Applied Mathematics and Computation 276 (2016) 394–406

Contents lists available at ScienceDirect

Applied Mathematics and Computation

journal homepage: www.elsevier.com/locate/amc

A class of efficient quadrature-based predictor–corrector methods for solving nonlinear systems

Cory L. Howk*

Department of Mathematics and Computer Science, Western Carolina University, Cullowhee, NC 28723, United States

ARTICLE INFO

Keywords:
Root-finding
Noninteger convergence
Nonlinear systems
Quasi-Newton Method
Quadrature-based technique

ABSTRACT

We extend a class of quadrature-based predictor–corrector techniques for root-finding to multivariate systems. They are found to have a rate of convergence of $1+\sqrt{2}$ regardless of the degree of precision for the quadrature technique from which they are derived, provided it is at least one. By reusing the linear system from the previous iterate, this class incorporates a significant improvement in computational time relative to the standard class through the inclusion of an LU-decomposition during the iteration. Complexity is equivalent to Newton's Method, as they only require knowledge of $F(x)$ and $F'(x)$.

1. Introduction

Systems of nonlinear equations are prevalent throughout the field of applied mathematics. They are especially prominent in the study of numerical discretization schemes for nonlinear ordinary (eg. [1–4]) and partial differential equations (eg. [1,5–11]). Approximation techniques for nonlinear systems generally rely on solving a sequence of linear systems that converge to the solution of the nonlinear system. In the absence of sparsity or special structure in these systems, the overall computational time can become unwieldy. Therefore, one of the main goals in developing techniques for approximating these solutions is to reduce the number of functional evaluations and overall computational time.

Recent studies (eg. [13,15,16,19,21–23,25–27,29] and references therein) have examined the application of quadrature formulae toward the development of root-finding methods exhibiting high rates of convergence. These studies have primarily focused on the solution of single univariate nonlinear equations, rather than systems. In a recent paper [21], the author examined convergence of the predictor–corrector method

$$\begin{aligned} x_n^* &= x_n - \frac{f(x_n)}{\sum_{k=1}^m w_k f'(\lambda_k x_{n-1} + (1-\lambda_k)x_{n-1}^*)} \\ x_{n+1} &= x_n - \frac{f(x_n)}{\sum_{k=1}^m w_k f'(\lambda_k x_n + (1-\lambda_k)x_n^*)}. \end{aligned} \tag{1}$$

This method is a modification of the modified Newton's method (mNm) family presented in [15]

* Tel.: +1 5073801082.
E-mail address: clhowk@email.wcu.edu

http://dx.doi.org/10.1016/j.amc.2015.12.032
0096-3003/

$$x_n^* = x_n - \frac{f(x_n)}{f'(x_n)}$$
$$x_{n+1} = x_n - \frac{f(x_n)}{\sum_{k=1}^{m} w_k f'(\lambda_k x_n + (1-\lambda_k)x_n^*)}. \tag{2}$$

System (1) utilizes fewer functional evaluations per iteration (provided $\lambda_k \neq 1$ for any k) and has a rate of convergence of $1+\sqrt{2}$, while System (2) has cubic convergence. When considering the balance of function evaluations and convergence rate via the efficiency index [17], System (1) is found to be superior provided $1 \leq m \leq 3$. In either system, convergence rate is independent of m, so the optimal method is the iteration scheme based on the Midpoint Method, which was studied in [25]:

$$x_n^* = x_n - \frac{f(x_n)}{f'(\frac{1}{2}[x_{n-1} + x_{n-1}^*])}$$
$$x_{n+1} = x_n - \frac{f(x_n)}{f'(\frac{1}{2}[x_n + x_n^*])}. \tag{3}$$

System (2) was extended to the p−dimensional case in [16],

$$x_n^* = x_n - [F'(x_n)]^{-1}F(x_n)$$
$$x_{n+1} = x_n - \left[\sum_{k=1}^{m} w_k F'((1-\lambda_k)x_n + \lambda_k x_n^*)\right]^{-1} F(x_n) \tag{4}$$

again resulting in cubic convergence. This method is advantageous to other cubically convergent methods for systems such as Halley's Method [18]

$$x_{n+1} = x_n - \left[I + \frac{1}{2}L_n\left(I - \frac{1}{2}L_n\right)^{-1}\right][F'(x_n)]^{-1}[F(x_n)]$$
$$L_n = [F'(x_n)]^{-1}[F''(x_n)][F'(x_n)]^{-1}[F(x_n)] \tag{5}$$

and Chebyshev's Method

$$x_{n+1} = x_n - \left[I + \frac{1}{2}L_n\right][F'(x_n)]^{-1}[F(x_n)] \tag{6}$$

as it does not require the second Frechet derivative of $F(x)$. However, it does require one to solve two linear systems per iteration. In the absence of special structure in these systems, this would typically necessitate two rounds of Gaussian Elimination per iteration, at a computational cost of $O(\frac{4}{3}n^3)$ flops per iterate.

In this paper we extend System (1) to the p-dimensional case,

$$x_n^* = x_n - \left[\sum_{k=1}^{m} w_k F'((1-\lambda_k)x_{n-1} + \lambda_k x_{n-1}^*)\right]^{-1} F(x_n)$$
$$x_{n+1} = x_n - \left[\sum_{k=1}^{m} w_k F'((1-\lambda_k)x_n + \lambda_k x_n^*)\right]^{-1} F(x_n). \tag{7}$$

This system is shown to have a rate of convergence of $1+\sqrt{2}$, independent of m. However, due to the reuse of the matrix

$$\Phi_{n-1} = \sum_{k=1}^{m} w_k F'((1-\lambda_k)x_{n-1} + \lambda_k x_{n-1}^*) \tag{8}$$

when calculating both x_n and x_n^*, one can generate the LU-decomposition of Φ_{n-1} during the calculation of x_n and use this data to calculate x_n^*. Only one round of Gaussian Elimination is required per iteration, reducing the computational time by nearly 50% per iteration for large systems, relative to System (4).

2. Iterative method

Suppose that $F : \mathbb{R}^p \to \mathbb{R}^p$ is continuously differentiable on a convex set $D \subset \mathbb{R}^p$. Then, for $x, y \in D$,

$$F(y) - F(x) = \int_0^1 F'(x + t(y-x))(y-x)dt. \tag{9}$$