$$x^{(k+1)} = x^{(k)} - F'(x^{(k)})^{-1}F(x^{(k)}), \quad k = 0, 1, \dots, \quad (1)$$

where $F'(x^{(k)})^{-1}$ is the inverse of Fréchet derivative $F'(x^{(k)})$ of the function $F(x)$ at $x^{(k)}$. In order to improve the order of convergence of Newton’s method, several methods have been proposed in literature, see, for example [5–11] and references therein. In particular, the third order method by Potra–Pták [11] for systems of nonlinear equations is given by

$$\begin{aligned} y^{(k)} &= x^{(k)} - F'(x^{(k)})^{-1}F(x^{(k)}), \\ x^{(k+1)} &= y^{(k)} - F'(x^{(k)})^{-1}F(y^{(k)}) \quad k = 0, 1, \dots \end{aligned} \quad (2)$$

It is quite clear that this scheme requires the evaluation of two functions, one derivative and one matrix inversion per iteration, that is usually avoided by solving a linear system. This algorithm is illustrious not only for its simplicity but also for its efficient character.

In this paper, based on Potra–Pták method (2), we develop a three-step scheme with increased order of convergence and still maintaining the efficient character. With these considerations, we propose a three-step iterative method with accelerated sixth order of convergence; of the three steps, the first two are those of Potra–Pták method whereas the third is a weighted Newton-step. Then, based on this scheme, a multi-step family with increasing order of convergence $3r + 6, r = 0, 1, 2, \dots$, is developed. The sixth order method is the special case of this multi-step scheme for $r = 0$. The family gives a four-step ninth order scheme for $r = 1$. As much higher order methods are not used in practice, so we study sixth and ninth order methods in particular.

The rest of the paper is organized as follows: In Section 2, we present the new three-step scheme for solving nonlinear systems and we analyze its order of convergence. In Section 2.1, the order of this scheme is improved in three units, by adding another step that needs a new functional evaluation and to solve a linear system with the same matrix of coefficients as before. This idea can be generalized for designing an iterative method with arbitrary order of convergence. The computational efficiency of the proposed schemes is studied in Section 3, doing a comparative analysis with the efficiency of other known methods. Section 4 is devoted to the numerical experiments with academical multivariate functions and nonlinear systems resulted of the discretization of boundary problems. For some cases, the basins of attraction of the methods are also showed. The paper finishes with some conclusions and the references used in it.

2. The Method and Analysis of Convergence

We start with the following iterative scheme:

$$\begin{aligned} y^{(k)} &= x^{(k)} - F'(x^{(k)})^{-1}F(x^{(k)}), \\ z^{(k)} &= y^{(k)} - F'(x^{(k)})^{-1}F(y^{(k)}), \\ x^{(k+1)} &= z^{(k)} - (aI + F'(x^{(k)})^{-1}[z^{(k)}, y^{(k)}; F])(bI + cF'(x^{(k)})^{-1}[z^{(k)}, y^{(k)}; F])F'(x^{(k)})^{-1}F(z^{(k)}), \end{aligned} \tag{3}$$

where the first two-steps are those of Potra–Pták scheme [11] for nonlinear systems and $[z^{(k)}, y^{(k)}; F]$ is the first order divided difference of F .

In order to discuss the behavior of scheme (3), we consider the following expression of divided difference operator $[\cdot, \cdot; F] : D \times D \subset R^m \times R^m \rightarrow L(R^m)$, see for example, [2,10],

$$[x + h, x; F] = \int_0^1 F'(x + th) dt, \forall x, h \in R^m. \tag{4}$$

By expanding $F'(x + th)$ in Taylor series at the point x and integrating, we have

$$[x + h, x; F] = \int_0^1 F'(x + th) dt = F'(x) + \frac{1}{2}F''(x)h + \frac{1}{6}F'''(x)h^2 + O(h^3). \tag{5}$$

where $h^i = (h, h, \dots, h)$, $h \in R^m$.

Let $e^{(k)} = x^{(k)} - \alpha$. Developing $F(x^{(k)})$ in a neighborhood of α and assuming that $\Gamma = [F'(\alpha)]^{-1}$ exists, we have

$$F(x^{(k)}) = F'(\alpha)(e^{(k)} + A_2(e^{(k)})^2 + A_3(e^{(k)})^3 + A_4(e^{(k)})^4 + O((e^{(k)})^5)), \tag{6}$$

where $A_i = \frac{1}{i!}\Gamma F^{(i)}(\alpha) \in L_i(R^m, R^m)$ and $(e^{(k)})^i = (e^{(k)}, e^{(k)}, \dots, e^{(k)})$, $e^{(k)} \in R^m$.

Also,

$$F'(x^{(k)}) = F'(\alpha)(I + 2A_2e^{(k)} + 3A_3(e^{(k)})^2 + 4A_4(e^{(k)})^3 + O((e^{(k)})^4)), \tag{7}$$

$$F''(x^{(k)}) = F'(\alpha)(2A_2 + 6A_3e^{(k)} + 12A_4(e^{(k)})^2 + O((e^{(k)})^3)), \tag{8}$$

$$F'''(x^{(k)}) = F'(\alpha)(6A_3 + 24A_4e^{(k)} + O((e^{(k)})^2)). \tag{9}$$

Inversion of $F'(x^{(k)})$ yields,

$$F'(x^{(k)})^{-1} = \left[I - 2A_2e^{(k)} + (4A_2^2 - 3A_3)(e^{(k)})^2 - (8A_2^3 - 6A_2A_3 - 6A_3A_2 + 4A_4)(e^{(k)})^3 + O((e^{(k)})^4) \right] \Gamma. \tag{10}$$

We are in a position to analyze the behavior of scheme (3). Thus, the following result is proven.

Theorem 1. *Let $F : D \subset R^m \rightarrow D$ be sufficiently differentiable function in an open neighborhood D of its zero α . Let us suppose that the Jacobian matrix $F'(x)$ is continuous and nonsingular at α . If an initial approximation $x^{(0)}$ is sufficiently close to α , then the local order of convergence of method (3) is at least 6, provided $a = 13/4, b = -7/2$ and $c = 5/4$.*

Proof. Let $e_y^{(k)} = y^{(k)} - \alpha$ is the local error of Newton’s method given by

$$e_y^{(k)} = A_2(e^{(k)})^2 - 2(A_2^2 - A_3)(e^{(k)})^3 + O((e^{(k)})^4). \tag{11}$$

Employing Equations (10) and (11) in the second step of (3), we get

$$e_z^{(k)} = z^{(k)} - \alpha = 2A_2e^{(k)}e_y^{(k)} + O((e^{(k)})^4). \tag{12}$$

Using Equations (7)–(9) in (5) for $x + h = z^{(k)}, x = y^{(k)}$ and $h = e_z^{(k)} - e_y^{(k)}$, it follows that

$$[z^{(k)}, y^{(k)}; F] = F'(\alpha)(I + A_2(e_y^{(k)} + e_z^{(k)}) + O((e^{(k)})^4)). \tag{13}$$

With the help of Equations (10) and (13), we can write

$$\begin{aligned} & aI + F'(x^{(k)})^{-1}[z^{(k)}, y^{(k)}; F](bI + cF'(x^{(k)})^{-1}[z^{(k)}, y^{(k)}; F]) \\ &= \left[(a + b + c)I - 2(b + 2c)A_2e^{(k)} + ((5b + 14c)A_2^2 - 3(b + 2c)A_3)(e^{(k)})^2 - (10(b + 4c)A_2^3 \right. \\ & \quad \left. - (8b + 22c)A_2A_3 - (6b + 18c)A_3A_2 + 4(b + 2c)A_4)(e^{(k)})^3 + O((e^{(k)})^4) \right] \Gamma. \end{aligned} \tag{14}$$

Expanding $F(z^{(k)})$ about α , we obtain

$$F(z^{(k)}) = F'(\alpha)(e_z^{(k)} + A_2(e_z^{(k)})^2 + O((e_z^{(k)})^3)). \tag{15}$$

Using (10), (14) and (15) in the last step of (3), we get

$$\begin{aligned} e^{(k+1)} = x^{(k+1)} - \alpha &= 2(1 - a - b - c)A_2^2(e^{(k)})^3 + 4(a + 2b + 3c)A_2^3(e^{(k)})^4 \\ & \quad - 2((4a + 13(b + 2c))A_2^2 - 3A_3(a + 2b + 3c))A_2^2(e^{(k)})^5 + 4((3a + 17b + 45c)A_2^5 \\ & \quad - (3a + 10(b + 2c))A_2A_3A_2^2 + (3a + 6(b + 3c))A_3A_2^3 + 2(a + 2b + 3c)A_4A_2^2)(e^{(k)})^6 \\ & \quad + O((e^{(k)})^7). \end{aligned} \tag{16}$$

In order to achieve sixth order of convergence it is clear that terms $1 - a - b - c, a + 2b + 3c$ and $4a + 13(b + 2c)$ must vanish for some values of $a, b,$ and c . This happens when $a = \frac{13}{4}, b = -\frac{7}{2}$ and $c = \frac{5}{4}$.

Thus, the error equation (16) becomes

$$e^{(k+1)} = x^{(k+1)} - \alpha = (26A_2^3 + A_2A_3 - 3A_3A_2)A_2^2(e^{(k)})^6 + O((e^{(k)})^7). \tag{17}$$

This proves the sixth order of convergence. \square

Thus, the proposed scheme (3) is given by

$$\begin{aligned}
 y^{(k)} &= x^{(k)} - F'(x^{(k)})^{-1}F(x^{(k)}), \\
 z^{(k)} &= y^{(k)} - F'(x^{(k)})^{-1}F(y^{(k)}), \\
 x^{(k+1)} &= z^{(k)} - \left(\frac{13}{4}I - F'(x^{(k)})^{-1}[z^{(k)}, y^{(k)}; F]\right) \left(\frac{7}{2}I - \frac{5}{4}F'(x^{(k)})^{-1}[z^{(k)}, y^{(k)}; F]\right) \\
 &\quad \times F'(x^{(k)})^{-1}F(z^{(k)}).
 \end{aligned} \tag{18}$$

Clearly this formula uses three functional evaluations, one evaluation of the Jacobian matrix, one of a divided difference, and one matrix inversion per iteration. We denote this scheme by $H_{6,1}$.

2.1. Multi-step Method with Order $3r + 6$

In this section, we improve the $H_{6,1}$ by adding a functional evaluation per each new step to get the multi-step version called $H_{3r+6,1}$ method. The method is defined as

$$\begin{aligned}
 x^{(k+1)} &= H_{3r+6,1}(x^{(k)}) = v_r(x^{(k)}), \\
 v_j(x^{(k)}) &= v_{j-1}(x^{(k)}) - \theta(x^{(k)})F'(x^{(k)})^{-1}F(v_{j-1}(x^{(k)})), \\
 \theta(x^{(k)}) &= \frac{13}{4}I - F'(x^{(k)})^{-1}[z^{(k)}, y^{(k)}; F] \left(\frac{7}{2}I - \frac{5}{4}F'(x^{(k)})^{-1}[z^{(k)}, y^{(k)}; F]\right), \\
 v_0 &= H_{6,1}(x^{(k)}), \quad j = 1, 2, \dots, r; \quad r \geq 1.
 \end{aligned} \tag{19}$$

Let us note that case $r = 0$ is $H_{6,1}$ method given by (18). This multi-step version has the order of convergence $3r + 6$, $r \geq 1$, which we shall prove through the following result.

Theorem 2. *Let us Assume that $F : D \subset \mathbb{R}^m \rightarrow D$ is a sufficiently Fréchet differentiable function in an open convex set D containing the zero α of $F(x)$ and $F'(x)$ is continuous and nonsingular at α . Then, sequence $\{x^{(k)}\}_{k \geq 0}$, $x^{(0)} \in D$ obtained by using method (19) converges to α with convergence order $3r + 6$.*

Proof. Let $e_{v_j}^{(k)} = v_j^{(k)} - \alpha$, for all $j = 1, 2, \dots, r$. Taylor’s expansion of $F(v_{j-1}^{(k)}(x^{(k)}))$ about α yields

$$F(v_{j-1}(x^{(k)})) = F'(\alpha)(e_{v_{j-1}}^{(k)} + A_2(e_{v_{j-1}}^{(k)})^2 + O((e_{v_{j-1}}^{(k)})^3)). \tag{20}$$

Using (10) and (14) for $a = 13/4$, $b = -7/2$ and $c = 5/4$, we have

$$\begin{aligned}
 \theta(x^{(k)})F'(x^{(k)})^{-1} &= \left(\frac{13}{4}I - F'(x^{(k)})^{-1}[z^{(k)}, y^{(k)}; F] \left(\frac{7}{2}I - \frac{5}{4}F'(x^{(k)})^{-1}[z^{(k)}, y^{(k)}; F]\right)\right) F'(x^{(k)})^{-1} \\
 &= [I - S_3(e^{(k)})^3 + O((e^{(k)})^4)] \Gamma,
 \end{aligned} \tag{21}$$

where $S_3 = 15A_2^3 + \frac{1}{2}A_2A_3 - \frac{3}{2}A_3A_2$.

With the help of (20) and (21), we can write

$$\begin{aligned}
 \theta(x^{(k)})F'(x^{(k)})^{-1}F(v_{j-1}(x^{(k)})) &= (I - S_3(e^{(k)})^2 + \dots)(e_{v_{j-1}}^{(k)} + A_2(e_{v_{j-1}}^{(k)})^2 + \dots) \\
 &= e_{v_{j-1}}^{(k)} - S_3(e^{(k)})^3 e_{v_{j-1}}^{(k)} + \dots.
 \end{aligned} \tag{22}$$

By substituting (22) in (19), we get

$$\begin{aligned}
 e_{v_j}^{(k)} &= v_j^{(k)} - \alpha = e_{v_{j-1}}^{(k)} - (e_{v_{j-1}}^{(k)} - S_3(e^{(k)})^3 e_{v_{j-1}}^{(k)} + \dots) \\
 &= S_3(e^{(k)})^3 e_{v_{j-1}}^{(k)} + \dots.
 \end{aligned} \tag{23}$$

As we know that $e_{v_0}^{(k)} = (26A_2^3 + A_2A_3 - 3A_3A_2)A_2^2(e^{(k)})^6$, so from (23), for $j = 1, 2, \dots$, we have that

$$\begin{aligned} e_{v_1}^{(k)} &= S_3(e^{(k)})^3 e_{v_0}^{(k)} \\ &= S_3(e^{(k)})^3 (26A_2^3 + A_2A_3 - 3A_3A_2)A_2^2(e^{(k)})^6 + \dots, \\ e_{v_2}^{(k)} &= S_3(e^{(k)})^2 e_{v_1}^{(k)} \\ &= S_3^2(e^{(k)})^6 (26A_2^3 + A_2A_3 - 3A_3A_2)A_2^2(e^{(k)})^6 + \dots. \end{aligned} \tag{24}$$

Proceeding by induction, we have

$$e_{v_r}^{(k)} = S_3^r (26A_2^3 + A_2A_3 - 3A_3A_2)A_2^2(e^{(k)})^{3r+6} + O((e^{(k)})^{3r+7}), \quad r \geq 0. \tag{25}$$

Hence, the result of the theorem is proven. \square

As a special case, this family when $r = 1$ gives a ninth order method:

$$\begin{aligned} y^{(k)} &= x^{(k)} - F'(x^{(k)})^{-1}F(x^{(k)}), \\ z^{(k)} &= x^{(k)} - F'(x^{(k)})^{-1}F(y^{(k)}), \\ v_0^{(k)} &= H_{6,1}(x^{(k)}, y^{(k)}), \\ x^{(k+1)} &= v_0^{(k)} - \theta(x^{(k)})F'(x^{(k)})^{-1}F(v_0^{(k)}) \quad k = 0, 1, 2, \dots \end{aligned} \tag{26}$$

It is clear that this scheme requires four functional evaluations, one Jacobian matrix, one divided difference, and one matrix inversion per full iteration. We denote this scheme by $H_{9,1}$.

Remark 1. Multi-step version H_{3r+6} ($r \geq 0$), utilizes $r + 3$ functional evaluations of F , one evaluation of F' , and one divided difference. Also, the scheme (19) requires only one matrix inversion per iteration.

3. Computational Efficiency

To obtain an estimation of the efficiency of the proposed methods we shall make use of efficiency index, according to which the efficiency of an iterative method is given by $E = \rho^{\frac{1}{c}}$, where ρ is the order of convergence and C is the computational cost per iteration. For a system of m nonlinear equations and m unknowns, the computational cost per iteration is given by (see [9,10])

$$C(\mu_0, \mu_1, m) = P_0(m)\mu_0 + P_1(m)\mu_1 + P(m), \tag{27}$$

where $P_0(m)$ represents the number of evaluations of scalar functions (f_1, f_2, \dots, f_m) used in the evaluations of F and $[y, x; F]$. The divided difference $[y, x; F]$ of F is an $m \times m$ matrix with elements (see [11,12])

$$\begin{aligned} [y, x; F]_{ij} &= (f_i(y_1, \dots, y_{j-1}, y_j, x_{j+1}, \dots, x_m) - f_i(y_1, \dots, y_{j-1}, x_j, x_{j+1}, \dots, x_m) \\ &\quad + f_i(x_1, \dots, x_{j-1}, y_j, y_{j+1}, \dots, y_m) \\ &\quad - f_i(x_1, \dots, x_{j-1}, x_j, y_{j+1}, \dots, y_m)) / (2(y_j - x_j)), \quad 1 \leq i, j \leq m. \end{aligned}$$

The number of evaluations of scalar functions of F' , i.e. $\frac{\partial f_i}{\partial x_j}, 1 \leq i, j \leq m$, is $P_1(m)$. $P(m)$ represents the number of products or quotients needed per iteration, and μ_0 and μ_1 are ratios between products and evaluations required to express the value of $C(\mu_0, \mu_1, m)$ in terms of products.

To compute F in any iterative method we calculate m scalar functions and if we compute the divided difference $[y, x; F]$ then we evaluate $2m(m - 1)$ scalar functions, where $F(x)$ and $F(y)$ are computed separately. We must add m^2 quotients from any divided difference. The number of scalar evaluations is m^2 for any new derivative F' . In order to compute an inverse linear operator we solve a linear system, where we have $m(m - 1)(2m - 1)/6$ products and $m(m - 1)/2$ quotients in the LU

decomposition and $m(m - 1)$ products and m quotients in the resolution of two triangular linear systems. We suppose that a quotient is equivalent to l products. Moreover, we add m^2 products for the multiplication of a matrix with a vector or of a matrix by a scalar and m products for the multiplication of a vector by a scalar.

Denoting the efficiency indices of $H_{\rho,i}$ ($\rho = 6, 9$ and $i = 1, 2, 3, 4$) by $E_{\rho,i}$ and computational cost by $C_{\rho,i}$, then taking into account the above considerations, we obtain

$$C_{6,1} = (2m^2 + m)\mu_0 + m^2\mu_1 + \frac{m}{6}(2m^2 + 39m - 11 + 9l(m + 3)) \quad \text{and} \quad E_{6,1} = 6^{1/C_{6,1}}. \quad (28)$$

$$C_{9,1} = (2m^2 + 2m)\mu_0 + m^2\mu_1 + \frac{m}{6}(2m^2 + 69m - 11 + 9l(m + 5)) \quad \text{and} \quad E_{9,1} = 9^{1/C_{9,1}}. \quad (29)$$

To check the performance of the new sixth order method, we compare it with some existing sixth order which belongs to the same class. So, we choose the sixth order methods presented in [10,13]. The sixth order methods presented in [10] are given by

$$\begin{aligned} y^{(k)} &= x^{(k)} - F'(x^{(k)})^{-1}F(x^{(k)}), \quad k = 0, 1, 2, \dots \\ z^{(k)} &= y^{(k)} - (2[y^{(k)}, x^{(k)}; F] - F'(x^{(k)}))^{-1}F(y^{(k)}), \\ x^{(k+1)} &= H_{6,2}(x^{(k)}, y^{(k)}, z^{(k)}) = z^{(k)} - (2[y^{(k)}, x^{(k)}; F] - F'(x^{(k)}))^{-1}F(z^{(k)}) \end{aligned} \quad (30)$$

and

$$\begin{aligned} y^{(k)} &= x^{(k)} - F'(x^{(k)})^{-1}F(x^{(k)}), \quad k = 0, 1, 2, \dots \\ z^{(k)} &= y^{(k)} - (2[y^{(k)}, x^{(k)}; F]^{-1} - F'(x^{(k)})^{-1})F(y^{(k)}), \\ x^{(k+1)} &= H_{6,3}(x^{(k)}, y^{(k)}, z^{(k)}) = z^{(k)} - (2[y^{(k)}, x^{(k)}; F]^{-1} - F'(x^{(k)})^{-1})F(z^{(k)}). \end{aligned} \quad (31)$$

The sixth order method proposed in [13] is given by

$$\begin{aligned} y^{(k)} &= x^{(k)} - F'(x^{(k)})^{-1}F(x^{(k)}), \quad k = 0, 1, 2, \dots \\ z^{(k)} &= y^{(k)} - (3I - 2F'(x^{(k)})^{-1}[y^{(k)}, x^{(k)}; F])F'(x^{(k)})^{-1}F(y^{(k)}), \\ x^{(k+1)} &= H_{6,4}(x^{(k)}, y^{(k)}, z^{(k)}) = z^{(k)} - (3I - 2F'(x^{(k)})^{-1}[y^{(k)}, x^{(k)}; F])F'(x^{(k)})^{-1}F(z^{(k)}). \end{aligned} \quad (32)$$

Per iteration these methods utilize the same number of function evaluations as that of $H_{6,1}$. The computational cost and efficiency for the methods $H_{6,2}$, $H_{6,3}$ and $H_{6,4}$ is given as below:

$$C_{6,2} = (2m^2 + m)\mu_0 + m^2\mu_1 + \frac{m}{3}(2m^2 + 9m - 8 + 6l(m + 1)) \quad \text{and} \quad E_{6,2} = 6^{1/C_{6,2}}, \quad (33)$$

$$C_{6,3} = (2m^2 + m)\mu_0 + m^2\mu_1 + \frac{m}{3}(2m^2 + 12m - 8 + 6l(m + 2)) \quad \text{and} \quad E_{6,3} = 6^{1/C_{6,3}}, \quad (34)$$

$$C_{6,4} = (2m^2 + m)\mu_0 + m^2\mu_1 + \frac{m}{3}(2m^2 + 39m - 5 + 9l(m + 3)) \quad \text{and} \quad E_{6,4} = 6^{1/C_{6,4}}. \quad (35)$$

3.1. Comparison among the Efficiencies

To compare the iterative methods $H_{\rho,i}$, we consider the ratio

$$R_{p,i;q,j} = \frac{\log E_{p,i}}{\log E_{q,j}} = \frac{\log(p) C_{q,j}}{\log(q) C_{p,i}}. \quad (36)$$

It is clear that if $R_{p,i;q,j} > 1$, the iterative method $H_{p,i}$ is more efficient than $H_{q,j}$. Moreover, if we need to compare the methods having the same order then using (36) we can say that the iterative method $H_{p,i}$

is more efficient than $H_{q,j}$, if $C_{q,j} > C_{p,i}$. This means that the comparison of the methods possessing same order can be done by just comparing their computational costs.

$H_{6,1}$ versus $H_{6,2}$ case:

In this case comparison of the corresponding values of $C_{6,1}$ and $C_{6,2}$ in (28) and (33) gives $E_{6,1} > E_{6,2}$ for all $m \geq 4$ and $l \geq 1$.

$H_{6,1}$ versus $H_{6,3}$ case:

Comparing the corresponding values of $C_{6,1}$ and $C_{6,3}$ in (28) and (34) we obtain $E_{6,1} > E_{6,3}$ for all $m \geq 2$ and $l \geq 1$.

$H_{6,1}$ versus $H_{6,4}$ case:

Comparison of computational costs $C_{6,1}$ and $C_{6,4}$ in (28) and (34) gives $E_{6,1} > E_{6,4}$ for all $m \geq 2$ and $l \geq 1$.

In addition to the above comparisons we compare the proposed methods $H_{6,1}$ and $H_{9,1}$ with each other.

$H_{9,1}$ versus $H_{6,1}$ case:

The particular boundary $R_{9,1,6,1} = 1$ expressed by μ_0 written as a function of μ_1 and m is

$$\mu_0 = \frac{1}{12} \frac{2(r-s)m^2 + 3m(2\mu_1(r-s) + 3l(r-s) + 3r - 23s) + 3l(r-5s) - 22s}{m(s-r) + s} \tag{37}$$

where $r = \ln(3)$ and $s = \ln(2)$. This function has a vertical asymptote for $m = s/(r-s) = 1.7095$.

Note that for $m \geq 44$, the numerator of (37) is positive and the denominator is negative, which shows that μ_0 is always negative for $m \geq 44$. That is, the boundary is out of admissible region for $m \geq 44$ and we have $E_{9,1} > E_{6,1} \forall (\mu_1, \mu_0) \in (0, +\infty) \times (0, +\infty)$ and $l \geq 1$.

We summarize the above results in following theorem:

Theorem 3. For all $\mu_0 > 0$, $\mu_1 > 0$ and $l \geq 1$ we have:

- (a) $E_{6,1} > E_{6,2} \forall m \geq 4$.
- (b) $E_{6,1} > E_{6,3}$ and $E_{6,1} > E_{6,4} \forall m \geq 2$.
- (c) $E_{9,1} > E_{6,1} \forall m \geq 44$.

Otherwise, the efficiency comparison depends on m , μ_0 , μ_1 and l .

4. Numerical Results

This section is devoted to check the effectiveness and efficiency of some of our proposed methods on different types of applications. In all cases, we apply our proposed schemes $H_{6,1}$ and $H_{9,1}$ and compare the results with those obtained by known methods $H_{6,2}$, $H_{6,3}$, and $H_{6,4}$. We consider academic examples, a special case of a nonlinear conservative problem which is transformed in a nonlinear system by approximating the derivatives by divided differences and, also the approximation of the solution of an elliptic partial differential equation that model a chemical problem.

All the experiments have been carried out in Matlab 2017 with variable precision arithmetics with 1000 digits of mantissa. These calculations have been made with an Intel Core processor i7-4700HQ with a CPU of 2.40GHz and 16.0 GB al RAM memory. In the tables we include the number of iterations (iter), the residual error of the corresponding function ($\|F(x^{(k+1)})\|$) in the last iterated and the difference between the two last iterates ($\|x^{(k+1)} - x^{(k)}\|$). We also present the approximated computational order of convergence (ACOC) defined in [14] with the expression

$$ACOC \approx \frac{\log \left(\|x^{(k+1)} - x^{(k)}\| / \|x^{(k)} - x^{(k-1)}\| \right)}{\log \left(\|x^{(k)} - x^{(k-1)}\| / \|x^{(k-1)} - x^{(k-2)}\| \right)}.$$

When the components of this vector are stable, it is an approximation of the theoretical order of convergence. In other case, it does not give us any information and we will denote it by $-$. Also the execution time (in seconds) has been calculated (by means of “cputime” Matlab command) with the mean value of 100 consecutive executions, for big-sized systems corresponding to Examples 1 to 3.

4.1. Example 1

We consider the case of a nonlinear conservative problem described by the differential equation

$$y''(t) + \Phi(y(t)) = 0, \quad t \in [0, 1],$$

with the boundary conditions

$$y(0) = y(1) = 0.$$

We transform this boundary problem into a system of nonlinear equations by approximating the second derivative by a divided difference of second order. We introduce points $t_i = 0 + ih$, $i = 0, 1, \dots, m + 1$, where $h = \frac{1}{m+1}$ and m is an appropriate positive integer. A scheme is then designed for the determination of numbers y_i as approximations of the solution $y(t)$ at point t_i . By using divided differences of second order

$$y''(t_i) \approx \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2},$$

we transform the boundary problem into the nonlinear system

$$y_{i+1} - 2y_i + y_{i-1} + h^2\Phi(y_i) = 0, \quad i = 1, 2, \dots, m. \tag{38}$$

Introducing vectors

$$y = (y_1, y_2, \dots, y_m)^T, \quad \Phi_y = (\Phi(y_1), \Phi(y_2), \dots, \Phi(y_m))^T$$

and the matrix of size $m \times m$

$$A = \begin{pmatrix} -2 & 1 & 0 & \dots & 0 \\ 1 & -2 & 1 & \dots & 0 \\ 0 & 1 & -2 & \dots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \dots & -2 \end{pmatrix},$$

system (38) can be written in the form

$$F(y) \equiv Ay + h^2\Phi_y = 0.$$

In this case, we choose the law $\Phi(y(t)) = 1 + y(t)^3$ for the heat generation in the boundary problem and we solve system (38) by using iterative methods $H_{6,1}$, $H_{6,2}$, $H_{6,3}$, $H_{6,4}$, and $H_{9,1}$. In all cases, we use as initial estimation $x^{(0)} = (0.5, 0.5, \dots, 0.5)^T$ and the stopping criterium $\|x^{(k+1)} - x^{(k)}\| < 10^{-100}$ or $\|F(x^{(k+1)})\| < 10^{-100}$. We can see in Table 1 the results obtained for $m = 20$ and $m = 50$.

There are no significant differences among the results obtained for different values of step h , i.e. for different sizes of the nonlinear system resulting from discretization. Let us remark that, for the lowest number of iterations, the best results in terms of lowest residuals have been obtained by methods $H_{9,1}$ and $H_{6,1}$.

Table 1. Numerical results for conservative boundary problem.

Method		H _{6,1}	H _{6,2}	H _{6,3}	H _{6,4}	H _{9,1}
m = 20	ACOC	5.5833	6.0210	-	-	6.2081
	iter	3	3	4	4	3
	$\ x^{(k+1)} - x^{(k)}\ $	2.78×10^{-35}	1.37×10^{-34}	2.69×10^{-98}	3.59×10^{-96}	8.63×10^{-59}
	$\ F(x^{(k+1)})\ $	6.10×10^{-125}	1.17×10^{-101}	4.55×10^{-229}	8.10×10^{-225}	1.87×10^{-210}
	CPU time (seconds)	0.025	0.024	0.028	0.026	0.022
m = 50	ACOC	3.1024	2.1869	-	-	6.0462
	iter	3	3	4	4	3
	$\ x^{(k+1)} - x^{(k)}\ $	3.81×10^{-34}	2.08×10^{-34}	6.93×10^{-97}	9.18×10^{-95}	2.24×10^{-57}
	$\ F(x^{(k+1)})\ $	2.76×10^{-121}	8.62×10^{-101}	1.03×10^{-225}	1.80×10^{-221}	7.16×10^{-206}
	CPU time (seconds)	0.042	0.042	0.043	0.042	0.044

4.2. Example 2

Let us consider the system of nonlinear equations

$$\sum_{j=1, j \neq i}^m x_j - e^{-x_i} = 0, \quad i = 1, 2, \dots, m,$$

for an arbitrary positive integer m . We solve this system with the same schemes as before, using as initial guess $x^{(0)} = (1, 1, \dots, 1)^T$ and two values for the size, $m = 20$ and $m = 50$, being the solution $\alpha = (0.05, 0.05, \dots, 0.05)^T$ and $\alpha = (0.02, 0.02, \dots, 0.02)^T$, respectively. The stopping criterium used is again $\|x^{(k+1)} - x^{(k)}\| < 10^{-100}$ or $\|F(x^{(k+1)})\| < 10^{-100}$, that is, the process finishes when one of them is satisfied. The obtained results are shown in Table 2.

Table 2. Numerical results for Example 2.

Method		H _{6,1}	H _{6,2}	H _{6,3}	H _{6,4}	H _{9,1}
m = 20	ACOC	5.9898	5.9078	5.9248	5.9442	8.4359
	iter	3	3	3	3	3
	$\ x^{(k+1)} - x^{(k)}\ $	3.10×10^{-45}	6.62×10^{-46}	1.67×10^{-46}	3.55×10^{-47}	8.19×10^{-78}
	$\ F(x^{(k+1)})\ $	3.45×10^{-155}	1.94×10^{-127}	1.24×10^{-128}	5.59×10^{-130}	6.49×10^{-271}
	CPU time (seconds)	0.031	0.028	0.030	0.028	0.029
m = 50	ACOC	4.3931	2.0177	5.8962	5.9425	7.0463
	iter	3	3	3	3	3
	$\ x^{(k+1)} - x^{(k)}\ $	1.04×10^{-49}	1.10×10^{-52}	2.90×10^{-53}	8.37×10^{-54}	2.50×10^{-83}
	$\ F(x^{(k+1)})\ $	9.16×10^{-170}	6.01×10^{-142}	4.15×10^{-143}	3.46×10^{-144}	5.37×10^{-289}
	CPU time (seconds)	0.060	0.058	0.058	0.061	0.059

In this case, all the methods use the same number of iterations to achieve the solution, but the lowest residual are those of proposed schemes H_{9,1} and H_{6,1}.

4.3. Example 3

Gas dynamics can be modeled by a boundary problem described by the following elliptic partial differential equation and the boundary conditions:

$$\begin{aligned} u_{xx} + u_{yy} &= u^3, & (x, y) &\in [0, 1] \times [0, 1], \\ u(x, 0) &= 2x^2 - x + 1, & u(x, 1) &= 2, \\ u(0, y) &= 2y^2 - y + 1, & u(1, y) &= 2. \end{aligned} \tag{39}$$

By using central divided differences and step $h = 1/5$ in both variables, this problem is discretized in the nonlinear system

$$F(x) = Ax + h^2\phi(x) - b = 0,$$

where

$$A = \begin{pmatrix} B & -I & 0 & 0 \\ -I & B & -I & 0 \\ 0 & -I & B & -I \\ 0 & 0 & -I & B \end{pmatrix}, B = \begin{pmatrix} 4 & -1 & 0 & 0 \\ -1 & 4 & -1 & 0 \\ 0 & -1 & 4 & -1 \\ 0 & 0 & -1 & 4 \end{pmatrix},$$

being I the identity matrix of size 4×4 , $\phi(x) = (x_1^3, x_2^3, \dots, x_{16}^3)^T$ and

$$b = \left(\frac{44}{25}, \frac{23}{25}, \frac{28}{25}, \frac{87}{25}, \frac{23}{25}, 0, 0, 2, \frac{28}{25}, 0, 0, 2, \frac{87}{25}, 2, 2, 4 \right)^T.$$

To solve problem (39) we have used as initial guess $x^{(0)} = (1, 1, \dots, 1)^T$. Also the stopping criteria $\|F(x^{(k+1)})\| < 10^{-100}$ or $\|x^{(k+1)} - x^{(k)}\| < 10^{-100}$ have been used and the process finishes when one of them is satisfied or the number of iterations reaches to 50.

The results are shown in Table 3. The first column shows the numerical aspects (approximated computational order of convergence ACOC, last difference between consecutive iterates $\|x^{(k+1)} - x^{(k)}\|$ and residual $\|F(x^{(k+1)})\|$) analyzed for the schemes used to solve the problem. In the rest of the columns we show the numerical results obtained by the methods $H_{6,1}$, $H_{6,2}$, $H_{6,3}$, $H_{6,4}$, and $H_{9,1}$.

Table 3. Numerical results for elliptic partial differential equation.

Method	$H_{6,1}$	$H_{6,2}$	$H_{6,3}$	$H_{6,4}$	$H_{9,1}$
ACOC	3.0100	2.0107	5.6132	5.8724	5.2651
iter	3	3	3	3	3
$\ x^{(k+1)} - x^{(k)}\ $	4.51×10^{-40}	1.82×10^{-48}	3.74×10^{-47}	3.56×10^{-46}	6.95×10^{-67}
$\ F(x^{(k+1)})\ $	6.27×10^{-138}	4.19×10^{-129}	1.67×10^{-126}	1.52×10^{-124}	2.45×10^{-234}
CPU time (seconds)	0.035	0.036	0.035	0.035	0.035

The low number of iterations needed justify that the ACOC does not approximate properly the theoretical order of convergence. The methods giving lowest exact error at the last iteration are again $H_{9,1}$ and $H_{6,1}$.

4.4. Example 4

Let us consider the nonlinear two-dimensional system $F(x_1, x_2) = 0$ with coordinate functions

$$\begin{aligned} f_1(x_1, x_2) &= \log(x_1^2) - 2 \log(\cos(x_2)), \\ f_2(x_1, x_2) &= x_1 \tan\left(\frac{x_1}{\sqrt{2}} + x_2\right) - \sqrt{2}. \end{aligned} \tag{40}$$

This system has two real solutions at points, approximately, $(0.954811, 0.301815)^T$ and $(-0.954811, -0.301815)^T$. By using $x^{(0)} = (1, 0.5)^T$ as initial estimation, we have obtained the results appearing in Table 4, in which the residuals $\|x^{(k+1)} - x^{(k)}\|$ and $\|F(x^{(k+1)})\|$ appear, for the three first iterations.

Table 4. Numerical results for Example 4.

	$H_{6,1}$	$H_{6,2}$	$H_{6,3}$	$H_{6,4}$	$H_{9,1}$
$\ x^{(1)} - x^{(0)}\ $	1.90×10^{-1}	1.55×10^{-1}	1.45×10^{-1}	1.27×10^{-1}	2.01×10^{-1}
$\ x^{(2)} - x^{(1)}\ $	1.44×10^{-2}	5.69×10^{-2}	7.36×10^{-2}	1.18×10^{-1}	2.62×10^{-3}
$\ x^{(3)} - x^{(2)}\ $	1.07×10^{-9}	5.15-7	1.25×10^{-5}	1.74×10^{-4}	1.69×10^{-18}
$\ F(x^{(1)})\ $	4.12×10^{-2}	1.21×10^{-1}	1.41×10^{-1}	1.85×10^{-1}	7.56×10^{-3}
$\ F(x^{(2)})\ $	2.41×10^{-9}	2.65×10^{-6}	3.05×10^{-5}	2.99×10^{-4}	6.77×10^{-18}
$\ F(x^{(3)})\ $	3.21×10^{-44}	1.54×10^{-23}	6.19×10^{-20}	6.93×10^{-16}	5.39×10^{-86}

It can be observed that, although the difference between two consecutive iterations is not very small, the precision in the estimation of the root is very high, being the best for proposed schemes $H_{6,1}$ and $H_{9,1}$.

Moreover, dynamical planes help us to get global information about the convergence process. In Figure 1 we can see the dynamical planes of the proposed methods $H_{6,1}$ and $H_{9,1}$ and known schemes $H_{6,2}$, $H_{6,3}$, and $H_{6,4}$ when they are applied on system $F(x_1, x_2) = 0$. This figures are obtained by using the routines described in [15]. To draw these images, a mesh of 400×400 initial points has been used, 80 was the maximum number of iterations involved and 10^{-3} the tolerance used as the stopping criterium. In this paper, we have used a white star to show the roots of the nonlinear system. A color is assigned to each initial estimation (each point of the mesh) depending on where they converge to: Blue and orange correspond to the basins of the roots of the system (40) (brighter as lower is the number of iterations needed to converge) and black color is adopted when the maximum number of iterations is reached or the process diverges.

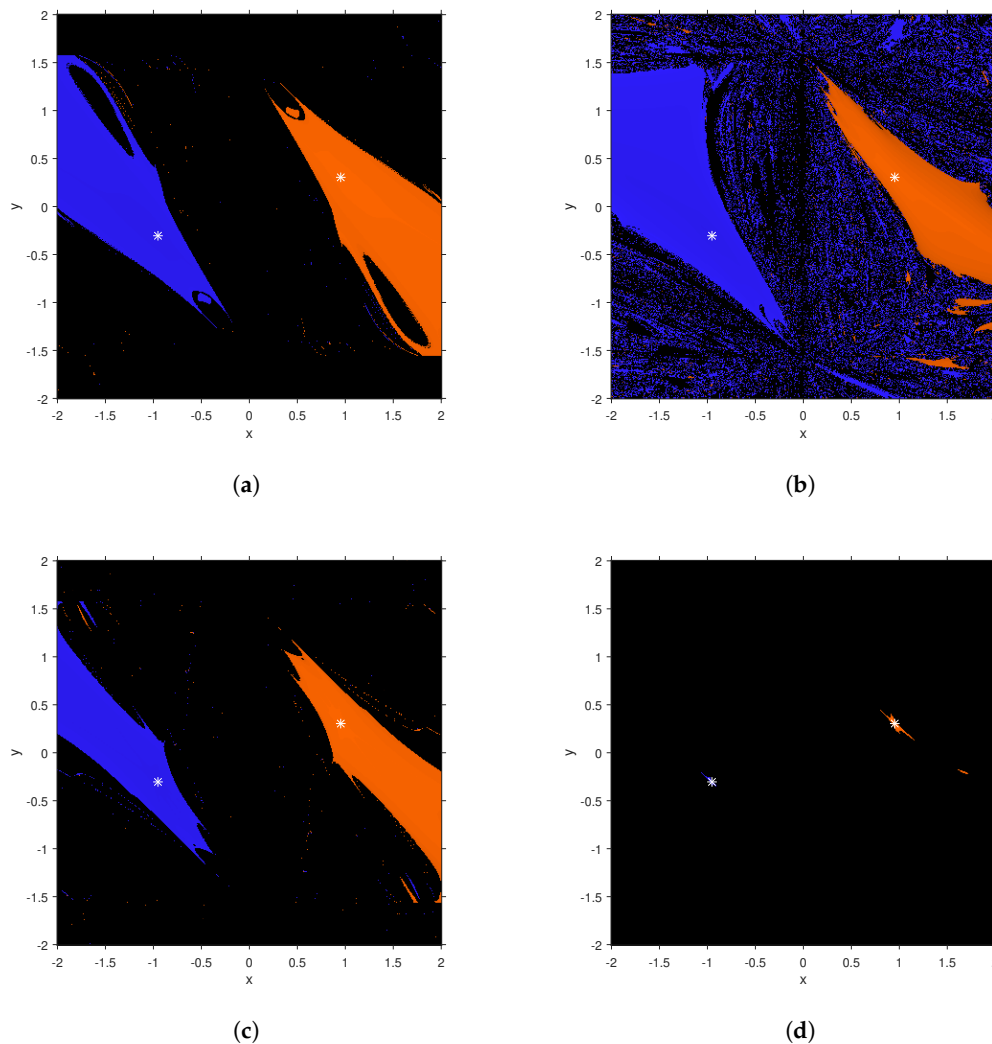
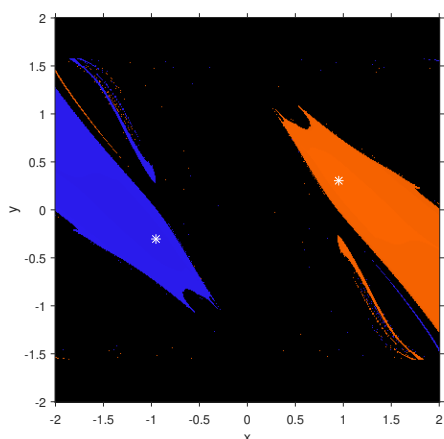


Figure 1. Cont.



(e)

Figure 1. Basins of attraction of known and proposed methods on $F(x_1, x_2) = 0$. (a) $H_{6,1}$; (b) $H_{6,2}$; (c) $H_{6,3}$; (d) $H_{6,4}$; (e) $H_{9,1}$.

In Figure 1, the shape and wideness of the basins of attraction show that $H_{6,1}$, $H_{6,2}$, $H_{6,3}$, and $H_{6,4}$ can find any of both roots by using a great amount of initial estimations, some of them far from the roots. However, $H_{6,4}$ is hardly able to find the roots and their basins are very small.

4.5. Example 5

Let us consider the nonlinear two-dimensional system $G(x_1, x_2) = 0$ with coordinate functions

$$g_1(x_1, x_2) = x_1^2 + x_2^2 - 1, \tag{41}$$

$$g_2(x_1, x_2) = x_1^2 - x_2^2 + \frac{1}{2}. \tag{42}$$

This system has four real solutions at points $\left(\frac{1}{2}, \frac{\sqrt{3}}{2}\right)^T$, $\left(\frac{1}{2}, -\frac{\sqrt{3}}{2}\right)^T$, $\left(-\frac{1}{2}, \frac{\sqrt{3}}{2}\right)^T$, and $\left(-\frac{1}{2}, -\frac{\sqrt{3}}{2}\right)^T$. Let us remark that they are symmetric as the system shows the intersection between two conical curves, an ellipse and a circumference. By using $x^{(0)} = (1, 1)^T$ as initial estimation, we have obtained the results appearing in Table 5, in which the residuals $\|x^{(k+1)} - x^{(k)}\|$ and $\|F(x^{(k+1)})\|$ also appear, for the three first iterations.

Table 5. Numerical results for Example 5.

	$H_{6,1}$	$H_{6,2}$	$H_{6,3}$	$H_{6,4}$	$H_{9,1}$
$\ x^{(1)} - x^{(0)}\ $	5.10×10^{-1}	5.15×10^{-1}	5.125×10^{-1}	5.10×10^{-1}	5.16×10^{-1}
$\ x^{(2)} - x^{(1)}\ $	7.96×10^{-3}	2.38×10^{-3}	5.63×10^{-3}	8.30×10^{-3}	1.46×10^{-3}
$\ x^{(3)} - x^{(2)}\ $	6.03×10^{-12}	3.54×10^{-16}	3.60×10^{-13}	8.89×10^{-12}	1.14×10^{-23}
$\ F(x^{(1)})\ $	1.13×10^{-2}	3.37×10^{-3}	8.00×10^{-3}	1.18×10^{-2}	2.07×10^{-3}
$\ F(x^{(2)})\ $	8.53×10^{-12}	5.00×10^{-16}	5.10×10^{-13}	1.26×10^{-11}	1.61×10^{-23}
$\ F(x^{(3)})\ $	2.56×10^{-56}	8.87×10^{-62}	8.99×10^{-57}	5.02×10^{-54}	6.87×10^{-161}

The numerical results appearing in Table 5 show as the estimation of the roots has similar errors in all sixth order methods, although $H_{6,2}$ and $H_{6,3}$ are slightly better than $H_{6,1}$. Indeed, the ninth order scheme has the best precision in the approximation of the roots of $G(x_1, x_2)$, as corresponds to its high order of convergence. In terms of computational time, the mean of one hundred consecutive iterations

have been used in order to get good estimations of the real time and, in case of proposed methods, they use similar times as the existing schemes getting better accuracy.

In Figure 2 the colors blue, orange, green and purple correspond to the basins of the roots of the system (41). We can see the dynamical planes of the proposed method methods $H_{6,1}$ and $H_{9,1}$ and known schemes $H_{6,2}$, $H_{6,3}$, and $H_{6,4}$ when they are applied on system $G(x_1, x_2) = 0$.

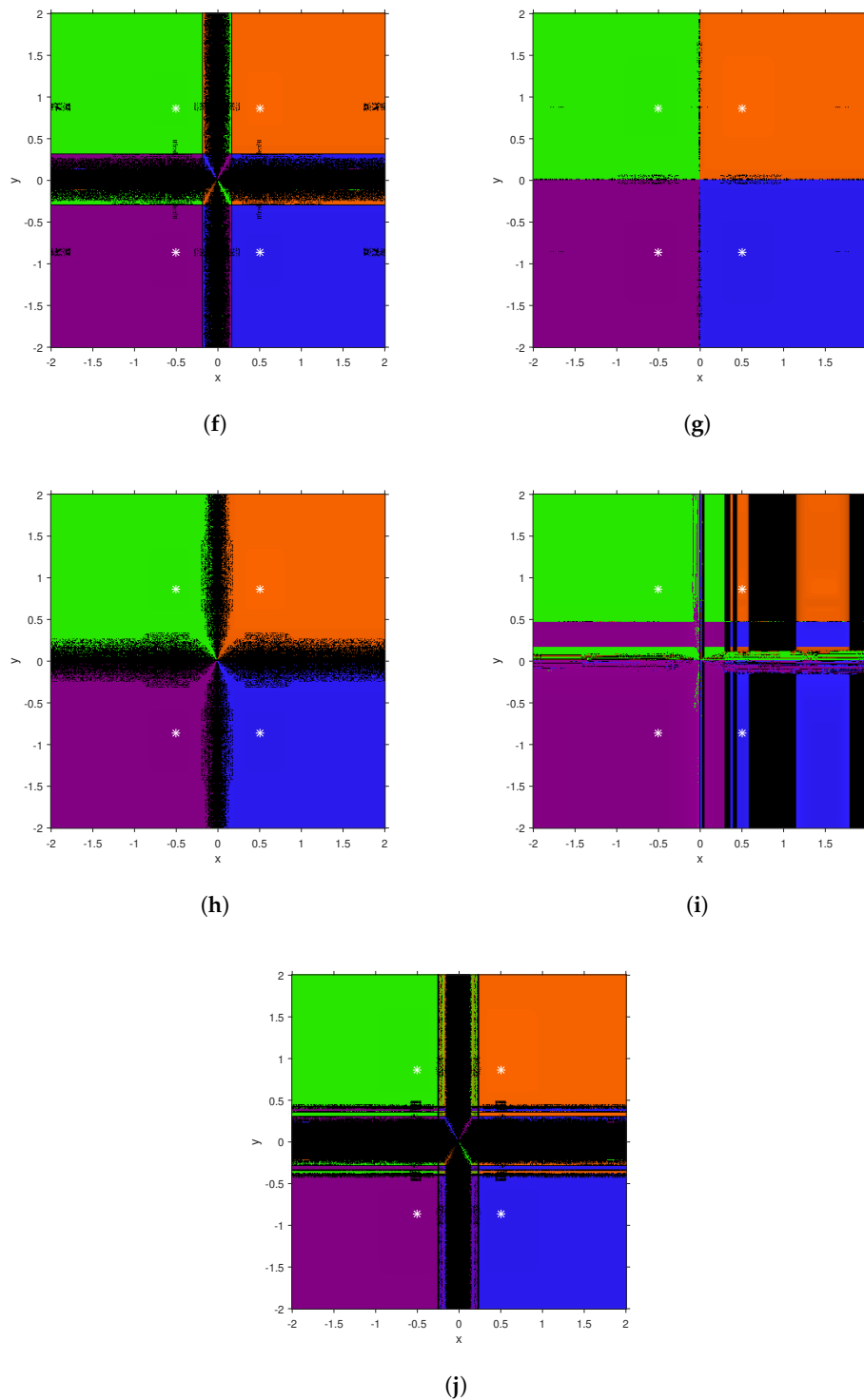


Figure 2. Basins of attraction of known and proposed methods on $G(x_1, x_2) = 0$. (a) $H_{6,1}$; (b) $H_{6,2}$; (c) $H_{6,3}$; (d) $H_{6,4}$; (e) $H_{9,1}$.

Regarding the stability of proposed and known iterative processes on $G(x_1, x_2)$, it can be observed in Figure 2 that, in all cases except $H_{6,4}$, the connected component of the basins of attraction that holds the root (usually known as immediate basin of attraction) is very wide. For all iterative methods black areas of low convergence or divergence appear, being more diffuse in case of $H_{6,2}$ and wider in $H_{6,4}$. Methods $H_{6,1}$, $H_{6,3}$ and $H_{9,1}$ present black areas that are mainly regions of slower convergence, that would be colored if a higher number of iterations is fixed as a maximum; meanwhile, the biggest black area of the dynamical plane associated to method $H_{6,4}$ corresponds to divergence.

5. Concluding Remarks

From Potra–Pták third order scheme, an efficient sixth order method for solving nonlinear system is proposed. Moreover, it has been extended by adding subsequent steps with the same structure that adds a new functional evaluation per new step. Then the order of the resulting procedure increases in three units per step. The proposed methods have been shown to be more efficient than several known methods of the same order. Some numerical tests with academic and real-life problems have been made to check the efficiency and applicability of the designed procedures. The numerical performance have been opposed to the stability shown by the dynamical planes of new and known methods on two-dimensional systems.

Author Contributions: Methodology, H.A.; Writing—original draft, J.R.T.; Writing—review & editing, A.C.

Funding: This research was partially supported by Ministerio de Economía y Competitividad under grants MTM2014-52016-C2-2-P and Generalitat Valenciana PROMETEO/2016/089.

Acknowledgments: The authors are grateful to the anonymous referees for their valuable comments and suggestions to improve the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ostrowski, A.M. *Solution of Equations and Systems of Equations*; Academic Press: New York, NY, USA, 1966.
2. Ortega, J.M.; Rheinboldt, W.C. *Iterative Solution of Nonlinear Equations in Several Variables*; Academic Press: New York, NY, USA, 1970.
3. Kelley, C.T. *Solving Nonlinear Equations with Newton's Method*; SIAM: Philadelphia, PA, USA, 2003.
4. Traub, J.F. *Iterative Methods for the Solution of Equations*; Prentice-Hall: Englewood Cliffs, NJ, USA, 1964.
5. Homeier, H.H.H. A modified Newton method with cubic convergence: The multivariable case. *J. Comput. Appl. Math.* **2004**, *169*, 161–169. [[CrossRef](#)]
6. Darvishi, M.T.; Barati, A. A fourth-order method from quadrature formulae to solve systems of nonlinear equations. *Appl. Math. Comput.* **2007**, *188*, 257–261. [[CrossRef](#)]
7. Cordero, A.; Hueso, J.L.; Martínez, E.; Torregrosa, J.R. A modified Newton-Jarratt's composition. *Numer. Algor.* **2010**, *55*, 87–99. [[CrossRef](#)]
8. Cordero, A.; Hueso, J.L.; Martínez, E.; Torregrosa, J.R. Efficient high-order methods based on golden ratio for nonlinear systems. *Appl. Math. Comput.* **2011**, *217*, 4548–4556. [[CrossRef](#)]
9. Grau-Sánchez, M.; Grau, À.; Noguera, M. On the computational efficiency index and some iterative methods for solving systems of nonlinear equations. *J. Comput. Appl. Math.* **2011**, *236*, 1259–1266. [[CrossRef](#)]
10. Grau-Sánchez, M.; Grau, À.; Noguera, M. Ostrowski type methods for solving systems of nonlinear equations. *Appl. Math. Comput.* **2011**, *218*, 2377–2385. [[CrossRef](#)]
11. Potra, F.-A.; Pták, V. *Nondiscrete Induction and Iterative Processes*; Pitman Publishing: Boston, MA, USA, 1984.
12. Grau-Sánchez, M.; Noguera, M.; Amat, S. On the approximation of derivatives using divided difference operators preserving the local convergence order of iterative methods. *J. Comput. Appl. Math.* **2013**, *237*, 363–372. [[CrossRef](#)]

13. Sharma, J.R.; Arora, H. On efficient weighted-Newton methods for solving systems of nonlinear equations. *Appl. Math. Comput.* **2013**, *222*, 497–506. [[CrossRef](#)]
14. Cordero, A.; Torregrosa, J. R. Variants of Newton's method using fifth order quadrature formulas. *Appl. Math. Comput.* **2007**, *190*, 686–698. [[CrossRef](#)]
15. Chicharro, F.I.; Cordero A.; Torregrosa, J.R. Drawing dynamical and parameter planes of iterative families and methods. *Sci. World J.* **2013**, *2013*, 780153. [[CrossRef](#)] [[PubMed](#)]



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).