

First-principles Derivation of a Third-order Method for Solving a Two-dimensional Nonlinear System

J.S.C. Prentice

Department of Applied Mathematics, University of Johannesburg

P.O. Box 524, Auckland Park, 2006, South Africa

Tel: 271-1559-3145 E-mail: jprentice@uj.ac.za

Abstract

In this paper, we adopt a ‘first-principles’ approach to deriving a cubically convergent unipoint iterative method for a two-dimensional system of nonlinear equations. We demand that the Jacobian and Hessian of an iteration function be identically zero at the fixed point, and these conditions allow us to determine various terms in the iteration function. We present analytical expressions for the inverses of two matrices appearing in the algorithm, which allows the iteration function to be written explicitly. We demonstrate the cubic convergence rate by means of a few numerical examples, and we determine the asymptotic error constant for these examples.

Keywords: Iterative method, Cubic convergence, Fixed point, Two dimensional system

1. Introduction

In this paper, we intend to derive an iterative method for a two-dimensional real-valued nonlinear system that has cubic convergence for simple roots. Many papers derive such methods based on a multipoint approach, wherein the method is presented *ab initio*, and then shown to be of cubic order (Amat & Busquier, 2007; Cordero et al, 2009; Darvishi & Barati, 2007; Hueso et al, 2007 & 2009; Kou, 2007; Nedzhibov, 2008; Traub, 1964; and references therein). Our approach will be different: we will define an iteration function $\mathbf{G}$, and by imposing conditions appropriate for cubic convergence, we will construct the method. We have dubbed this a ‘first-principles’ approach. Furthermore, we will find analytical expressions for any inverted matrices present in the algorithm, so that we will be able to write $\mathbf{G}$ in an explicit form that requires no matrix inversion at all. We emphasize that our approach is a unipoint approach, so that $\mathbf{G}$ is written explicitly in terms of the previous iterate. Lastly, we demonstrate the algorithm by means of a few numerical examples, and we suggest a few applications. Our paper has a tutorial quality; as such, it represents a contribution to mathematical pedagogy in the field of numerical methods.

2. Theory

Let

$$\mathbf{p} = \begin{bmatrix} p_1 \\ p_2 \end{bmatrix} \quad (1)$$

denote the solution of the system

$$\begin{aligned} f_1(x_1, x_2) &= 0 \\ f_2(x_1, x_2) &= 0 \end{aligned} \quad (2)$$

Define

$$\mathbf{G}(\mathbf{x}) = \begin{bmatrix} g_1(x_1, x_2) \\ g_2(x_1, x_2) \end{bmatrix} = \mathbf{x} + \mathbf{A}\mathbf{F}(\mathbf{x}) + \mathbf{B}\mathbf{F}^2(\mathbf{x}) \quad (3)$$

where

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad \mathbf{F}(\mathbf{x}) = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} \quad \mathbf{F}^2(\mathbf{x}) = \begin{bmatrix} f_1 f_1 \\ f_1 f_2 \\ f_2 f_1 \\ f_2 f_2 \end{bmatrix} \quad (4)$$

and $\mathbf{A}$ and $\mathbf{B}$ are matrices to be determined. Intuitively, one might expect that $f_2 f_1$ should also appear in $\mathbf{F}^2$; we will see later that this would lead to a singularity. Indeed, since $f_2 f_1 = f_1 f_2$ (since f_1 and f_2 are real and continuous), the inclusion of $f_2 f_1$ in $\mathbf{F}^2$ is superfluous, at the very least.

We intend to use $\mathbf{G}(\mathbf{x})$ in functional iteration (note that $\mathbf{G}(\mathbf{p}) = \mathbf{p}$), so consider the following Taylor expansion:

$$\mathbf{G}(\mathbf{x}) = \mathbf{G}(\mathbf{p} + \mathbf{e}) = \mathbf{G}(\mathbf{p}) + \mathbf{G}'(\mathbf{p})\mathbf{e} + \frac{1}{2}\mathbf{G}''(\mathbf{p})\mathbf{e}^2 + h.o.t. \quad (5)$$

where

$$\mathbf{e} = \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} = \begin{bmatrix} x_1 - p_1 \\ x_2 - p_2 \end{bmatrix} \quad \mathbf{e}^2 = \begin{bmatrix} e_1 e_1 \\ e_1 e_2 \\ e_2 e_1 \\ e_2 e_2 \end{bmatrix} \quad (6)$$

$$\mathbf{G}'(\mathbf{p}) = \begin{bmatrix} g_1^1 & g_1^2 \\ g_2^1 & g_2^2 \end{bmatrix}_{\mathbf{x}=\mathbf{p}} \quad (7)$$

$$\mathbf{G}''(\mathbf{p}) = \begin{bmatrix} g_{11}^{11} & g_{11}^{12} & g_{11}^{21} & g_{11}^{22} \\ g_{12}^{11} & g_{12}^{12} & g_{12}^{21} & g_{12}^{22} \\ g_{21}^{11} & g_{21}^{12} & g_{21}^{21} & g_{21}^{22} \\ g_{22}^{11} & g_{22}^{12} & g_{22}^{21} & g_{22}^{22} \end{bmatrix}_{\mathbf{x}=\mathbf{p}} \quad (8)$$

and we have used the notation

$$g_i^j = \frac{\partial g_i}{\partial x_j} \quad g_i^{jk} = \frac{\partial^2 g_i}{\partial x_k \partial x_j} \quad (9)$$

in which $i, j, k = 1, 2$. We refer to $\mathbf{G}'$ as the Jacobian of $\mathbf{G}$, and $\mathbf{G}''$ as the Hessian of $\mathbf{G}$.

Functional iteration on $\mathbf{G}$ implies

$$\begin{aligned} \mathbf{x}_{t+1} &= \mathbf{G}(\mathbf{x}_t) \\ \Rightarrow \mathbf{e}_{t+1} = \mathbf{x}_{t+1} - \mathbf{p} &= \mathbf{G}(\mathbf{x}_t) - \mathbf{p} \\ &= \mathbf{G}(\mathbf{p} + \mathbf{e}_t) - \mathbf{p} \\ &= \mathbf{G}'(\mathbf{p})\mathbf{e}_t + \frac{1}{2}\mathbf{G}''(\mathbf{p})\mathbf{e}_t^2 + O(\mathbf{e}_t^3). \end{aligned} \quad (10)$$

since $\mathbf{G}(\mathbf{p}) = \mathbf{p}$, and where t denotes the iteration count. Clearly, for cubic convergence, we require

$$\begin{aligned} \mathbf{G}'(\mathbf{p}) &= \mathbf{0} \\ \mathbf{G}''(\mathbf{p}) &= \mathbf{0} \end{aligned} \quad (11)$$

The first of these conditions leads to the well-known result

$$\mathbf{A} = -\mathbf{J}^{-1} \quad (12)$$

where $\mathbf{J}$ is the Jacobian of $\mathbf{F}$, so that

$$\mathbf{G}(\mathbf{x}) = \mathbf{x} - \mathbf{J}^{-1}\mathbf{F}(\mathbf{x}) + \mathbf{B}\mathbf{F}^2(\mathbf{x}). \quad (13)$$

The second condition $\mathbf{G}''(\mathbf{p}) = \mathbf{0}$ enables us to determine $\mathbf{B}$, and such analysis is the subject of this section.

Now, since $\mathbf{G}$ is 2×1 and $\mathbf{F}^2$ is 3×1 , we must have that $\mathbf{B}$ is 2×3 , i.e. $\mathbf{B}$ has six elements. This means that we need to impose six independent conditions in order to determine these elements. The condition $\mathbf{G}''(\mathbf{p}) = \mathbf{0}$ implies

$$\begin{bmatrix} g_{11}^{11} & g_{11}^{12} & g_{11}^{21} & g_{11}^{22} \\ g_{12}^{11} & g_{12}^{12} & g_{12}^{21} & g_{12}^{22} \\ g_{21}^{11} & g_{21}^{12} & g_{21}^{21} & g_{21}^{22} \\ g_{22}^{11} & g_{22}^{12} & g_{22}^{21} & g_{22}^{22} \end{bmatrix}_{\mathbf{x}=\mathbf{p}} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (14)$$

which provides eight independent conditions, two too many. However, assuming that g_1 and g_2 are continuous functions of x_1 and x_2 , we have that

$$\begin{aligned} g_{12}^{12} &= g_{11}^{21} \\ g_{21}^{12} &= g_{22}^{21} \end{aligned} \quad (15)$$

so that we actually only have six independent conditions

$$\begin{bmatrix} g_{11}^{11} & g_{11}^{12} & g_{11}^{22} \\ g_{12}^{11} & g_{12}^{12} & g_{12}^{22} \\ g_{21}^{11} & g_{21}^{12} & g_{21}^{22} \\ g_{22}^{11} & g_{22}^{12} & g_{22}^{22} \end{bmatrix}_{\mathbf{x}=\mathbf{p}} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \quad (16)$$

So, we have

$$\begin{bmatrix} g_1 \\ g_2 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \end{bmatrix} \begin{bmatrix} f_1 f_1 \\ f_1 f_2 \\ f_2 f_2 \end{bmatrix} \quad (17)$$

which gives

$$\begin{aligned} g_1 &= x_1 + a_{11}f_1 + a_{12}f_2 + b_{11}f_1f_1 + b_{12}f_1f_2 + b_{13}f_2f_2 \\ g_2 &= x_2 + a_{21}f_1 + a_{22}f_2 + b_{21}f_1f_1 + b_{22}f_1f_2 + b_{23}f_2f_2 \end{aligned} \quad (18)$$

Computing the relevant second-order derivatives and evaluating at $\mathbf{p}$, and imposing the conditions (16), yields

$$0 = g_1^{11} = (a_{11}^1 f_1^1 + a_{11}^1 f_1^1 + a_{12}^1 f_2^1 + a_{12}^1 f_2^1) + [a_{11} f_1^{11} + a_{12} f_2^{11}] + 2b_{11} f_1^1 f_1^1 + b_{12} (f_1^1 f_2^1 + f_1^1 f_2^1) + 2b_{13} f_2^1 f_2^1 \quad (19)$$

$$0 = g_1^{12} = (a_{11}^1 f_1^2 + a_{11}^2 f_1^1 + a_{12}^1 f_2^2 + a_{12}^2 f_2^1) + [a_{11} f_1^{12} + a_{12} f_2^{12}] + 2b_{11} f_1^2 f_1^1 + b_{12} (f_1^1 f_2^2 + f_1^2 f_2^1) + 2b_{13} f_2^2 f_2^1 \quad (20)$$

$$0 = g_1^{22} = (a_{11}^2 f_1^2 + a_{11}^2 f_1^2 + a_{12}^2 f_2^2 + a_{12}^2 f_2^2) + [a_{11} f_1^{22} + a_{12} f_2^{22}] + 2b_{11} f_1^2 f_1^2 + b_{12} (f_1^2 f_2^2 + f_1^2 f_2^2) + 2b_{13} f_2^2 f_2^2 \quad (21)$$

$$0 = g_2^{11} = (a_{21}^1 f_1^1 + a_{21}^1 f_1^1 + a_{22}^1 f_2^1 + a_{22}^1 f_2^1) + [a_{21} f_1^{11} + a_{22} f_2^{11}] + 2b_{21} f_1^1 f_1^1 + b_{22} (f_1^1 f_2^1 + f_1^1 f_2^1) + 2b_{23} f_2^1 f_2^1 \quad (22)$$

$$0 = g_2^{12} = (a_{21}^1 f_1^2 + a_{21}^2 f_1^1 + a_{22}^1 f_2^2 + a_{22}^2 f_2^1) + [a_{21} f_1^{12} + a_{22} f_2^{12}] + 2b_{21} f_1^2 f_1^1 + b_{22} (f_1^1 f_2^2 + f_1^2 f_2^1) + 2b_{23} f_2^2 f_2^1 \quad (23)$$

$$0 = g_2^{22} = (a_{21}^2 f_1^2 + a_{21}^2 f_1^2 + a_{22}^2 f_2^2 + a_{22}^2 f_2^2) + [a_{21} f_1^{22} + a_{22} f_2^{22}] + 2b_{21} f_1^2 f_1^2 + b_{22} (f_1^2 f_2^2 + f_1^2 f_2^2) + 2b_{23} f_2^2 f_2^2 \quad (24)$$

It is a consequence of the evaluation at $\mathbf{p}$ that all terms proportional to either f_1 or f_2 have vanished (since $f_{1,2}(\mathbf{p}) = 0$).

Now, if we define the matrices

$$\mathbf{L}_1 = \begin{bmatrix} f_1^1 & f_1^2 & f_2^1 & f_2^2 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \mathbf{L}_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ f_1^1 & f_1^2 & f_2^1 & f_2^2 \end{bmatrix} \quad (25)$$

$$\mathbf{Z}_1 = \begin{bmatrix} 2a_{11}^1 & a_{11}^2 & 0 \\ 0 & a_{11}^1 & 2a_{11}^2 \\ 2a_{12}^1 & a_{12}^2 & 0 \\ 0 & a_{12}^1 & 2a_{12}^2 \end{bmatrix} \quad \mathbf{Z}_2 = \begin{bmatrix} 2a_{21}^1 & a_{21}^2 & 0 \\ 0 & a_{21}^1 & 2a_{21}^2 \\ 2a_{22}^1 & a_{22}^2 & 0 \\ 0 & a_{22}^1 & 2a_{22}^2 \end{bmatrix} \quad (26)$$

$$\mathbf{H} = \begin{bmatrix} f_1^{11} & f_1^{12} & f_1^{22} \\ f_2^{11} & f_2^{12} & f_2^{22} \end{bmatrix} \quad \mathbf{M} = \begin{bmatrix} 2f_1^1 f_1^1 & 2f_1^2 f_1^1 & 2f_1^1 f_2^1 \\ 2f_1^1 f_2^1 & f_1^1 f_2^2 + f_1^2 f_2^1 & 2f_1^1 f_2^2 \\ 2f_2^1 f_2^1 & 2f_2^2 f_2^1 & 2f_2^2 f_2^2 \end{bmatrix}, \quad (27)$$

equations (19)–(24) can be written as

$$\mathbf{L}_1 \mathbf{Z}_1 + \mathbf{L}_2 \mathbf{Z}_2 - \mathbf{J}^{-1} \mathbf{H} + \mathbf{B} \mathbf{M} = \mathbf{0} \\ \Rightarrow \mathbf{B} = -(\mathbf{L}_1 \mathbf{Z}_1 + \mathbf{L}_2 \mathbf{Z}_2 - \mathbf{J}^{-1} \mathbf{H}) \mathbf{M}^{-1}. \quad (28)$$

This gives

$$\mathbf{G}(\mathbf{x}) = \mathbf{x} - \mathbf{J}^{-1} \mathbf{F}(\mathbf{x}) - (\mathbf{L}_1 \mathbf{Z}_1 + \mathbf{L}_2 \mathbf{Z}_2 - \mathbf{J}^{-1} \mathbf{H}) \mathbf{M}^{-1} \mathbf{F}^2(\mathbf{x}). \quad (29)$$

Moreover, analytical expressions for $\mathbf{J}^{-1}$ and $\mathbf{M}^{-1}$ are easily determined:

$$\mathbf{J}^{-1} = \frac{1}{(f_1^1 f_2^2 - f_1^2 f_2^1)} \begin{bmatrix} f_2^2 & -f_1^2 \\ -f_2^1 & f_1^1 \end{bmatrix} \quad (30)$$

$$\mathbf{M}^{-1} = \frac{1}{4(f_1^1 f_2^2 - f_1^2 f_2^1)^2} \begin{bmatrix} 2f_2^2 f_2^2 & -4f_1^2 f_2^2 & 2f_1^2 f_2^1 \\ -4f_2^2 f_2^1 & 4(f_1^1 f_2^2 + f_1^2 f_2^1) & -4f_1^1 f_2^1 \\ 2f_2^1 f_2^1 & -4f_1^2 f_2^1 & 2f_1^1 f_2^1 \end{bmatrix} \quad (31)$$

Note that $\mathbf{J}^{-1}$ and $\mathbf{M}^{-1}$ are singular only when

$$\det(\mathbf{J}) = f_1^1 f_2^2 - f_1^2 f_2^1 = 0. \quad (32)$$

At this juncture we can comment about the exclusion of the term $f_2 f_1$ from $\mathbf{F}^2$, as mentioned earlier. Had $f_2 f_1$ been included (so that $\mathbf{F}^2$ was a 4×1 vector), the matrix $\mathbf{M}$ would have had the form

$$\mathbf{M} = \begin{bmatrix} 2f_1^1 f_1^1 & 2f_1^2 f_1^1 & 2f_2^2 f_2^1 & 2f_1^2 f_2^1 \\ 2f_1^1 f_2^1 & f_1^1 f_2^2 + f_1^2 f_2^1 & f_2^1 f_2^2 + f_2^2 f_1^1 & 2f_1^2 f_2^2 \\ 2f_2^1 f_1^1 & f_2^1 f_2^2 + f_2^2 f_1^1 & f_1^1 f_2^2 + f_1^2 f_2^1 & 2f_2^2 f_1^1 \\ 2f_2^1 f_2^1 & 2f_2^2 f_2^1 & 2f_1^2 f_1^1 & 2f_2^2 f_2^2 \end{bmatrix} \quad (33)$$

and, since the second and third rows are identical, this matrix is singular.

It is worth checking to see that this method reduces to the expected form for the scalar case. In the scalar case we have

$$\begin{aligned} \mathbf{x} &= x & \mathbf{L}_1 &= f' \\ \mathbf{F}(\mathbf{x}) &= f(x) & \mathbf{L}_2 &= f' \\ \mathbf{F}^2(\mathbf{x}) &= f^2 & \mathbf{Z}_1 &= 2 \frac{d}{dx} \left(-\frac{1}{f'} \right) = \frac{2f''}{[f']^2} \\ \mathbf{J} &= f' & \mathbf{Z}_2 &= 0 \\ \mathbf{J}^{-1} &= \frac{1}{f'} & \mathbf{H} &= f'' \\ \mathbf{A} &= -\mathbf{J}^{-1} = -\frac{1}{f'} & \mathbf{M} &= 2[f']^2 \\ & & \mathbf{M}^{-1} &= \frac{1}{2[f']^2} \end{aligned} \quad (34)$$

and so

$$\begin{aligned} G(x) &= x - \frac{f}{f'} - \left(\frac{2f'f''}{[f']^2} - \frac{f''}{f'} \right) \frac{f^2}{2[f']^2} \\ &= x - \frac{f}{f'} - \frac{f''f^2}{2[f']^3} \end{aligned} \quad (35)$$

as expected (see Schröder's formula in Traub, 1964). In the above, $\mathbf{Z}_2 = 0$ because $\mathbf{Z}_2$ is composed of elements from the second row of $\mathbf{A}$ but, in the scalar case, there are no such elements.

3. Implementation

Given the analytical expressions derived above, the implementation of the algorithm is clear: using $f_1(x_1, x_2)$ and $f_2(x_1, x_2)$, we construct an explicit expression for $\mathbf{G}(x_1, x_2)$, which is then used for functional iteration. In this way, there is no need to explicitly invert matrices, or to evaluate functions through the passing of arguments. The function $\mathbf{G}$ can be constructed using computer algebra software, as we have done in our own work, and thereafter functional iteration is applied numerically.

4. Limitation

We acknowledge a limitation on the quality of this method - the fact that the matrix $\mathbf{M}$ scales rapidly with the dimensionality of the system. As the dimension increases, so does the size of $\mathbf{M}$. For example, a three-dimensional system requires that $\mathbf{M}$ be 6×6 , and a four-dimensional system requires that $\mathbf{M}$ be 10×10 . In general, for a d -dimensional system, the Jacobian $\mathbf{J}$ is $d \times d$, and $\mathbf{M}$ is $\frac{d(d+1)}{2} \times \frac{d(d+1)}{2}$. Note that the dimensions of $\mathbf{M}$ are dictated by those of $\mathbf{F}^2$: for a d -dimensional system there are $\frac{d(d+1)}{2}$ elements in $\mathbf{F}^2$ since, if the term $f_i f_j$ is in $\mathbf{F}^2$, then $f_j f_i$ is not (refer to our earlier discussion regarding $f_2 f_1$). By contrast, in the method of Kou (Kou, 2007), no matrix exceeds $d \times d$, although admittedly Kou's method is a multipoint method - an observation that certainly informs the superiority of multipoint methods over unipoint methods, in terms of computational efficiency.

5. Numerical examples

As an example, consider

$$\begin{aligned} f_1(x_1, x_2) &= x_1 x_2 + x_2 - 4 \\ f_2(x_1, x_2) &= x_1^2 - x_2^2 + 3 \end{aligned} \quad (36)$$

which has solution

$$\mathbf{p} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}. \quad (37)$$

Iteration with initial value $\mathbf{x}_0 = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$ gives

$$\begin{aligned} \mathbf{e}_1 &= \begin{bmatrix} 0.724 \times 10^{-1} \\ 0.298 \times 10^{-1} \end{bmatrix}, \mathbf{e}_2 = \begin{bmatrix} 0.419 \times 10^{-4} \\ -0.323 \times 10^{-4} \end{bmatrix} \\ \mathbf{e}_3 &= \begin{bmatrix} -0.921 \times 10^{-14} \\ 0.799 \times 10^{-14} \end{bmatrix}, \mathbf{e}_4 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \end{aligned} \quad (38)$$

and with initial value $\mathbf{x}_0 = \begin{bmatrix} 0.5 \\ 2.5 \end{bmatrix}$ gives

$$\begin{aligned} \mathbf{e}_1 &= \begin{bmatrix} -0.266 \times 10^{-2} \\ -0.204 \times 10^{-1} \end{bmatrix}, \mathbf{e}_2 = \begin{bmatrix} 0.866 \times 10^{-6} \\ -0.437 \times 10^{-6} \end{bmatrix} \\ \mathbf{e}_3 &= \begin{bmatrix} -0.726 \times 10^{-19} \\ 0.122 \times 10^{-19} \end{bmatrix}, \mathbf{e}_4 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \end{aligned} \quad (39)$$

which clearly indicates cubic convergence. The value for $\mathbf{e}_4$ indicates that the error is smaller than the machine precision. Now, since $\mathbf{G}'(\mathbf{p}) = \mathbf{0}$ and $\mathbf{G}''(\mathbf{p}) = \mathbf{0}$, by construction, we have in the expansion (10)

$$\mathbf{e}_{t+1} = \frac{1}{6} \mathbf{G}'''(\mathbf{p}) \mathbf{e}_t^3 + h.o.t. \quad (40)$$

where

$$\mathbf{G}'''(\mathbf{p}) = \begin{bmatrix} g_{111}^{111} & g_{111}^{112} & g_{111}^{121} & g_{111}^{211} & g_{111}^{122} & g_{111}^{212} & g_{111}^{221} & g_{111}^{222} \\ g_{112}^{111} & g_{112}^{112} & g_{112}^{121} & g_{112}^{211} & g_{112}^{122} & g_{112}^{212} & g_{112}^{221} & g_{112}^{222} \\ g_{121}^{111} & g_{121}^{112} & g_{121}^{121} & g_{121}^{211} & g_{121}^{122} & g_{121}^{212} & g_{121}^{221} & g_{121}^{222} \\ g_{211}^{111} & g_{211}^{112} & g_{211}^{121} & g_{211}^{211} & g_{211}^{122} & g_{211}^{212} & g_{211}^{221} & g_{211}^{222} \\ g_{122}^{111} & g_{122}^{112} & g_{122}^{121} & g_{122}^{211} & g_{122}^{122} & g_{122}^{212} & g_{122}^{221} & g_{122}^{222} \\ g_{212}^{111} & g_{212}^{112} & g_{212}^{121} & g_{212}^{211} & g_{212}^{122} & g_{212}^{212} & g_{212}^{221} & g_{212}^{222} \\ g_{221}^{111} & g_{221}^{112} & g_{221}^{121} & g_{221}^{211} & g_{221}^{122} & g_{221}^{212} & g_{221}^{221} & g_{221}^{222} \\ g_{222}^{111} & g_{222}^{112} & g_{222}^{121} & g_{222}^{211} & g_{222}^{122} & g_{222}^{212} & g_{222}^{221} & g_{222}^{222} \end{bmatrix}_{\mathbf{x}=\mathbf{p}} \quad (41)$$

$$\mathbf{e}_t^3 = \begin{bmatrix} e_1 e_1 e_1 \\ e_1 e_1 e_2 \\ e_1 e_2 e_1 \\ e_2 e_1 e_1 \\ e_1 e_2 e_2 \\ e_2 e_1 e_2 \\ e_2 e_2 e_1 \\ e_2 e_2 e_2 \end{bmatrix}_t \quad (42)$$

and

$$g_i^{jkl} = \frac{\partial^3 g_i}{\partial x_j \partial x_k \partial x_l} \quad (43)$$

where $i, j, k, l = 1, 2$. For the example considered here, we have

$$\mathbf{G}'''(\mathbf{p}) = \begin{bmatrix} 0 & \frac{5}{9} & \frac{5}{9} & \frac{5}{9} & \frac{1}{18} & \frac{1}{18} & \frac{1}{18} & -\frac{6}{9} \\ -\frac{1}{2} & -\frac{2}{9} & -\frac{2}{9} & -\frac{2}{9} & \frac{7}{18} & \frac{7}{18} & \frac{7}{18} & \frac{1}{6} \end{bmatrix}. \quad (44)$$

The quantity $\frac{1}{6} \mathbf{G}'''(\mathbf{p})$ is the 'asymptotic error constant' for this example.

As another example, consider

$$\begin{aligned} f_1(x_1, x_2) &= x_1 - \cos x_2 \\ f_2(x_1, x_2) &= \frac{x_2}{2} + \sin x_1 \end{aligned} \quad (45)$$

which has solution

$$\mathbf{p} = \begin{bmatrix} 0.53038868953899 \\ -1.01173733418201 \end{bmatrix}. \quad (46)$$

Iteration with initial value $\mathbf{x}_0 = \begin{bmatrix} 1 \\ -2 \end{bmatrix}$ gives

$$\begin{aligned} \mathbf{e}_1 &= \begin{bmatrix} 0.117 \\ -0.018 \end{bmatrix}, \mathbf{e}_2 = \begin{bmatrix} 0.312 \times 10^{-3} \\ 0.331 \times 10^{-3} \end{bmatrix} \\ \mathbf{e}_3 &= \begin{bmatrix} -0.014 \times 10^{-10} \\ 0.128 \times 10^{-10} \end{bmatrix}, \mathbf{e}_4 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \end{aligned} \quad (47)$$

The asymptotic error constant for this example is

$$\frac{\mathbf{G}'''(\mathbf{p})}{6} = \frac{1}{6} \begin{bmatrix} \frac{2159}{87} & -\frac{555}{2404} & -\frac{555}{2404} & -\frac{555}{2404} & -\frac{227}{6272} & -\frac{227}{6272} & -\frac{227}{6272} & -\frac{575}{3197} \\ \frac{2254}{87} & \frac{6272}{367} & \frac{6272}{367} & \frac{6272}{367} & -\frac{3026}{6272} & -\frac{3026}{6272} & -\frac{3026}{6272} & \frac{984}{3223} \\ \frac{87}{77} & \frac{367}{2404} & \frac{367}{2404} & \frac{367}{2404} & -\frac{555}{6272} & -\frac{555}{6272} & -\frac{555}{6272} & \frac{3223}{3197} \end{bmatrix}, \quad (48)$$

where the entries in this matrix are rational approximations.

6. Applications

Although the method presented here is a generic one, and is applicable to any situation in which a two-dimensional system must be solved, there are a few specific applications worth noting.

6.1 Root of a complex function

Say $f(z)$ is a complex-valued function, where $z = \alpha + \beta i$, $\alpha, \beta \in \mathbb{R}$. Furthermore, assume that f has the explicit form

$$\begin{aligned} f(z) &= f_1(z) + f_2(z)i \\ &= f_1(\alpha, \beta) + f_2(\alpha, \beta)i \end{aligned}$$

where f_1 and f_2 are real-valued. The root of f can be found by solving the system

$$\begin{aligned} f_1(\alpha, \beta) &= 0 \\ f_2(\alpha, \beta) &= 0 \end{aligned}$$

for α and β .

6.2 Extremum of a two-dimensional function

If we seek the extremum (maximum or minimum) of the function $f(x, y)$, then we might seek to solve the system

$$\begin{aligned} \frac{\partial f}{\partial x} &= 0 \\ \frac{\partial f}{\partial y} &= 0 \end{aligned}$$

for x and y .

6.3 Implicit Runge-Kutta method

The two-stage fourth-order Runge-Kutta method (Butcher, 2003)

$$\begin{array}{c|cc} c_1 & a_{11} & a_{12} \\ c_2 & a_{21} & a_{22} \\ \hline & b_1 & b_2 \end{array} = \begin{array}{cc|cc} \frac{1}{2} - \frac{\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{4} - \frac{\sqrt{3}}{6} & \\ \frac{1}{2} + \frac{\sqrt{3}}{6} & \frac{1}{4} + \frac{\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{2} \end{array}$$

for solving the initial-value problem

$$y' = f(x, y)$$

requires solving the two simultaneous stage equations

$$\begin{aligned} k_1 &= f(x_i + c_1 h, w_i + h a_{11} k_1 + h a_{12} k_2) \\ k_2 &= f(x_i + c_2 h, w_i + h a_{21} k_1 + h a_{22} k_2) \end{aligned}$$

for k_1 and k_2 , where x_i , w_i and h are known input. The computational effort required to solve these stage equations could render the method inefficient; hence, a cubically convergent algorithm can contribute positively to the overall efficiency of this Runge-Kutta method.

7. Conclusion

We have shown how a cubically convergent iteration scheme for solving a system of two nonlinear equations can be constructed, by appealing to fixed-point theory. We have defined a unipoint iteration function and, by demanding that the Jacobian and Hessian of this function should vanish at the fixed point, we have determined the entries of relevant matrices in the method. From a consistency point of view, we have shown that, in the univariate case, the method reduces to the known one-dimensional algorithm of Schröder. Some numerical examples serve to illustrate the cubic character of the method, and we have given values for the asymptotic error constant in these examples.

References

- Amat, S., and Busquier, S. (2007). Third-order iterative methods under Kantorovich conditions, *Journal of Mathematical Analysis and Applications*, 336, 243-261.
- Butcher, J.C. (2003). *Numerical methods for ordinary differential equations*, Chichester: Wiley.
- Cordero, A., Martínez, E., and Torregrosa, J.R. (2009). Iterative methods of order four and five for systems of nonlinear equations, *Journal of Computational and Applied Mathematics*, 231, 541-551.

- Darvishi, M.T., and Barati, A. (2007). A third-order Newton-type method to solve systems of nonlinear equations, *Applied Mathematics and Computation*, 187, 630-635.
- Darvishi, M.T., and Barati, A. (2007). Super cubic iterative methods to solve systems of nonlinear equations, *Applied Mathematics and Computation*, 188, 1678-1685.
- Hueso, J.L., Martínez, E., and Torregrosa, J.R. (2009). Third order iterative methods free from second derivative for nonlinear systems, *Applied Mathematics and Computation*, 215, 58-65.
- Hueso, J.L., Martínez, E., and Torregrosa, J.R. (2009). Third and fourth order iterative methods free from second derivative for nonlinear systems, *Applied Mathematics and Computation*, 211, 190-197.
- Kou, J. (2007). A third-order modification of Newton method for systems of non-linear equations, *Applied Mathematics and Computation*, 191, 117-121.
- Nedzhibov, G.H. (2008). A family of multi-point iterative methods for solving systems of nonlinear equations, *Journal of Computational and Applied Mathematics*, 222, 244-250.
- Traub, J.F. (1964). *Iterative Methods for the Solution of Equations*, New Jersey: Prentice-Hall.