International Journal of
Mathematical Modelling & Computations
Vol. 05, No. 01, Winter 2015, 41- 48

A Modified Steffensen's Method with Memory for Nonlinear Equations

F. Khaksar Haghani *

Department of Mathematics, Shahrekord Branch, Islamic Azad University, Shahrekord, Iran.

Abstract. In this work, we propose a modification of Steffensen's method with some free parameters. These parameters are then be used for further acceleration via the concept of with memorization. In this way, we derive a fast Steffensen-type method with memory for solving nonlinear equations. Numerical results are also given to support the underlying theory of the article.

Received: 9 July 2014, Revised: 9 August 2014, Accepted: 25 August 2014.

Keywords: Steffensen's method, Iterative methods, Root, Order, With memory.

Index to information contained in this paper

1. Introduction

Finding the root of a nonlinear equation

$$f(x) = 0, \tag{1}$$

where f is a sufficiently differentiable function in a neighborhood of a simple zero α is a classical problem in scientific computing, [10].

A derivative-free family of methods proposed by Steffensen in [9] for solving (1) as follows (SM)

$$x_{k+1} = x_k - \frac{f(x_k)}{f[x_k, w_k]}, \quad \beta \in \mathbb{R}\backslash\{0\}, \; k = 0, 1, 2, \cdots , \tag{2}$$

*Corresponding author. Email: haghani1351@yahoo.com

where $w_k = x_k + \beta f(x_k)$. Considering this iterative expression, many higher order methods with higher computational efficiency indices have been developed in the literature (see e.g. [3], [4] and [7]).

The aim of this short communication is to state a one-step method with memory of very high computational efficiency. We start from a modification of the one-step two-point method of Steffensen (2) without memory with order 2, and increase the convergence order to 3.56155 without any additional function evaluation which provides a high computational efficiency index.

In general, the free nonzero parameter β in Steffensen-like methods such (2) is of great importance for accelerating the R-order of convergence without additional calculations.

In this manner, we would like to obtain new methods for finding simple roots of nonlinear equations, whose computational efficiency is higher than the efficiency of existing methods known in literature in the class of one-step methods and even higher than the efficiency indices of optimal three- and four-step methods of orders eight and sixteen, respectively.

Recently, Džunić in [2] designed an efficient one-step bi-parametric iterative method with memory (DM) possessing $\frac{1}{2}(3+\sqrt{17})$ R-order of convergence as follows:

$$\begin{cases} w_k = x_k + \beta_k f(x_k), \\ \beta_k = -\dfrac{1}{N_2'(x_k)}, \qquad p_k = -\dfrac{N_3''(w_k)}{2N_3'(w_k)}, \qquad k \geqslant 1, \\ x_{k+1} = x_k - \dfrac{f(x_k)}{f[x_k, w_k] + p_k f(w_k)}, \qquad k \geqslant 0, \end{cases} \tag{3}$$

where $N_j(l)$ stands for the Newton's interpolatory polynomial of j-th order passing through $j+1$ nodes at the point l. For example, we can define $N_3(t)$ as the Newton's interpolation polynomial of third degree, set through four available approximations $x_k, w_k, w_{k-1}, x_{k-1}$.

It should be remarked that the notation of divided difference is used frequently in this study.

Here, we aim at presenting a simple quadra-parametric modification of Steffensen's method without memory and make it with memory. Higher order of convergence is attained without additional function evaluations, making the derived method very efficient. Numerical examples are also given to demonstrate excellent convergence features of the presented method with memory.

The rest of this paper is organized as follows. In Section 2, a modification of (2) without memory is given possessing quadratic convergence. The main goal of this paper is presented in Section 3 by contributing an iterative method with memory. The proposed scheme is an extension over (2) and has a simple structure with an increased computational efficiency. Its efficiency index is the same to (3). In Section 4, numerical reports are stated. Some discussions will be given in Section 5 to end the paper.

2. Modified Steffensen's Method

In order to modify (2) and have as much as possible of free parameters, we first apply the backward finite difference approximation in the denominator of (2) and then introduce a weight function in what follows (MSM):

$$\begin{cases} w_k = x_k - \beta f(x_k), & \beta \in \mathbb{R}\backslash\{0\}, \qquad k \geqslant 0, \\ x_{k+1} = x_k - \dfrac{f(x_k)}{f[x_k, w_k] + qf(x_k) + pf(w_k)} \left(1 + \xi \frac{f(w_k)}{f[x_k, w_k]}\right), & q, p, \xi \in \mathbb{R}. \end{cases} \tag{4}$$

Theorem 2.1 Let the function $f(x)$ be sufficiently differentiable in a neighborhood of its simple zero α. If an initial approximation x_0 is sufficiently close to α. Then, the order of convergence of the method (4) without memory is two with four parameters.

Proof Introducing the notations $c_i = \frac{1}{i!}\frac{f^i(\alpha)}{f'(\alpha)}$, $dfa = f'(\alpha)$, $e1 = x_{k+1} - \alpha$, $e = x_k - \alpha$, $ew = w_k - \alpha$, and $t = f(w_k)/f[x_k, w_k]$, we can provide the final error equation of (4) by applying the following piece of code written in the symbolic package of Mathematica:

```
ClearAll["Global`*"]
f[e_] := dfa (e^1 + c2 e^2 + c3 e^3 + c4 e^4)
fe = f[e]; f1e = f'[e];
ew = e - \[Beta] fe; fw = f[ew];
FD1 = (fe - fw)/(e - ew); t = fw/FD1;
e1 = e - Series[fe/( FD1 + q fe + p fw),
        {e, 0, 2}] (1 + \[Zeta] t) //FullSimplify
```

This gives the following error equation

$$e_{k+1} = (c_2 + p + q - c_2 f'(\alpha)\beta - \zeta + f'(\alpha)\beta(-p + \zeta))e_k^2 + O(e_k^3). \tag{5}$$

The relation (5) shows the quadratic convergence of (4). Now, the proof is complete. ■

In the next section, we aim at accelerating convergence without imposing further functional evaluations per cycle. This means that using two function evaluations, we must increase the R-order of convergence more than two [5]. This is possible by approximating the free parameters involved in (4). The best method for our recent goal would be the following bi-parametric case of (4)

$$\begin{cases} w_k = x_k - \beta f(x_k), & \beta \in \mathbb{R}\backslash\{0\}, \qquad k \geqslant 0, \\ x_{k+1} = x_k - \dfrac{f(x_k)}{f[x_k, w_k]} \left(1 + \xi \frac{f(w_k)}{f[x_k, w_k]}\right), & \xi \in \mathbb{R}, \end{cases} \tag{6}$$

where it reads

$$e_{k+1} = -(-1 + f'(\alpha)\beta)(c_2 - \zeta)e_k^2 + O(e_k^3). \tag{7}$$

Note that the effect of the other two parameters p and q could be ignored by considering the other two parameters.

3. Modified Steffensen's Method with Memory

Now we try to construct a method with memory consists of the calculation of parameters

$$\beta = \beta_k, \tag{8}$$

and

$$\xi = \xi_k, \tag{9}$$

as the iteration proceeds by the formulas $\beta_k = \dfrac{1}{\bar{f}'(\alpha)}$ and $\xi_k = \bar{c}_2$, for $k = 1, 2, \cdots$, where $\bar{f}'(\alpha)$ and $\bar{c}_2$ are approximations to $f(\alpha)$ and c_2, respectively.

As a matter of fact, in this way we minimize the factors $-1 + f'(\alpha)\beta_k$ and $c_2 - \zeta_k$ that appear in (7). Hence, we present the following scheme (PM)

$$\begin{cases} w_k = x_k - \beta_k f(x_k), \\ \beta_k = \dfrac{1}{N_2'(x_k)}, \qquad \xi_k = \dfrac{N_3''(w_k)}{2N_3'(w_k)}, \qquad k \geqslant 1, \\ x_{k+1} = x_k - \dfrac{f(x_k)}{f[x_k, w_k]} \left(1 + \xi_k \frac{f(w_k)}{f[x_k, w_k]}\right), \qquad k \geqslant 0. \end{cases} \tag{10}$$

The two self-accelerating parameters, i.e., β_k and ξ_k are calculated using information available from the current and previous iterations. Moreover, it is assumed that initial estimates ξ_0 and β_0 should be chosen before starting the iterative process.

Theorem 3.1 Let the function $f(x)$ be sufficiently differentiable in a neighborhood of its simple zero α. If an initial approximation x_0 is sufficiently close to α. Then, the R-order of convergence of the one-step method (10) with memory is at least $\frac{1}{2}(3 + \sqrt{17})$.

Proof. Let $\{x_k\}$ be a sequence of approximations generated by (10). We first must find the asymptotic error constants for the two self-accelerator parameters. Following the substitutions $\beta_k = \dfrac{1}{N_2'(x_k)}$ and $\xi_k = \dfrac{N_3''(w_k)}{2N_3'(w_k)}$, and similar notations as in Theorem 2.1, we could write the following two pieces of Mathematica codes:

```
ClearAll["Global‘*"]
A[t_]:= InterpolatingPolynomial[{{e, fx}, {ew, fw}, {e1, fx1}}, t]
Approximation = 1/A’[e1] // Simplify;
fx = f1a*(e + c2*e^2 + c3*e^3 + c4*e^4);
fw = f1a*(ew + c2*ew^2 + c3*ew^3 + c4*ew^4);
fx1 = f1a*(e1 + c2*e1^2 + c3*e1^3 + c4*e1^4);
b = Series[Approximation, {e, 0, 2}, {ew, 0, 2},
                         {e1, 0, 0}] //Simplify;
Collect[Series[-1 + b*f1a, {e, 0, 1}, {ew, 0, 1},
                         {e1, 0, 0}], {e, ew, e1}, Simplify]
```

which results in

$$-1 + \beta_k f'(\alpha) \sim c_3 e_{k-1} e_{k-1,w}, \tag{11}$$

and also

```
ClearAll["Global`*"]
A[t_] :=InterpolatingPolynomial[{{e, fx}, {ew, fw},
                    {e1, fx1}, {ew1, fw1}}, t]
Approximation = A''[ew1]/(2 A'[ew1]) // Simplify;
fx = f1a*(e + c2*e^2 + c3*e^3 + c4*e^4);
fw = f1a*(ew + c2*ew^2 + c3*ew^3 + c4*ew^4);
fx1 = f1a*(e1 + c2*e1^2 + c3*e1^3 + c4*e1^4);
fw1 = f1a*(ew1 + c2*ew1^2 + c3*ew1^3 + c4*ew1^4);
h = Series[Approximation, {e, 0, 2}, {ew, 0, 2},
                 {e1, 0, 0}, {ew1, 0, 0}] //Simplify;
Collect[Series[c2 - h, {e, 0, 1}, {ew, 0, 1}, {e1, 0, 0},
                    {ew1, 0, 0}], {e, ew, e1, ew1}, Simplify]
```

which results in

$$c_2 - \xi_k \sim c_4 e_{k-1} e_{k-1,w}, \tag{12}$$

wherein $ew = e_{k-1,w}$, $e = e_{k-1}$, $e1 = e_k$. Therefore, one may obtain

$$e_{k+1} \sim -(c_3 e_{k-1} e_{k-1,w})(c_4 e_{k-1} e_{k-1,w}) e_k^2. \tag{13}$$

We also have $e_{k-1,w} \sim (-1 + \beta_{k-1} f'(\alpha)) e_{k-1}$. So, we attain

$$e_{k+1} \sim -(c_3 c_4 e_{k-1}^2)((-1 + \beta_{k-1} f'(\alpha)) e_{k-1})^2 e_k^2. \tag{14}$$

and

$$e_{k+1} \sim -c_3 c_4 (-1 + \beta_{k-1} f'(\alpha))^2 e_{k-1}^4 e_k^2. \tag{15}$$

Note that in general we know that the error equation should read $e_{k+1} \sim A e_k^p$, where A and p are to be determined. Hence, one has $e_k \sim A e_{k-1}^p$, and subsequently

$$e_{k-1} \sim A^{-1/p} e_k^{1/p}. \tag{16}$$

Hence, we firstly have

$$e_{k+1} \sim -c_3 c_4 (-1 + \beta_{k-1} f'(\alpha))^2 \left(\frac{e_k}{-(-1 + \beta_{k-1} f'(\alpha))(c_2 - \zeta_{k-1})} \right) e_{k-1}^2 e_k^2. \tag{17}$$

Thus, it is easy to obtain

$$e_k^p \sim A^{-2/p} C e_k^{3+\frac{2}{p}}, \tag{18}$$

wherein

$$C = -c_3 c_4 (-1 + \beta_{k-1} f'(\alpha))^2 \left(\frac{1}{-(-1 + \beta_{k-1} f'(\alpha))(c_2 - \zeta_{k-1})} \right). \tag{19}$$

This results in the equation $p = 3 + \frac{2}{p}$, with two solutions $\{\frac{1}{2}\left(3 - \sqrt{17}\right)$, $\frac{1}{2}\left(3 + \sqrt{17}\right)\}$. Clearly the value for $p = \frac{1}{2}\left(3 + \sqrt{17}\right) \approx 3.56155$ is acceptable and would be the convergence R-order of the method (10) with memory. The proof is complete. ■

We emphasize that the increase of the convergence order is obtained without any additional function evaluations, which points to very high computational efficiency.

The computational efficiency index of (10) is $3.56155^{\frac{1}{2}} \approx 1.8872$, which is same to (3), and is clearly much higher than $2^{\frac{1}{2}} \approx 1.4142$ of (2) and (4), $8^{\frac{1}{4}} \approx 1.6817$ of optimal eighth-order method [6] and $16^{\frac{1}{5}} \approx 1.7411$ of optimal sixteenth-order schemes.

4. Numerical Computations

In this section we compare the behavior of different methods for solving an example in programming package Mathematica [11]. Numerical experiments have been performed with 1500 precision digits, being large enough to minimize round-off errors as well as to clearly observe the computed asymptotic error constants requiring small number divisions. In fact, it clearly shows the high computational R-order of the proposed method.

We compare the methods SM using $\beta = 0.1$, MSM using $\beta = 0.1$, $p = q = 1/4$, $\xi = 0$, DM and PM. All experiments have been carried out on a personal computer equipped with an AMD 3.1 GHz dual-core processor and Windows 32-bit XP operating system.

In the meantime, the computational order of convergence (coc) has been computed by

$$\rho = \frac{\ln |f(x_k)/f(x_{k-1})|}{\ln |f(x_{k-1})/f(x_{k-2})|}. \tag{20}$$

Here the stop termination is $|f(x_k)| \leqslant 10^{-250}$. We have used $\beta_0 = \xi_0 = 0.1$ whenever required.

Example 4.1 We consider the following nonlinear test function

$$f(x) = (x - 2\tan(x))(x^3 - 8), \tag{21}$$

where $\alpha = 2$. The results are provided in Tables 1-2 for two different initial approximations.

The values of initial guess x_0 were selected close to α to guarantee convergence of iterative methods.

Tables 1-2 evidently show that proposed scheme (10) exhibits 3.56 convergence. Iterative scheme (10) is evidently believed to be more favorable than other listed methods due to its fast speed and acceptable accuracy.

We remark that convergence behavior was verified for additional test functions and we attained similar superiority for our proposed Steffensen-like method with memory.

5. Conclusion

For the first time, iterative methods with memory for solving nonlinear equations, that use information from the current and previous iterate, were considered in [10]. After Traub's research this class of methods was studied seldom in the literature

Table 1. Results of comparisons for Example 4.1 by $x_0 = 1.7$.

Methods	$\|f(x_3)\|$	$\|f(x_4)\|$	$\|f(x_5)\|$	$\|f(x_6)\|$	ρ
SM	4.1583	3.0743	1.4436	0.25430	2.
MSM	23.499	18.452	12.275	0.60559	2.
DZ	0.13132	2.0026×10^{-7}	1.0181×10^{-27}	5.1731×10^{-99}	3.57002
PM	1.8921×10^{-6}	4.5864×10^{-24}	1.0569×10^{-88}	7.5269×10^{-318}	3.54512

Table 2. Results of comparisons for Example 4.1 by $x_0 = 1.92$.

Methods	$\|f(x_3)\|$	$\|f(x_4)\|$	$\|f(x_5)\|$	$\|f(x_6)\|$	ρ
SM	0.032743	0.00010819	$1.1761 * 10^{-9}$	1.3898×10^{-19}	2.
MSM	0.0018889	2.9274×10^{-7}	7.0285×10^{-15}	4.0516×10^{-30}	2.
DZ	0.041691	5.5105×10^{-8}	8.4457×10^{-32}	5.2177×10^{-115}	3.57209
PM	1.4425×10^{-15}	1.3731×10^{-57}	1.6322×10^{-207}	2.4848×10^{-741}	3.56056

in spite of its capability to reach high computational efficiency. Recent results published in [2] and [3] showed considerably high computational efficiency of methods with memory using new accelerating techniques based on varying free parameters calculated by interpolating polynomials in each iteration.

For these reasons, in this paper we have constructed a family of Steffensen-type methods without memory possessing quadratic convergence with four free parameters. Then, by using suitable accelerators we have constructed a method with memory possessing a high computational efficiency index.

Numerical results have also been provided to support the theoretics given in Sections 2-3. Observing the analytical and numerical results, one may conclude that the proposed variant of Steffensen's method with memory is an efficient tool for solving nonlinear equations.

The application of the developed method for matrix problems (such as the ones in [8]) and its dynamical studies (see e.g. [1]), will be pursued in future works.

Acknowledgements

The author sincerely thanks 'Islamic Azad University, Shahrekord Branch, Shahrekord, Iran' for financial support.

Conflict of Interests

The author declares that there is no conflict of interests regarding the publication of this paper.

References

[1] A. Cordero, F. Soleymani and J. R. Torregrosa, Dynamical analysis of iterative methods for nonlinear systems or how to deal with the dimension?, Applied Mathematics and Computation, **244** (2014) 398-412.

[2] J. Džunić, On efficient two-parameter methods for solving nonlinear equations, Numer. Algor. **63** (2013) 549-569.

[3] T. Lotfi and E. Tavakoli, On a new efficient Steffensen-like iterative class by applying a suitable self-accelerator parameter, The Scientific World Journal, Article ID 769758, 9 pages.(2014).

[4] F. Mirzaee and A. Hamzeh, A sixth order method for solving nonlinear equations, International Journal of Mathematical Modelling & Computations,**04** (2014) 55-60.
[5] J. M. Ortega and W. C. Rheinbolt, Iterative Solution of Nonlinear Equations in Several Variables, Academic Press, New York, (1970).
[6] F. Soleymani, An optimally convergent three-step class of derivative-free methods, World Applied Sciences Journal, **13** (2011) 2515-2521.
[7] F. Soleymani, M. Sharifi, S. Shateyi and F. Khaksar Haghani, A class of Steffensen-type iterative methods for nonlinear systems, Journal of Applied Mathematics,9 Pages, (2014).
[8] F. Soleymani, M. Sharifi, S. Karimi Vanani, F. Khaksar Haghani and A. Kiliçman, An inversion-free method for finding positive definite solution of a rational matrix equation, The Scientific World Journal, Article ID 560931, 5 pages,(2014).
[9] J. F. Steffensen, Remarks on iteration, Skand. Aktuarietidskr, **16** (1933) 64-72.
[10] J. F. Traub, Iterative Methods for the Solution of Equations, Prentice Hall, New York, (1964).
[11] S. Wagon, Mathematica in Action, Third edition, Springer, New York, NY, USA, (2010).