Algorithms **2015**, *8*, 786-798; doi:10.3390/a8030786

OPEN ACCESS

algorithms

ISSN 1999-4893

www.mdpi.com/journal/algorithms

Article

A Family of Newton Type Iterative Methods for Solving Nonlinear Equations

Xiaofeng Wang [1,*], Yuping Qin [2], Weiyi Qian [1], Sheng Zhang [1] and Xiaodong Fan [1]

[1] School of Mathematics and Physics, Bohai University, Jinzhou 121013, China;
E-Mais: tgzybhu@163.com (W.Q.); 15041699258@163.com (S.Z.); fxd@emails.bjut.edu.cn (X.F.)

[2] College of Engineering, Bohai University, Jinzhou 121013, China;
E-Mai: jzqinyuping@gmail.com

* Author to whom correspondence should be addressed; E-Mail: w200888w@163.com;
Tel.: +86-416-340-0145.

Academic Editors: Alicia Cordero, Juan R. Torregrosa and Francisco I. Chicharro

Received: 9 July 2015 / Accepted: 15 September 2015 / Published: 22 September 2015

Abstract: In this paper, a general family of n-point Newton type iterative methods for solving nonlinear equations is constructed by using direct Hermite interpolation. The order of convergence of the new n-point iterative methods without memory is 2^n requiring the evaluations of n functions and one first-order derivative in per full iteration, which implies that this family is optimal according to Kung and Traub's conjecture (1974). Its error equations and asymptotic convergence constants are obtained. The n-point iterative methods with memory are obtained by using a self-accelerating parameter, which achieve much faster convergence than the corresponding n-point methods without memory. The increase of convergence order is attained without any additional calculations so that the n-point Newton type iterative methods with memory possess a very high computational efficiency. Numerical examples are demonstrated to confirm theoretical results.

Keywords: multipoint iterative methods; nonlinear equations; R-order convergence; root-finding methods

---

1. Introduction

Solving nonlinear equations by iterative methods have been of great interest to numerical analysts. The most famous one-point iterative method is probably Newton's Equation [1]: $x_{k+1} = x_k - f(x_k)/f'(x_k)$, which converges quadratically. However, the condition $f'(x) \neq 0$ in a neighborhood of the required root is severe indeed for convergence of Newton method, which restricts its applications in practical. For resolving this problem, Wu in [2] proposed the following one-point iterative method

$$x_{k+1} = x_k - \frac{f(x_k)}{\lambda f(x_k) + f'(x_k)} \tag{1}$$

where $\lambda \in R$, $0 < |\lambda| < +\infty$ and λ is chosen such that the corresponding function values $\lambda f(x_k)$ and $f'(x_k)$ have the same signs. This method converges quadratically under the condition $\lambda f(x_k) + f'(x_k) \neq 0$, while $f'(x_k) = 0$ in some points is permitted. Wang and Zhang in [3] obtained the error equation of the Equation (1) as follows

$$e_{k+1} = (c_2 + \lambda)e_k^2 + O(e_k^3) \tag{2}$$

where $e_k = x_k - a,\ c_k = (1/k!)f^{(k)}(a)/f'(a),\ k = 2, 3, \cdots$ and a is the root of the nonlinear equation $f(x) = 0$.

The convergence order and computational efficiency of the one-point iterative methods are lower than multipoint iterative methods. Multipoint iterative methods can overcome theoretical limits of one-point methods concerning the convergence order and computational efficiency. In recent years, many multipoint iterative methods have been proposed for solving nonlinear equations, see [4–18]. Wang and Liu in [4] developed the following eighth-order iterative method without memory by Hermite interpolation methods

$$\begin{cases} y_k = x_k - \dfrac{f(x_k)}{f'(x_n)} \\ z_k = y_k - \dfrac{f(y_k)}{2f[y_k, x_k] - f'(x_k)} \\ x_{k+1} = z_k - \dfrac{f(z_k)}{N(x_k, y_k, z_k)} \end{cases} \tag{3}$$

where $N(x_k, y_k, z_k) = f[z_k, y_k] + 2f[z_k, x_k] - 2f[y_k, x_k] + f[y_k, x_k, x_k](y_k - z_k)$. Using the same strategy, Kou in [5] presented a family of eighth-order iterative method without memory. The Equation (3) is a special case of the Kou's method. Petković in [6] claimed a general class of optimal n-point methods without memory by Hermite interpolation methods, which have the order of convergence 2^n and require evaluations of n functions and one first-order derivative. The Equation (3) is a special case of the Petković's n-point Method for $n = 3$. But, the Petković's n-point method gives no specific iterative scheme and error relation for $n \geq 4$. In this paper, we construct a class of n-point iterative methods with and without memory by Hermite interpolation methods and give the specific iterative scheme and error relation for all $n \geq 2$.

This paper is organized as follows. In Section 2, based on Wu's Equation [2] and Petković's n-point Equation [6], we derive a family of n-point iterative methods without memory for solving nonlinear equations. We prove that the order of convergence of the n-point methods without memory is 2^n requiring the evaluations of n functions and one first-order derivative in per full iteration. Kung and Traub in [7] conjectured that a multipoint iteration without memory based on n functional evaluations could achieve an optimal convergence of order 2^{n-1}. The new methods without memory agree with the conjecture. Further accelerations of convergence speed are attained in Section 3. A family of n-point iterative methods with memory is obtained by using a self-accelerating parameter in per full iteration. This self-accelerating parameter is calculated using information available from the current and previous iterations. Numerical examples are given in Section 4 to confirm theoretical results.

2. The Optimal Fourth-, Eighth- and 2^n th Order Iterative Methods

Based on Wu's Equation [2] and Petković's n-point methods [6], we derive a general optimal 2^nth order family and write it in the following form:

$$\begin{cases} y_{k,1} = y_{k,0} - \dfrac{f(y_{k,0})}{\lambda f(y_{k,0}) + f'(y_{k,0})} \\ y_{k,2} = y_{k,1} - \dfrac{f(y_{k,1})}{f[y_{k,1}, y_{k,0}] + f[y_{k,1}, y_{k,0}, y_{k,0}](y_{k,1} - y_{k,0})} \\ \quad \cdots \\ y_{k,n} = y_{k,n-1} - \dfrac{f(y_{k,n-1})}{N(y_{k,n-1}, y_{k,n-2} \cdots, y_{k,1}, y_{k,0})} \end{cases} \tag{4}$$

where $N(y_{k,n-1}, y_{k,n-2}, \cdots, y_{k,1}, y_{k,0}) = f[y_{k,n-1}, y_{k,n-2}] + \cdots + f[y_{k,n-1}, y_{k,n-2}, \cdots, y_{k,1}, y_{k,0}, y_{k,0}]$ $(y_{k,n-1} - y_{k,n-2}) \cdots (y_{k,n-1} - y_{k,0})$, $y_{k,0} = x_k$, $\lambda \in R$ is a constant and k being the iteration index. The entries $y_{k,0}, \cdots y_{k,n}$ are approximations with the associated error $e_{k,j} = y_{k,j} - a$ $(j = 0, 1, \cdots, n)$.

Using the Taylor series and symbolic computation in the programming package Mathematica, we can find the order of convergence and the asymptotic error constant (AEC) of the n-point methods Equation (4) for $n = 1, n = 2$ and $n = 3$, respectively. For simplicity, we sometimes omit the iteration index n and write e instead of e_k. The approximation x_{k+1} to the root a will be denoted by $\hat{x}$. Regarding Equation (4), let us define $x = y_{k,0}, y = y_{k,1}, z = y_{k,2}, e = x - a, d = y - a, p = z - a, e1 = \hat{x} - a$.

The following abbreviations are used in the program.

$$\text{ck} = f^{(k)}(a)/(k!f'(a)),\ \text{e} = x - a,\ \text{d} = y - a,\ \text{p} = z - a,\ \text{e1} = \hat{x} - a,\ \text{fx} = f(y_{k,0}),\ \text{fy} = f(y_{k,1})$$

$$\text{dfx} = f'(y_{k,0}),\ \text{fxxy} = f[y_{k,0}, y_{k,0}, y_{k,1}],\ \text{fla} = f'(a),\ \ \text{fyz} = f[y_{k,1}, y_{k,2}], \text{fxz} = f[y_{k,0}, y_{k,2}],$$

$$\text{fz} = f(y_{k,2}),\ \text{L} = \lambda, \text{fzxx} = f[y_{k,2}, y_{k,0}, y_{k,0}],\ \text{fxy} = f[y_{k,0}, y_{k,1}], \text{fzxxy} = f[y_{k,2}, y_{k,0}, y_{k,0}, y_{k,1}].$$

Program (written in Mathematica)

```
fx=fla*(e+c2*e^2+c3*e^3+c4*e^4+c5*e^5+c6*e^6+c7*e^7+c8*e^8);
dfx=D[fx,e];
```

```
t=Series[fx/(L*fx+dfx),{e,0,8}];
d=Series[e-t,{e,0,8}]//Simplify
fy=Series[fla*(d+c2*d^2+c3*d^3+c4*d^4),{e,0,8}];
fxy=Series[(fy-fx)/(d-e),{e,0,8}];
z=Series[d-fy/(2*fxy-dfx),{e,0,8}]//Simplify
fz=Series[fla*(z+c2*z^2),{e,0,8}];
fyz=Series[(fy-fz)/(d-z),{e,0,8}];
fxz=Series[(fx-fz)/(e-z),{e,0,8}];
fxxy=Series[(dfx-fxy)/(e-d),{e,0,8}];
fzxx=Series[(dfx-fxz)/(e-z),{e,0,8}];
fzxxy=Series[(fzxx-fxxy)/(z-d),{e,0,8}];
fzxy=Series[(fxz-fyz)/(e-d),{e,0,8}];
e1=Series[z-fz/(fyz+fzxy*(z-d)+fzxxy*(z-e)*(z-d)),{e,0,8}]//Simplify
```

$$\text{Out[d]} = (c_2 + L)e^2 + O[e]^3 \tag{5}$$

$$\text{Out[z]} = (c_2 + L)(c_2^2 - c_3 + c_2 L)e^4 + O[e]^5 \tag{6}$$

$$\text{Out[e1]} = (c_2 + L)^2(c_2^2 - c_3 + c_2 L)(c_2^3 - c_2 c_3 + c_4 + c_2^2 L)e^8 + O[e]^9 \tag{7}$$

We obtain the asymptotic error constants of n-point methods Equation (4) with $n = 1, 2, 3$. Altogether, we can state the following theorem.

Theorem 1. Let I be an open interval and $a \in I$ a simple zero point of a sufficiently differentiable function $f : I \to R$. Then the new method defined by Equation (4) $(n = 2)$is fourth order, and satisfies the error equation

$$e_{k+1} = (c_2 + \lambda)d_2 e_k^4 + O(e_k^5) \tag{8}$$

the Equation (4) $(n = 3)$ is eighth-order and satisfies the error equation

$$e_{k+1} = (c_2 + \lambda)^2 d_3 e_k^8 + O(e_k^9) \tag{9}$$

where $e_k = x_k - a,\ d_0 = 1,\ d_1 = 1,\ d_2 = c_2^2 + c_2\lambda - c_3$ and $d_3 = d_2(c_2 d_2 + c_4 d_1 d_0)$.

The order of the convergence of the Equation (4) is analyzed in the following theorem.

Theorem 2. Let I be an open interval and $a \in I$ a simple zero point of a sufficiently differentiable function $f : I \to R$. Then the n-point family Equation (4) converges with at least 2^nth order and satisfies the error relation

$$e_{k+1} = e_{k,n} = y_{k,n} - a = (c_2 + \lambda)^{2^{n-2}} d_n e_k^{2^n} + O(e_k^{2^n+1}) \tag{10}$$

where $e_k = e_{k,0} = y_{k,0} - a$ and $d_n = d_{n-1}(c_2 d_{n-1} + (-1)^{n-1} c_{n+1} d_{n-2} \cdots d_1 d_0)$, $n \geq 3$

Proof. We prove the theorem by induction. For $n = 3$, the theorem is valid by Theorem 1. Let us assume that Equation (10) is true for the intermediate error relations, then the intermediate error relations are of the form

$$e_{k,j} = y_{k,j} - a = (c_2 + \lambda)^{2^{j-2}} d_j e_k^{2^j} + O(e_k^{2^j+1}) \tag{11}$$

where $e_{k,j} = y_{k,j} - a$, $d_j = d_{j-1}(c_2 d_{j-1} + (-1)^{j-1} c_{j+1} d_{j-2} \cdots d_1 d_0)$, $j = 3, \cdots n-1$. Using Equations (4) and (11) and noting that $e_{k,0}e_{k,0}e_{k,1}e_{k,2}\cdots e_{k,n-1} = O(e_k^{1+1+2+4+\cdots+2^{n-2}}) = O(e_k^{2^{n-1}})$, we have

$$\begin{aligned}
e_{k+1} =& e_{k,n-1}(f'(a)+O(e_k))^{-1}(f[y_{k,n-1}, y_{k,n-2}, y_{n-3}]e_{k,n-1}\cdots + (-1)^{n-1} \\
&\times f[y_{k,n-1}, y_{k,n-2}, \cdots, y_{k,1}, y_{k,0}, y_{k,0}, a]e_{k,n-2}\cdots e_{k,0}e_{k,0} + O(e_k^{2^{n-1}+1})) \\
=& e_{k,n-1}\left(c_2 e_{k,n-1} + (-1)^{n-1}c_{n+1}e_{k,n-2}\cdots e_{k,0}e_{k,0} + O(e_k^{2^{n-1}+1})\right) \\
=& (c_2+\lambda)^{2^{n-3}} d_{n-1} e_k^{2^{n-1}} (c_2(c_2+\lambda)^{2^{n-3}} d_{n-1} e_k^{2^{n-1}} + (-1)^{n-1}c_{n+1} \\
&\times (c_2+\lambda)^{2^{n-4}} d_{n-2} e_k^{2^{n-2}} \cdots (c_2+\lambda) d_2 e_k^4 (c_2+\lambda) d_1 e_k^2 d_0 e_k^2) \\
=& (c_2+\lambda)^{2^{n-2}} d_{n-1}[c_2 d_{n-1} + (-1)^{n-1} c_{n+1} d_{n-2}\cdots d_2 d_1 d_0] e_k^{2^n}
\end{aligned} \tag{12}$$

Hence, by induction, we conclude that the error relations can be written in the following form

$$e_{k+1} = e_{k,n} = y_{k,n} - a = (c_2+\lambda)^{2^{n-2}} d_n e_k^{2^n} + O(e_k^{2^n+1}) \tag{13}$$

3. New Families of Iterative Methods with Memory

In this section we will improve the convergence order of the family Equation (4). We observe from Equation (13) that the order of convergence of the family Equation (4) is 2^n when $\lambda \neq -c_2$. With the choice $\lambda = -c_2 = -f''(a)/(2f'(a))$, it can be proved that the order of the family Equation (4) would exceed 2^n. However, the exact values of $f'(a)$ and $f''(a)$ are not available in practice and such acceleration of convergence can not be realized. But we could approximate the parameter λ by λ_k. The parameter λ_k can be computed by using information available from the current and previous iterations and satisfies $\lim_{k\to\infty}\lambda_k = -c_2 = -f''(a)/(2f'(a))$, such that the 2^nth order asymptotic convergence constant to be zero in Equation (13). We consider the following three methods for λ_k:

$$\lambda_k = -H_2''(y_{k,0})/(2f'(y_{k,0})) \tag{14}$$

where $H_2(x) = f(y_{k,0}) + f[y_{k,0}, y_{k,0}](x - y_{k,0}) + f[y_{k,0}, y_{k,0}, y_{k-1,n-1}](x-y_{k,0})^2$, and $H_2''(y_{k,0}) = 2f[y_{k,0}, y_{k,0}, y_{k-1,n-1}]$.

$$\lambda_k = -H_3''(y_{k,0})/(2f'(y_{k,0})) \tag{15}$$

where $H_3(x) = H_2(x) + f[y_{k,0}, y_{k,0}, y_{k-1,n-1}, y_{k-1,n-2}](x-y_{k,0})^2(x-y_{k-1,n-1})$, and $H_3''(y_{k,0}) = H_2''(y_{k,0}) + 2f[y_{k,0}, y_{k,0}, y_{k-1,n-1}, y_{k-1,n-2}](y_{k,0} - y_{k-1,n-1})$.

$$\lambda_k = -H_4''(y_{k,0})/(2f'(y_{k,0})) \tag{16}$$

where $H_4(x) = H_3(x) + f[y_{k,0}, y_{k,0}, y_{k-1,n-1}, y_{k-1,n-2}, y_{k-1,n-3}](x-y_{k,0})^2(x-y_{k-1,n-1})(x-y_{k-1,n-2})$ and $H_4''(y_{k,0}) = H_3''(y_{k,0}) + 2f[y_{k,0}, y_{k,0}, y_{k-1,n-1}, y_{k-1,n-2}, y_{k-1,n-3}](y_{k,0} - y_{k-1,n-1})(y_{k,0} - y_{k-1,n-2})$.

The parameter λ_k is recursively calculated as the iteration proceeds using Equations (14)–(16) in Equation (4). Substituting λ_k instead of λ in Equation (4), we can obtain the following iterative method with memory

$$\begin{cases} y_{k,1} = y_{k,0} - \dfrac{f(y_{k,0})}{\lambda_k f(y_{k,0}) + f'(y_{k,0})} \\ y_{k,2} = y_{k,1} - \dfrac{f(y_{k,1})}{f[y_{k,1}, y_{k,0}] + f[y_{k,1}, y_{k,0}, y_{k,0}](y_{k,1} - y_{k,0})} \\ \quad \cdots \\ y_{k,n} = y_{k,n-1} - \dfrac{f(y_{k,n-1})}{N(y_{k,n-1}, y_{k,n-2}, \cdots, y_{k,1}, y_{k,0})} \end{cases} \tag{17}$$

where $N(y_{k,n-1}, y_{k,n-2}, \cdots, y_{k,1}, y_{k,0}) = f[y_{k,n-1}, y_{k,n-2}] + \cdots + f[y_{k,n-1}, y_{k,n-2}, \cdots, y_{k,1}, y_{k,0}, y_{k,0}] \times (y_{k,n-1} - y_{k,n-2}) \cdots (y_{k,n-1} - y_{k,0})$, and the parameter$\lambda_k$ is calculated by using one of the Equations (14)–(16) and depends on the data available from the current and the previous iterations.

Lemma 1. Let H_m be the Hermite interpolating polynomial of the degree m that interpolates a function f at m distinct interpolation nodes $y_{k,0}, y_{k-1,n-1}, \cdots y_{k-1,n-m+1}$ contained in an interval $I$and the derivative $f^{(m+1)}$ is continuous in I and the Hermite interpolating polynomial $H_m(x)$ satisfied the condition $H_m(y_{k,0}) = f(y_{k,0}), H_m'(y_{k,0}) = f'(y_{k,0}), H_m(y_{k-1,n-j}) = f(y_{k-1,n-j})(j = 1, \cdots m-1)$. Define the errors $e_{k-1,n-j} = y_{k-1,n-j} - a(i = 1 \cdots m-1)$ and assume that

1. all nodes $y_{k,0}, y_{k-1,n-1,} \cdots y_{k-1,n-m+1}$ are sufficiently close to the zero a;
2. the condition $e_k = O(e_{k-1,n-1} \cdots e_{k-1,n-m+1})$ holds.

Then

$$H_m''(y_{k,0}) \sim 2f'(a)\left(c_2 - (-1)^{m-1} c_{m+1} \prod_{j=1}^{m-1} e_{k-1,n-j}\right) \tag{18}$$

and

$$\frac{H_m''(y_{k,0})}{2f'(y_{k,0})} \sim \left(c_2 - (-1)^{m-1} c_{m+1} \prod_{j=1}^{m-1} e_{k-1,n-j}\right) \tag{19}$$

Proof. The error of the Hermite interpolation can be expressed as follows

$$f(x) - H_m(x) = \frac{f^{(m+1)}(\xi)}{(m+1)!}(x - y_{k,0})^2 \prod_{j=1}^{m-1} (x - y_{k-1,n-j}) \; (\xi \in I) \tag{20}$$

Differentiating Equation (20) at the point $x = y_{k,0}$, we obtain

$$f''(y_{k,0}) - H_m''(y_{k,0}) = 2\frac{f^{(m+1)}(\xi)}{(m+1)!} \prod_{j=1}^{m-1} (y_{k,0} - y_{k-1,n-j}) \; (\xi \in I) \tag{21}$$

$$H_m''(y_{k,0}) = f''(y_{k,0}) - 2\frac{f^{(m+1)}(\xi)}{(m+1)!} \prod_{j=1}^{m-1} (y_{k,0} - y_{k-1,n-j}) \; (\xi \in I) \tag{22}$$

Taylor's series of derivatives of $f$at the point $y_{k,0} \in I$ and $\xi \in I$ about the zero a of $f$give

$$f'(y_{k,0}) = f'(a)(1 + 2c_2 e_{k,0} + 3c_3 e_{k,0}^2 + O(e_{k,0}^3)) \tag{23}$$

$$f''(y_{k,0}) = f'(a)(2c_2 + 6c_3 e_{k,0} + O(e_{k,0}^2)) \tag{24}$$

$$f^{(m+1)}(\xi) = f'(a)((m+1)!c_{m+1} + (m+2)!c_{m+1}e_\xi + O(e_\xi^2)) \tag{25}$$

where $e_\xi = \xi - a$.

Substituting Equations (24) and (25) into Equation (22), we have

$$H''_m(y_{k,0}) = 2f'(a)(c_2 - (-1)^{m-1}c_{m+1}\prod_{j=1}^{m-1} e_{k-1,n-j}) \tag{26}$$

and

$$\frac{H''_m(y_{k,0})}{2f'(y_{k,0})} \sim \left(c_2 - (-1)^{m-1}c_{m+1}\prod_{j=1}^{m-1} e_{k-1,n-j}\right) \tag{27}$$

The concept of the R-order of convergence [1] and the following assertion (see [8]) will be applied to estimate the convergence order of the iterative method with memory Equation (17). Now we can state the following convergence theorem for iterative method with memory Equation (17).

Theorem 3. Let the varying parameter λ_k in the iterative Equation (17) be calculated by Equation (14). If an initial approximation x_0 is sufficiently close to a simple zero a of $f(x)$, then the R-order of convergence of the n-point Equation (17) with memory is at least $2^n + 2^{n-3}$ for $n \geq 3$ and at least $(5+\sqrt{17})/2 \approx 4.5616$ for $n = 2$.

Proof. First, let us consider the case $n \geq 3$ and assume that the iterative sequences $\{y_{k,n}\}$ and $\{y_{k,n-1}\}$ have the R-order r and q, respectively, we have

$$e_{k+1} = e_{k,n} \sim D_{k,r}e_k^r \sim D_{k,r}(D_{k-1,r}e_{k-1}^r)^r = D_{k,r}D_{k-1,r}^r e_{k-1}^{r^2} \tag{28}$$

$$e_{k,n-1} \sim D_{k,q}e_k^q \sim D_{k,q}(D_{k-1,r}e_{k-1}^r)^q = D_{k,q}D_{k-1,r}^q e_{k-1}^{rq} \tag{29}$$

where $D_{k,j}$ $(j \in R)$ tends to the asymptotic error constant D_j when $k \to \infty$.

From Equation (13), we obtain the following error relations

$$e_{k,n-1} = y_{k,n-1} - a \sim (c_2 + \lambda)^{2^{n-3}} d_{n-1} e_k^{2^{n-1}} \tag{30}$$

$$e_{k+1} = y_{k,n} - a \sim (c_2 + \lambda)^{2^{n-2}} d_n e_k^{2^n} \tag{31}$$

Using the Lemma 1 for $m = 2$, we obtain

$$\lambda_k \sim -(c_2 + c_3 e_{k-1,n-1}) \tag{32}$$

Substituting Equation (32) into Equations (30) and (31) instead of λ, we have

$$\begin{aligned} e_{k,n-1} = y_{k,n-1} - a &\sim (-c_3)^{2^{n-3}} D_{k-1,q}^{2^{n-3}} e_{k-1}^{q2^{n-3}} d_{n-1}(D_{k-1,r}e_{k-1}^r)^{2^{n-1}} \\ &\sim (-c_3)^{2^{n-3}} D_{k-1,q}^{2^{n-3}} D_{k-1,r}^{2^{n-1}} d_{n-1} e_{k-1}^{r2^{n-1}+q2^{n-3}} \end{aligned} \tag{33}$$

$$\begin{aligned} e_{k+1} = y_{k,n} - a &\sim (-c_3 e_{k-1,n-1})^{2^{n-2}} d_n e_k^{2^n} \\ &\sim (-c_3)^{2^{n-2}} D_{k-1,q}^{2^{n-2}} D_{k-1,r}^{2^n} d_{n-1} e_{k-1}^{r2^n+q2^{n-2}} \end{aligned} \tag{34}$$

By comparing exponents of e_{k-1} appearing in two pairs of relations Equations (29)–(33) and Equations (28)–(34), we get the following system of equations

$$\begin{cases} r2^{n-1} + q2^{n-3} = rq \\ r2^{n} + q2^{n-2} = r^2 \end{cases} \tag{35}$$

The solution of the system Equation (35) is given by $r = 2^n + 2^{n-3}$ and $q = 2^{n-1} + 2^{n-4}$. Therefore, the R-order of the methods with memory Equation (17) is at least $r = 2^n + 2^{n-3}$ for $n \geq 3$. For example, the R-order of the three-point family Equation (17) is at least 9, the four-point family has the R-order at least 18, assuming that λ_k is calculated by Equation (14).

The case $n = 2$ differs from the previous analysis; Hermit's interpolating polynomial is constructed at the nodes $y_{k,0}, y_{k-1,1}$. Substituting Equation (32) into Equation (2) and Equation (8) instead of λ, we have

$$\begin{aligned} e_{k,1} =& y_{k,1} - a \sim (-c_3 e_{k-1,1}) d_1 e_k^2 \sim -c_3 D_{k-1,q} e_{k-1}^q d_1 (D_{k-1,r} e_{k-1}^r)^2 \\ &\sim -c_3 D_{k-1,q} d_1 D_{k-1,r}^2 e_{k-1}^{2r+q} \end{aligned} \tag{36}$$

$$\begin{aligned} e_{k+1} =& e_{k,2} = y_{k,2} - a \sim (-c_3 e_{k-1,1}) d_2 e_k^4 \sim -c_3 D_{k-1,q} e_{k-1}^q d_2 (D_{k-1,r} e_{k-1}^r)^4 \\ &\sim -c_3 D_{k-1,q} d_1 D_{k-1,r}^4 e_{k-1}^{4r+q} \end{aligned} \tag{37}$$

By comparing exponents of e_{k-1} appearing in two pairs of relations Equations (29)–(36) and Equations (28)–(37), we get the following system of equations

$$\begin{cases} 2r + q = rq \\ 4r + q = r^2 \end{cases} \tag{38}$$

Positive solution of the system Equation (38) is given by $r = (5 + \sqrt{17})/2$ and $q = (1 + \sqrt{17})/2$. Therefore, the R-order of the methods with memory Equations (17) with (14) is at least $(5 + \sqrt{17})/2 \approx 4.5616$ for $n = 2$.

Theorem 4. Let the varying parameter λ_k in the iterative Equation (17) be calculated by Equation (15). If an initial approximation x_0 is sufficiently close to a simple zero a of $f(x)$, then the R-order of convergence of the n-point methods Equation (17) with memory is at least $2^n + 2^{n-3} + 2^{n-4}$ for $n \geq 4$, at least $5 + \sqrt{21} \approx 9.5826$ for $n = 3$ and at least $(5 + \sqrt{21})/2 \approx 4.7913$ for $n = 2$.

Proof. The proof is similar to the Theorem 3.

Theorem 5. Let the varying parameter λ_k in the iterative Equation (17) be calculated by Equation (16). If an initial approximation x_0 is sufficiently close to a simple zero a of $f(x)$, then the R-order of convergence of the n-point Equation (17) with memory is at least $2^n + 2^{n-3} + 2^{n-4} + 2^{n-5}$ for $n \geq 5$, at least $10 + \sqrt{92} \approx 19.5917$ for $n = 4$ and at least $5 + \sqrt{23} \approx 9.7958$ for $n = 3$.

Proof. The proof is similar to the Theorem 3.

4. Numerical Results

Now, the new family Equation (4) without memory and the corresponding family Equation (17) with memory are employed to solve some nonlinear equations and compared with several known iterative methods. All algorithms are implemented using Symbolic Math Toolbox of MATLAB 7.0. For demonstration, we have selected three methods displayed below.

King's methods without memory (KM4, see [9]):

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = y_n - \dfrac{f(x_n)+\beta f(y_n)}{f(x_n)+(\beta-2)f(y_n)} \dfrac{f(y_n)}{f'(x_n)} \end{cases} \tag{39}$$

where $\beta \in R$.

Bi-Wu-Ren method without memory (BRM8, see [10]):

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ z_n = y_n - h(\dfrac{f(y_n)}{f(x_n)}) \dfrac{f(y_n)}{f'(x_n)} \\ x_{n+1} = z_n - \dfrac{f(x_n)+(\gamma+2)f(z_n)}{f(x_n)+\gamma f(z_n)} \dfrac{f(z_n)}{f[z_n,y_n]+f[x_n,x_n,z_n](z_n-y_n)} \end{cases} \tag{40}$$

where $h(t)$is a real-valued function satisfying the conditions$h(0) = 1, h'(0) = 2, h''(0) = 10$, $|h'''(0)| < \infty$ and $\gamma \in R$.

Petković-Ilić-Džunić method with memory (PD, see [12])

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f[x_n,z_n]}, \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f[x_n,z_n]} \left(1 + \dfrac{f(y_n)}{f(x_n)} + \dfrac{f(y_n)}{f(z_n)}\right) \end{cases} \tag{41}$$

where $z_n = x_n - \gamma_n f(x_n)$. The parameter$\gamma_n$ can be calculated by the following three formulas:

$$\gamma_n = (x_n - y_{n-1})/(f(x_n) - f(y_{n-1})) \tag{42}$$

$$\gamma_n = 1/(f[x_n, y_{n-1}] + f[x_n, x_{n-1}] - f[x_{n-1}, y_{n-1}]) \tag{43}$$

The absolute errors$|x_k - a|$in the first four iterations are given in Tables 1–4, where a is the exact root computed with 2400 significant digits. The computational order of convergence ρ is defined by [19]:

$$\rho \approx \frac{\ln(|x_{n+1} - x_n| / |x_n - x_{n-1}|)}{\ln(|x_n - x_{n-1}| / |x_{n-1} - x_{n-2}|)} \tag{44}$$

The iterative processes of the Equations (4) and (17) are given in Figure 1, where Equation (4) (n=1) is one-point method. The parameters of the Equations (4) and (17) are $\lambda = \lambda_0 = 1.0$. The initial value is $x_0 = -1.3$. The stopping criterium is $|f(x)| < 10^{-150}$. We will call $f(x)$ the nonlinear residual or residual. The Figure 1 is a semilog plot of residual history, the norm of the nonlinear residual against the iteration number.

Following test functions are used:

$$f_1(x) = xe^{x^2} - \sin^2(x) + 3\cos(x) + 5,\ a \approx -1.2076478271309189, x_0 = -1.3.$$

$$f_2(x) = x^5 + x^4 + 4x^2 - 15,\ a \approx 1.3474280989683050, x_0 = 1.6.$$

Table 1. Numerical results for $f_1(x)$ by the methods without memory.

Methods	$\|x_1 - a\|$	$\|x_2 - a\|$	$\|x_3 - a\|$	ρ
Equation (4) $n = 2, \lambda = 0.5$	0.32719E−4	0.57076E−18	0.52848E−73	4.0000005
Equation (4) $n = 2, \lambda = 1$	0.58111E−4	0.71445E−17	0.16328E−68	3.9999938
$KM4, \beta = 2$	0.24269E−3	0.13078E−13	0.11033E−54	3.9999864
$BRM8, h(t) = 1 + 2t + 5t^2, \gamma = 1$	0.40513E−6	0.32351E−48	0.53484E−385	8.0000001
Equation (4) $n = 3, \lambda = 1$	0.22673E−8	0.83510E−70	0.28282E−561	8.0000000
Equation (4) $n = 3, \lambda = 1.5$	0.18012E−9	0.75259E−83	0.69916E−670	8.0000000

Table 2. Numerical results for $f_2(x)$ by the methods without memory.

Methods	$\|x_1 - a\|$	$\|x_2 - a\|$	$\|x_3 - a\|$	ρ
Equation (4)$n = 2, \lambda = -1.5$	0.29673E−2	0.37452E−10	0.94752E−42	4.0001713
Equation (4)$n = 2, \lambda = -0.5$	0.27276E−4	0.11867E−19	0.42516E−81	4.0000025
$KM4, \beta = 0.5$	0.37189E−2	0.32631E−9	0.19533E−37	3.9993916
$BRM8, h(t) = 1 + 2t + 5t^2, \gamma = 1$	0.84179E−4	0.62964E−31	0.61512E−248	8.0000456
Equation (4)$n = 3, \lambda = -1$	0.34838E−7	0.19030E−62	0.15080E−504	8.0000000
Equation (4)$n = 3, \lambda = -0.5$	0.11873E−7	0.80149E−66	0.34562E−531	8.0000000

Table 3. Numerical results for $f_1(x)$ by the methods with memory.

Methods	$\|x_1 - a\|$	$\|x_2 - a\|$	$\|x_3 - a\|$	ρ
Equation (42) $- PD, \gamma_0 = -0.01$	0.10690E−2	0.10554E−12	0.24668E−58	4.5605896
Equation (43) $- PD, \gamma_0 = -0.01$	0.10690E−2	0.58225E−14	0.18875E−67	4.7487424
Equation (14) $-$ (17)$, n = 2, \lambda_0 = 0.5$	0.32719E−4	0.42649E−19	0.26035E−87	4.5827899
Equation (15) $-$ (17)$, n = 2, \lambda_0 = 0.5$	0.32719E−4	0.47493E−20	0.16676E−96	4.8272294
Equation (14) $-$ (17)$, n = 2, \lambda_0 = 1$	0.58111E−4	0.25364E−18	0.61743E−84	4.5691828
Equation (15) $-$ (17)$, n = 2, \lambda_0 = 1$	0.58111E−4	0.28197E−19	0.69228E−93	4.8066915
Equation (14) $-$ (17)$, n = 3, \lambda_0 = 1$	0.22673E−8	0.14247E−76	0.38886E−690	8.9963034
Equation (15) $-$ (17)$, n = 3, \lambda_0 = 1$	0.22673E−8	0.53419E−81	0.96778E−777	9.5795515
Equation (16) $-$ (17)$, n = 3, \lambda_0 = 1$	0.22673E−8	0.45910E−83	0.96092E−815	9.7957408
Equation (14) $-$ (17)$, n = 3, \lambda_0 = 1.5$	0.18012E−9	0.49194E−86	0.27126E−775	9.0024260
Equation (15) $-$ (17)$, n = 3, \lambda_0 = 1.5$	0.18012E−9	0.13193E−91	0.20518E−878	9.5794268
Equation (16) $-$ (17)$, n = 3, \lambda_0 = 1.5$	0.18012E−9	0.11706E−93	0.17692E−918	9.7974669

Table 4. Numerical results for $f_2(x)$by the methods with memory.

Methods	$\lvert x_1 - a \rvert$	$\lvert x_2 - a \rvert$	$\lvert x_3 - a \rvert$	ρ
Equation (42) $- PD, \gamma_0 = -0.01$	0.14930E−1	0.54292E−8	0.20342E−37	4.5697804
Equation (43) $- PD, \gamma_0 = -0.01$	0.14930E−1	0.32753E−9	0.16659E−45	4.7387964
Equation (14) $-$ (17), $n = 2, \lambda_0 = -1.5$	0.29673E−2	0.10381E−11	0.90169E−55	4.5538013
Equation (15) $-$ (17), $n = 2, \lambda_0 = -1.5$	0.29673E−2	0.13370E−13	0.29875E−67	4.7285160
Equation (14) $-$ (17), $n = 2, \lambda_0 = -0.5$	0.27276E−4	0.76276E−20	0.21310E−91	4.6005252
Equation (15) $-$ (17), $n = 2, \lambda_0 = -0.5$	0.27276E−4	0.62055E−21	0.70672E−102	4.8635157
Equation (14) $-$ (17), $n = 3, \lambda_0 = -1$	0.34838E−7	0.12841E−67	0.15487E−611	9.0002878
Equation (15) $-$ (17), $n = 3, \lambda_0 = -1$	0.34838E−7	0.34679E−73	0.10151E−705	9.5835521
Equation (16) $-$ (17), $n = 3, \lambda_0 = -1$	0.34838E−7	0.41211E−75	0.11560E−741	9.8127640
Equation (14) $-$ (17), $n = 3, \lambda_0 = -0.5$	0.11873E−7	0.35119E−73	0.13260E−661	8.9795793
Equation (15) $-$ (17), $n = 3, \lambda_0 = -0.5$	0.11873E−7	0.43166E−77	0.67183E−743	9.5883270
Equation (16) $-$ (17), $n = 3, \lambda_0 = -0.5$	0.11873E−7	0.45981E−83	0.29759E−820	9.7754885

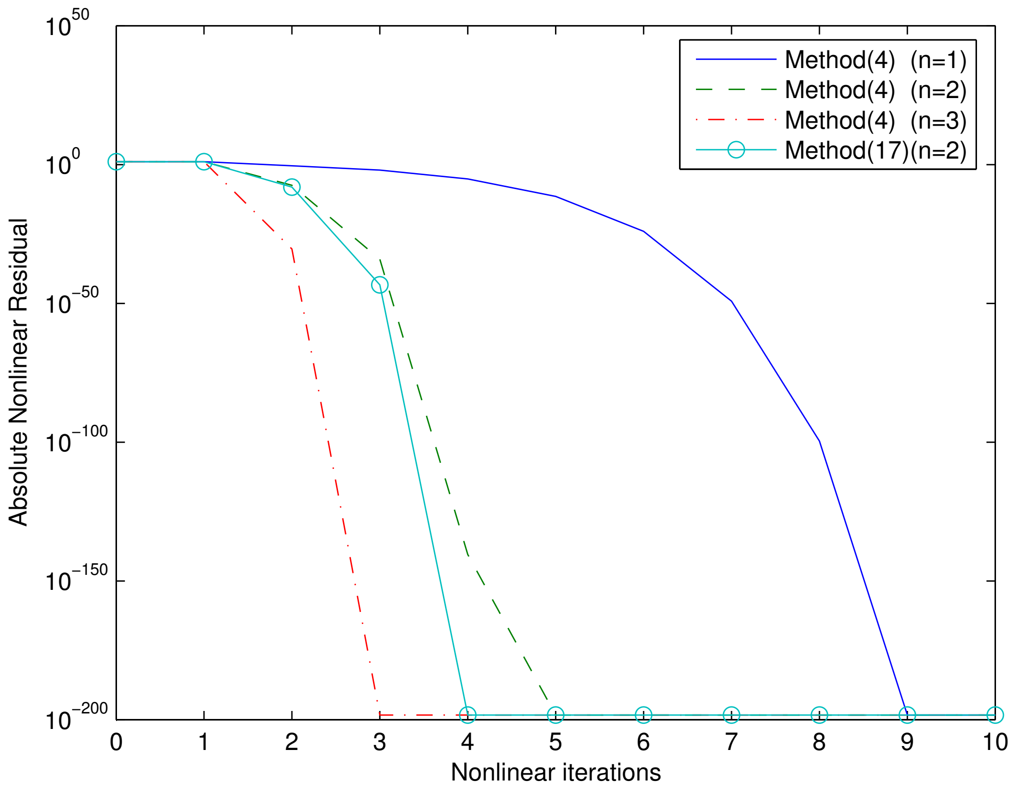

Figure 1. Iterative processes of different methods for the function $f_1(x)$.

5. Conclusions

Figure 1 shows that the convergence speed of the multipoint iterative method is faster than the one-point iterative method. As shown in Tables 1 and 2, the results obtained with our methods without memory are better than the other methods without memory. From the results displayed in

Tables 3 and 4, it can be concluded that the convergence of the tested multipoint Equation (17) with memory is remarkably fast. The R-order of convergence of the family Equation (17) with memory is increased by applying a self-accelerating parameter given by Equations (14)–(16). In addition, above all, the increase of convergence order is obtained without any additional function evaluations, which indicates a very high computational efficiency of our methods with memory.

Acknowledgments

The project supported by the National Natural Science Foundation of China (Nos. 11371071 and 11201037), the PhD Start-up Fund of Liaoning Province of China (Nos. 20141137 and 20141139), the Liaoning BaiQianWan Talents Program (No.2013921055) and the Educational Commission Foundation of Liaoning Province of China (Nos. L2014443,L2014444 and L2015012).

Author Contributions

Xiaofeng Wang and Yuping Qin conceived and designed the experiments; Weiyi Qian and Sheng Zhang analyzed the data; Xiaofeng Wang and Xiaodong Fan wrote the paper.

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Ortega, J.M.; Rheinbolt, W.C. *Iterative Solution of Nonlinear Equations in Several Variables*; Academic Press: New York, NY, USA, 1970.
2. Wu, X. A new continuation Newton-like method and its deformation. *Appl. Math. Comput.* **2000**, *112*, 75–78.
3. Wang, X.; Zhang, T. A new family of Newton-type iterative methods with and without memory for solving nonlinear equations. *Calcolo* **2014**, *51*, 1–15.
4. Wang, X.; Liu, L. Modified Ostrowski's method with eighth-order convergence and high efficiency index. *Appl. Math. Lett.* **2010**, *23*, 549–554.
5. Kou, J.; Wang, X.; Li, Y. Some eighth-order root-finding three-step methods. *Commn. Nonli. Sci. Numer. Simulat.* **2010**, *15*, 536–544.
6. Petković, M.S. Remarks on "on a general class of multipoint root-finding methods of high computational efficiency". *SIAM J. Numer. Anal.* **2011**, *49*, 1317–1319.
7. Kung, H.T.; Traub, J.F. Optimal order of one-point and multipoint iterations. *J. Appl. Comput. Math.* **1973**, *10*, 643–651.
8. Alefeld, G.; Herzberger, J. *Introduction to Interval Computation*, Academic Press: New York, NY, USA, 1983.
9. King, R.F. A family of fourth order methods for nonlinear equations. *SIAM J. Numer. Anal.* **1973**, *10*, 876–879.

10. Bi, W.; Wu, Q.; Ren, H. A new family of eighth-order iterative methods for solving nonlinear equations. *Appl. Math. Comput.* **2009**, *214*, 236–245.
11. Petković, M.S.; Ilić, S.; Džuni, J. Derivative free two-point methods with and without memory for solving nonlinear equations. *Appl. Math. Comput.* **2010**, *217*, 1887–1895.
12. Thukral, R.; Petković, M.S. A family of three-point methods of optimal order for solving nonlinear equations. *J. Comp. Appl. Math.* **2010**, *223*, 2278–2284.
13. Wang, X.; Zhang, T. A family of Steffensen type methods with seventh-order convergence. *Numer. Algor.* **2013**, *62*, 429–444.
14. Wang, X.; Džunić, J.; Zhang, T. On an efficient family of derivative free three-point methods for solving nonlinear equations. *Appl. Math. Comput.* **2012**, *219*, 1749–1760.
15. Soleymani, F.; Sharifi, M.; Mousavi, B.S. An inprovement of Ostrowski's and King's techniques with optimal convergence order eight. *J. Optim. Theory. Appl.* **2012**, *153*, 225–236.
16. Yun, B.I. A non-iterative method for solving non-linear equations. *Appl. Math. Comput.* **2008**, *198*, 691–699.
17. Sharma, J.R.; Sharma, R. A new family of modified Ostrowski's methods with accelerated eighth order convergence. *Numer. Algor.* **2010**, *54*, 445–458.
18. Cordero, A.; Ezquerro, J.A.; Hernánde-Verón, M.A.; Torregrosa, J.R. On the local convergence of a fifth-order iterative method in Banach spaces. *Appl. Math. Comput.* **2015**, *215*, 396–403.
19. Grau-Sánchez, M.; Noguera, M.; Gutiérrez, J.M. On some computational orders of convergence. *Appl. Math. Lett.* **2010**, *23*, 472–478.