

SOR- Steffensen-Newton Method to Solve Systems of Nonlinear Equations

M. T. Darvishi*, Norollah Darvishi

Department of Mathematics, Razi University, Kermanshah, 67149, Iran

Abstract In this paper, we present SOR-Steffensen-Newton (SOR-SN) algorithm to solve systems of nonlinear equations. We study the convergence of the method. The computational aspects of the method is also studied using some numerical experiment. In comparison of new method with SOR-Newton, SOR-Steffensen and SOR-Secant methods, our method are better in CPU time and number of iterations.

Keywords Nonlinear System, SOR-Newton Method, SOR-Steffensen Method, SOR-Secant Method, CPU Time

1. Introduction

There are cases where thousands of nonlinear equations depending on some independent variables must be solved effectively. Consider the nonlinear system

$$\mathbf{F}(\mathbf{x}) = 0 \quad (1)$$

where $\mathbf{F} : \mathbb{R}^n \rightarrow \mathbb{R}^n$, that is, a system with n equations and n unknowns. Finding roots of systems of nonlinear equations efficiently is of major importance and has widespread applications in numerical and applied mathematics. There are many approaches to solve (1). The second order Newton's method is one of the most common iterative methods adopted for finding approximate solutions of nonlinear system (1) (for more details see[1]). The Newton's method to solve (1) is an important and basic method which converges quadratically. Recently, some third and fourth order iterative methods have been proposed and analyzed for solving systems of nonlinear equations that improve some classical methods such as the Newton's method and Chebyshev-Halley methods. It has been demonstrated that the methods are efficient, and can compete with Newton's method, for more details see[2-10]. To solve system (1), Darvishi and Barati[11-13] presented some high order iterative methods, free from second order derivative of function \mathbf{F} . Darvishi[14,15] presented some multi-step iterative methods, free from second order derivative of function \mathbf{F} . Frontini and Sormani[16] obtained a third-order method based on a quadrature formula to solve systems of nonlinear equations. Babajee et al.[17] proposed a fourth order iterative method to solve system (1).

One-step SOR-Newton method to solve nonlinear system (1) which presented in[1] is as follows:

$$x_i^{(k+1)} = x_i^{(k)} - \omega \frac{F_i(\mathbf{x}^{(k,i-1)})}{F_{ii}(\mathbf{x}^{(k,i-1)})} \quad i = 1, \dots, n$$

$$k = 0, 1, 2, \dots$$

where $F_{ii} = \partial F_i / \partial x_i$, ω is the relaxation parameter in SOR method and

$$\mathbf{x}^{(k,0)} = (x_1^{(k,0)}, x_2^{(k,0)}, \dots, x_n^{(k,0)})^t = \mathbf{x}^{(k)}$$

$$\mathbf{x}^{(k,\ell)} = (x_1^{(k,\ell)}, x_2^{(k,\ell)}, \dots, x_n^{(k,\ell)})^t =$$

$$(x_1^{(k+1)}, x_2^{(k+1)}, \dots, x_j^{(k+1)}, x_{j+1}^{(k)}, \dots, x_n^{(k)})^t$$

$$\text{for } j = 1, 2, \dots, n-1.$$

It is obvious that we have

$$x_\ell^{(k,\ell)} = x_\ell^{(k,j-1)}, \quad \ell \neq j, \quad \ell = 1, 2, \dots, n.$$

The aim of this paper is to introduce the SOR-SN algorithm. To achieve this, we follow the two-step Steffensen type method which presented by Ren et al[18]. They have presented a class of one-parameter Steffensen type methods with fourth-order convergence to solve nonlinear equations

$$f(x)=0 \quad (2)$$

where f is a real function. We extend their method to solve systems of nonlinear equations and then combine the new method and SOR-Newton method to obtain the mixed method SOR-SN to solve (1). The convergence analysis of method will be presented. Also, we compare CPU time and number of iterations of SOR-SN method and another SOR methods to solve some nonlinear systems.

2. SOR- SN Algorithm

* Corresponding author:

darvishimt@yahoo.com (M. T. Darvishi)

Published online at <http://journal.sapub.org/am>

Copyright © 2012 Scientific & Academic Publishing. All Rights Reserved

In this section we introduce the SOR-SN algorithm, to solve systems of nonlinear equations. The SOR-SN method to solve $\mathbf{F}(\mathbf{x}) = 0$ for $\mathbf{F} = (F_1, F_2, \dots, F_n)^t : \mathbb{R}^n \rightarrow \mathbb{R}^n$

is defined by solving the following equation for x_ℓ

$$F_\ell(x_1^{(k+1)}, \dots, x_{\ell-1}^{(k+1)}, x_\ell, x_{\ell+1}^{(k)}, \dots, x_n^{(k)}) = 0. \quad (3)$$

We set

$$\mathbf{x}^{(k,\ell)} = (x_1^{(k+1)}, \dots, x_{\ell-1}^{(k+1)}, x_\ell^{(k)}, x_{\ell+1}^{(k)}, \dots, x_n^{(k)})^t$$

Then we obtain x_ℓ from (3) as follows

$$x_\ell = y_\ell^n - \frac{F_\ell(x_1^{(k+1)}, \dots, x_{\ell-1}^{(k+1)}, y_\ell^n, x_{\ell+1}^{(k)}, \dots, x_n^{(k)})}{F_{\ell\ell}(x_1^{(k+1)}, \dots, x_{\ell-1}^{(k+1)}, y_\ell^n, x_{\ell+1}^{(k)}, \dots, x_n^{(k)})}$$

where

$$y_\ell^n = x_\ell^{(k)} - \frac{F_\ell(\mathbf{x}^{(k,\ell-1)})}{F_\ell(\mathbf{x}^{(k,\ell-1)}) + F_\ell(\mathbf{x}^{(k,\ell-1)})e^\ell - F_\ell(\mathbf{x}^{(k,\ell-1)})}$$

whereas $e^\ell = (0, 0, \dots, 1, 0, \dots, 0)^t$. Finally, we set

$$x_\ell^{(k+1)} = y_\ell^n + \omega(x_\ell - y_\ell^n)$$

or

$$x_\ell^{(k+1)} = y_\ell^n - \omega \frac{F_\ell(x_1^{(k+1)}, \dots, x_{\ell-1}^{(k+1)}, y_\ell^n, x_{\ell+1}^{(k)}, \dots, x_n^{(k)})}{F_{\ell\ell}(x_1^{(k+1)}, \dots, x_{\ell-1}^{(k+1)}, y_\ell^n, x_{\ell+1}^{(k)}, \dots, x_n^{(k)})}, \quad (4)$$

for $\ell = 1, \dots, n$.

2.1. Convergence Analysis

In this part, we state and prove convergence theorem for our SOR-SN method. The proof is similar to proof of convergency of SOR algorithm in [19]. First, to follow [19], we need some assumptions. These assumptions are as follows

a) D is a convex set in \mathbb{R}^n .

b) For system $\mathbf{F}(\mathbf{x}) = 0$, \mathbf{F} is gradient of ϕ , namely, $\text{grad}\phi = \mathbf{F}$ and function $\phi : D \rightarrow \mathbb{R}$ is strictly convex.

c) On domain D , function ϕ is twice continuously differentiable.

d) Set $S_\gamma = \{\mathbf{x} \in D : \phi(\mathbf{x}) \leq \gamma\}$ for some $\gamma \in \mathbb{R}$ is nonempty and compact. Note that $\gamma \in \mathbb{R}$ exists and set S_γ is nonempty and compact.

e) $F_{ii}(\mathbf{y}) \neq 0$ for $i = 1, 2, \dots, n$ and $\mathbf{y} \in S_\gamma$, unless \mathbf{y} is the point at which ϕ attains its minimum, where $[F_{ij}(\mathbf{y})]$ for $i, j = 1, 2, \dots, n$ is Hessian matrix of ϕ at \mathbf{y} .

Assumptions a) and b) show that the Hessian matrix of ϕ is positive definite. It is obvious that sets S_γ are convex for all $\gamma \in \mathbb{R}$. Assumption d) shows that ϕ attains its minimum at some point $\mathbf{x}^* \in D$. If assumptions a), b) and c) hold and

function ϕ attains its minimum at some point $\mathbf{x}^* \in D$ then assumption d) is nontrivially satisfied, that is, there exists $\gamma \in \mathbb{R}$ such that $\gamma > \phi(\mathbf{x}^*)$ and S_γ is compact also ϕ is a non-increasing function. From b) the minimum point \mathbf{x}^* is unique. At last, a point $\mathbf{x}^* \in D$ is the minimum point of function ϕ if and only if $\text{grad}\phi(\mathbf{x}^*) = 0$.

Now, we can present the following theorem on convergency of SOR-SN method.

Theorem 1. If sequence $\{\mathbf{x}^{(k)}\}$ be generated by

$$x_j^{(k+1)} = x_j^{(k)}, \quad j \neq \ell$$

and

$$x_\ell^{(k+1)} = y_\ell^n - \omega \frac{F_\ell(x_1^{(k+1)}, \dots, x_{\ell-1}^{(k+1)}, y_\ell^n, x_{\ell+1}^{(k)}, \dots, x_n^{(k)})}{F_{\ell\ell}(x_1^{(k+1)}, \dots, x_{\ell-1}^{(k+1)}, y_\ell^n, x_{\ell+1}^{(k)}, \dots, x_n^{(k)})}$$

where

$$y_\ell^n = x_\ell^{(k)} - \frac{F_\ell(\mathbf{x}^{(k)})}{F_\ell(\mathbf{x}^{(k)}) + F_\ell(\mathbf{x}^{(k)})e^\ell - F_\ell(\mathbf{x}^{(k)})}$$

in fact, we have

$$x_\ell^{(k+1)} = x_\ell^{(k)} - \frac{F_\ell(\mathbf{x}^{(k)})}{F_\ell(\mathbf{x}^{(k)}) + F_\ell(\mathbf{x}^{(k)})e^\ell - F_\ell(\mathbf{x}^{(k)})} - \omega \frac{F_\ell(\mathbf{y}^n)}{F_{\ell\ell}(\mathbf{y}^n)} \quad (5)$$

where

$$\mathbf{y}^n = (x_1^{(k+1)}, \dots, x_{\ell-1}^{(k+1)}, y_\ell^n, x_{\ell+1}^{(k)}, \dots, x_n^{(k)})^t$$

Also if $\{I_\ell^k\}_{k=0}^\infty$, $\ell = 1, 2, \dots, n$, be defined by

$$I_\ell^k = \{\mathbf{x} : \phi(\mathbf{x}) \leq \phi(\mathbf{x}^{(k,\ell-1)}) \text{ and } x_j = x_j^{(k,\ell-1)}, j \neq \ell\}$$

and δ_k be defined by $\delta_k = \frac{D_k}{F_{\ell\ell}(\mathbf{x}^{(k)})}$, where

$$D_k = \max \{F_{\ell\ell}(\mathbf{y}) : F_{\ell\ell}(\mathbf{y}) \leq F_{\ell\ell}(\mathbf{x}^{(k)}), \mathbf{y} \in I_\ell^k\},$$

and for some $\delta > 0$ which satisfies in

$$0 < \delta \leq \omega \leq (1 - \frac{\delta_k}{2}) - \delta < (1 - \frac{\delta_k}{2}) < 1, \quad k = 0, 1, 2,$$

then for any $\mathbf{x}^{(0)} \in S_\gamma = \{\mathbf{x} \in D : \phi(\mathbf{x}) \leq \gamma\}$, sequence $\{\mathbf{x}^{(k)}\}$ is well-defined and converges to \mathbf{x}^* .

Proof. We first prove that sequence $\{\mathbf{x}^{(k)}\}$ is well-defined. By consideration of structure of sets I_1^0 and I_2^0 we have

$$I_1^0 = \{\mathbf{x} : \phi(\mathbf{x}) \leq \phi(\mathbf{x}^{(0,0)}), x_j = x_j^{(0,0)}, j \neq 1\}$$

$$I_2^0 = \{\mathbf{x} : \phi(\mathbf{x}) \leq \phi(\mathbf{x}^{(0,1)}), x_j = x_j^{(0,1)}, j \neq 2\}.$$

Note that we have

$$\mathbf{x}^{(0)} = \mathbf{x}^{(0,0)} = (x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)})^t$$

$$\mathbf{x}^{(0,1)} = (x_1^{(1)}, x_2^{(0)}, \dots, x_n^{(0)})^t,$$

this shows that $x_j^{(1)} = x_j^{(0)}$, $j \neq 1$. But sequence $\{\phi(\mathbf{x}^{(k)})\}$ is non-increasing, thus $\phi(\mathbf{x}^{(0,1)}) \leq \phi(\mathbf{x}^{(0,0)})$, from this we have $\mathbf{x}^{(0,1)} \in I_1^0 \subset S_\gamma$. In a similar manner and by mathematical induction we can prove that all terms of sequence $\{\mathbf{x}^{(k)}\}$ are contained in S_γ hence the sequence is well-defined. We now prove that $\{\mathbf{x}^{(k)}\}$ converges. By using Taylor's theorem on function ϕ for $\mathbf{y} \in (\mathbf{x}^{(k)}, \mathbf{x}^{(k+1)}) \subset I_\ell^k$ (where $(\mathbf{x}^{(k)}, \mathbf{x}^{(k+1)})$ denotes the open line segment joining $\mathbf{x}^{(k)}$ and $\mathbf{x}^{(k+1)}$) we have

$$\begin{aligned} \phi(\mathbf{x}^{(k+1)}) - \phi(\mathbf{x}^{(k)}) &= F_\ell(\mathbf{x}^{(k)}) (x_\ell^{(k+1)} - x_\ell^{(k)}) + \frac{1}{2} F_{\ell\ell}(\mathbf{y}) (x_\ell^{(k+1)} - x_\ell^{(k)})^2 \\ \phi(\mathbf{x}^{(k)}) - \phi(\mathbf{x}^{(k+1)}) &= F_\ell(\mathbf{x}^{(k)}) (x_\ell^{(k)} - x_\ell^{(k+1)}) - \frac{1}{2} F_{\ell\ell}(\mathbf{y}) (x_\ell^{(k)} - x_\ell^{(k+1)})^2 \\ &\geq F_\ell(\mathbf{x}^{(k)}) (x_\ell^{(k)} - x_\ell^{(k+1)}) - \frac{\delta_k}{2} F_{\ell\ell}(\mathbf{x}^{(k)}) (x_\ell^{(k)} - x_\ell^{(k+1)})^2. \end{aligned} \quad (6)$$

Note that in the Steffensen's method we have

$$F_{\ell\ell}(\mathbf{x}^{(k)}) \approx \frac{F_\ell(\mathbf{x}^{(k)}) + F_\ell(\mathbf{x}^{(k)})e^\ell - F_\ell(\mathbf{x}^{(k)})}{F_\ell(\mathbf{x}^{(k)})}$$

thus from (5) we have

$$F_\ell(\mathbf{x}^{(k)}) = (x_\ell^{(k)} - x_\ell^{(k+1)}) F_{\ell\ell}(\mathbf{x}^{(k)}) - \omega \frac{F_{\ell\ell}(\mathbf{x}^{(k)})}{F_{\ell\ell}(\mathbf{y}^n)} F_\ell(\mathbf{y}^n) \quad (7)$$

setting (7) in (6) yields

$$\begin{aligned} \phi(\mathbf{x}^{(k)}) - \phi(\mathbf{x}^{(k+1)}) &\geq F_{\ell\ell}(\mathbf{x}^{(k)}) (x_\ell^{(k)} - x_\ell^{(k+1)})^2 \\ &- \omega \frac{F_{\ell\ell}(\mathbf{x}^{(k)})}{F_{\ell\ell}(\mathbf{y}^n)} F_\ell(\mathbf{y}^n) (x_\ell^{(k)} - x_\ell^{(k+1)}) - \frac{\delta_k}{2} F_{\ell\ell}(\mathbf{x}^{(k)}) (x_\ell^{(k)} - x_\ell^{(k+1)})^2. \end{aligned} \quad (8)$$

From the convergence of the Steffensen's method for \mathcal{E} which satisfies in

$$0 < \mathcal{E} < F_{\ell\ell}(\mathbf{y}^n) |x_\ell^{(k)} - x_\ell^{(k+1)}|$$

we have

$$|F_\ell(\mathbf{y}^n)| \leq \mathcal{E} < F_{\ell\ell}(\mathbf{y}^n) |x_\ell^{(k)} - x_\ell^{(k+1)}|.$$

Consequently, from this and (8) we have

$$\begin{aligned} \phi(\mathbf{x}^{(k)}) - \phi(\mathbf{x}^{(k+1)}) &\geq F_{\ell\ell}(\mathbf{x}^{(k)}) (x_\ell^{(k)} - x_\ell^{(k+1)})^2 \\ &- \omega F_{\ell\ell}(\mathbf{x}^{(k)}) |x_\ell^{(k)} - x_\ell^{(k+1)}| (x_\ell^{(k)} - x_\ell^{(k+1)}) \\ &- \frac{\delta_k}{2} F_{\ell\ell}(\mathbf{x}^{(k)}) (x_\ell^{(k)} - x_\ell^{(k+1)})^2 \\ &= F_{\ell\ell}(\mathbf{x}^{(k)}) (x_\ell^{(k)} - x_\ell^{(k+1)})^2 \pm \omega F_{\ell\ell}(\mathbf{x}^{(k)}) (x_\ell^{(k)} - x_\ell^{(k+1)})^2 \\ &- \frac{\delta_k}{2} F_{\ell\ell}(\mathbf{x}^{(k)}) (x_\ell^{(k)} - x_\ell^{(k+1)})^2. \end{aligned}$$

Thus

$$\phi(\mathbf{x}^{(k)}) - \phi(\mathbf{x}^{(k+1)}) \geq m (x_\ell^{(k)} - x_\ell^{(k+1)})^2 ((1 - \frac{\delta_k}{2}) \pm \omega)$$

where $m = \text{glb}\{F_{\ell\ell}(\mathbf{x}^{(k)}), k = 0, 1, 2, \dots\}$. Since the Hessian matrix of ϕ is positive semidefinite, hence $m \geq 0$. We consider two cases for m :

Case 1. $m = 0$

By assumption e) there exists some subsequence $\{\mathbf{y}^{(k)}\}$ converges to \mathbf{x}^* and we know that $\mathbf{y}^{(k)} \in (\mathbf{x}^{(k)}, \mathbf{x}^{(k+1)})$ and as $\{\phi(\mathbf{x}^{(k)})\}$ is a non-increasing sequence, we have $\phi(\mathbf{x}^{(k)}) \geq \phi(\mathbf{y}^{(k)}) \geq \phi(\mathbf{x}^{(k+1)})$. Therefore, by our assumption, as ϕ is a continuous function, we have $\mathbf{x}^{(k)} \rightarrow \mathbf{x}^*$.

Case 2. $m > 0$

We know that $\delta > 0$ satisfies in

$$0 < \delta \leq \omega \leq (1 - \frac{\delta_k}{2}) - \delta < (1 - \frac{\delta_k}{2}) < 1,$$

we consider two possible cases as follows:

Case 1.

$$\phi(\mathbf{x}^{(k)}) - \phi(\mathbf{x}^{(k+1)}) \geq m (x_\ell^{(k)} - x_\ell^{(k+1)})^2 ((1 - \frac{\delta_k}{2}) - \omega).$$

We know that the sequence $\{\phi(\mathbf{x}^{(k)})\}$ is non-increasing and bounded from below, hence it is a convergent sequence. Then $|x_\ell^{(k)} - x_\ell^{(k+1)}| \rightarrow 0$, we suppose that $\bar{\mathbf{x}}$ is a limit point of $\{\mathbf{x}^{(k)}\}$ and we consider the following sets $I = \{\ell : F_\ell(\bar{\mathbf{x}}) = 0, \ell = 1, 2, \dots, n\}$ and $J = \{1, 2, \dots, n\} \setminus I$. From (7) we have

$$\begin{aligned} F_\ell(\mathbf{x}^{(k)}) &= (x_\ell^{(k)} - x_\ell^{(k+1)}) F_{\ell\ell}(\mathbf{x}^{(k)}) - \omega \frac{F_{\ell\ell}(\mathbf{x}^{(k)})}{F_{\ell\ell}(\mathbf{y}^n)} F_\ell(\mathbf{y}^n) \\ |F_\ell(\mathbf{x}^{(k)})| &\leq |x_\ell^{(k)} - x_\ell^{(k+1)}| |F_{\ell\ell}(\mathbf{x}^{(k)})| + \omega \frac{F_{\ell\ell}(\mathbf{x}^{(k)})}{F_{\ell\ell}(\mathbf{y}^n)} |F_\ell(\mathbf{y}^n)| \end{aligned}$$

from the convergency of the Steffensen's method we have

$$\begin{aligned} |F_\ell(\mathbf{y}^n)| &\leq |x_\ell^{(k)} - x_\ell^{(k+1)}| |F_{\ell\ell}(\mathbf{y}^n)| \\ |F_\ell(\mathbf{x}^{(k)})| &\leq (1 + \omega) |x_\ell^{(k)} - x_\ell^{(k+1)}| |F_{\ell\ell}(\mathbf{x}^{(k)})| \end{aligned}$$

As $|x_\ell^{(k)} - x_\ell^{(k+1)}| \rightarrow 0$ and $F_{\ell\ell}(\mathbf{x}^{(k)})$ is bounded, hence we have

$$\lim_{k \rightarrow \infty} (x_\ell^{(k)} - x_\ell^{(k+1)}) F_{\ell\ell}(\mathbf{x}^{(k)}) = 0.$$

Thus

$$\lim_{k \rightarrow \infty} F_{\ell\ell}(\mathbf{x}^{(k)}) = 0.$$

This shows that I is a nonempty set. If J be an empty set, hence $\bar{\mathbf{x}} = \mathbf{x}^*$ and as $\{\phi(\mathbf{x}^{(k)})\}$ is a non-increasing sequence, $\mathbf{x}^{(k)} \rightarrow \mathbf{x}^*$. Else if J be a nonempty set, thus

$J = \{j_1, \dots, j_l\}$, $I = \{i_1, \dots, i_{n-l}\}$. For $i = 1, 2, \dots, n$, we define the following sets

$$H_i = \{\mathbf{u} \in S_\gamma : F_i(\mathbf{u}) = 0\}.$$

Clearly sets H_i are closed and nonempty sets, because $F_i(\bar{\mathbf{x}}) = 0$ and $\bar{\mathbf{x}} \in S_\gamma$, hence $\bar{\mathbf{x}} \in H_i$.

Also we define $H_J = \cup_{j \in J} H_j$. Since H_J is a closed set then there exists $\tau > 0$ such that for $\mathbf{y} \in H_J$, we have $|\bar{\mathbf{x}} - \mathbf{y}| \geq \tau$. Remember that $(\mathbf{x}^{(k)} - \mathbf{x}^{(k+1)}) \rightarrow 0$ hence, there is an integer N such that for $k \geq N$ we have $|\mathbf{x}^{(k)} - \mathbf{x}^{(k+1)}| < \frac{\tau}{2}$. Let $\Lambda = \min_{j \in J} (|F_j(\bar{\mathbf{x}})|)$. From the continuity of F_j there exists $\varepsilon > 0$ such that $|\bar{\mathbf{x}} - \mathbf{y}| < \varepsilon$

implies that $F_j(\mathbf{y}) > \frac{\Lambda}{2}$.

We define set $B_\varepsilon \subset \mathbb{R}^n$ such that for all $\mathbf{x} \in B_\varepsilon$, the following relations hold

$$|(x_{j_1}, \dots, x_{j_L})^t - (\bar{x}_{j_1}, \dots, \bar{x}_{j_L})^t| < \varepsilon' \quad (9)$$

and

$$\phi(\mathbf{x}) < \phi(\bar{\mathbf{x}}) + \varepsilon' = \bar{\phi} + \varepsilon'. \quad (10)$$

Note that for any $\varepsilon > 0$ we may select $\varepsilon'(\varepsilon) > 0$ such that if $\mathbf{x} \in B_\varepsilon$, then $|\mathbf{x} - \bar{\mathbf{x}}| < \varepsilon$, otherwise for every sequence $\{\varepsilon'_i\}$ which reduces to zero, there exists $\{\mathbf{z}^{(l)}\}$ such that $\mathbf{z}^{(l)} \in B_\varepsilon$, and $|\mathbf{z}^{(l)} - \bar{\mathbf{x}}| \geq \varepsilon$. All such $\mathbf{z}^{(l)}$ are contained in $S_{\bar{\phi} + \varepsilon_1}$, which is a compact set. This results that,

there exists a limit point $\mathbf{z}^{(0)}$ of $\{\mathbf{z}^{(l)}\}$ which is such that $|\mathbf{z}^{(0)} - \bar{\mathbf{x}}| \geq \varepsilon$. We have from (9) that

$$|(z_{j_1}^{(0)}, \dots, z_{j_L}^{(0)}) - (\bar{x}_{j_1}, \dots, \bar{x}_{j_L})| = 0.$$

And also from (10) we have

$$\phi(\mathbf{z}^{(0)}) = \bar{\phi} = \phi(\bar{\mathbf{x}})$$

hence

$$\langle \text{grad} \phi(\bar{\mathbf{x}}), \bar{\mathbf{x}} - \mathbf{z}^{(0)} \rangle = \phi(\bar{\mathbf{x}}) - \phi(\mathbf{z}^{(0)}) = 0,$$

where $\langle \mathbf{x}, \mathbf{y} \rangle$ denotes the inner product of vectors \mathbf{x} and \mathbf{y} . This leads to a contradiction, because ϕ is a strictly convex function. Thus there exists $\varepsilon' > 0$ such that $\mathbf{y} \in B_\varepsilon$ implies that $|\bar{\mathbf{x}} - \mathbf{y}| < \varepsilon$. Also from

$\lim_{k \rightarrow \infty} F_{\ell\ell}(\mathbf{x}^{(k)}) = 0$, there exists N_1 such that for

$k \geq N_1$ we have $|F_{\ell\ell}(\mathbf{x}^{(k)})| < \frac{\Lambda}{2}$. Also there exists

$p \geq N_1$ such that $\mathbf{x}^{(p)} \in B_\varepsilon$, hence there is an

$\ell' \in \{1, 2, \dots, n\}$ such that $F_{\ell'}(\mathbf{x}^{(p)}) = 0$ thus $\ell' \in I$.

If $\mathbf{x}^{(q)} \in B_\varepsilon$, $\mathbf{x}^{(q+1)} \in H_J$ for $q \geq p$, then we have

$$\frac{\tau}{2} > |\mathbf{x}^{(q)} - \mathbf{x}^{(q+1)}| \geq |\mathbf{x}^{(q+1)} - \bar{\mathbf{x}}| - |\mathbf{x}^{(q)} - \bar{\mathbf{x}}| > \tau - \varepsilon > \frac{\tau}{2}$$

which is a contradiction. Hence $\mathbf{x}^{(q)} \in B_\varepsilon$ for $q \geq p$.

This shows that for $q \geq p$ there is not any $\ell \in \{1, 2, \dots, n\}$ such that $F_{\ell}(\mathbf{x}^{(p)}) \neq 0$, therefore there is not any ℓ in J thus J is an empty set, which is a contradiction. Thus $\mathbf{x}^{(k)} \rightarrow \mathbf{x}^*$.

Case 2.

$$\phi(\mathbf{x}^{(k)}) - \phi(\mathbf{x}^{(k+1)}) \geq m (x_\ell^{(k)} - x_\ell^{(k+1)})^2 ((1 - \frac{\delta_k}{2}) + \omega).$$

We can state a similar discussion for this case.

3. Numerical Results

In this section we solve some nonlinear systems by SOR-SN method. We compare the CPU time and number of iterations of our method with SOR-Newton(SOR-N), SOR-Steffensen(SOR-St) and SOR-Secant(SOR-Se) methods. Numerical computations have been carried out in MATLAB. The stopping criteria is $\|F(\mathbf{x}^{(k)})\|_\infty < 10^{-13}$,

where $\|\cdot\|_\infty$ shows the Infinity norm.

Note that in SOR-secant method we need to two initial guesses, namely $\mathbf{x}^{(0)}$ and $\mathbf{x}^{(1)}$, while for another methods we only need to one initial guess. In the following parts, we present some examples to compare these methods:

Example 1. Consider the following nonlinear system

$$\begin{aligned} x_1^2 + x_2^2 + x_3^2 &= 9 \\ x_1 x_2 x_3 &= 1 \\ x_1 + x_2 - x_3 &= 0. \end{aligned}$$

The exact solution of the above system is

$$\mathbf{x} = (2.4913756968, .24274587875, 1.6535179393)^t.$$

For an initial guess we set $\mathbf{x}^{(0)} = (2.5, .5, 1.5)^t$ and $\omega = 0.4$ also for SOR-Secant method we set $\mathbf{x}^{(0)} = (.1, .1, .1)^t$, $\mathbf{x}^{(1)} = (2.5, .5, 1.5)^t$. The numerical results are given in Table 1.

Table 1. Results for Example 1

Method	err (10^{-14})	CPU time (s)	Iter.
SOR-SN	4.8738	14.087318	21
SOR-N	7.6309	22.913211	65
SOR-St	6.3949	35.742580	65
SOR-Se	6.7503	26.924918	65

Example 2. We consider the following nonlinear system

$$\begin{aligned} x_1 x_3 + x_4 x_1 + x_4 x_3 &= 0 \\ x_2 x_3 + x_4 x_2 + x_4 x_3 &= 0 \\ x_1 x_2 + x_1 x_3 + x_2 x_3 &= 0 \\ x_1 x_2 + x_4 x_1 + x_4 x_2 &= 0 \end{aligned}$$

Its exact solution is $\mathbf{x} = (0, 0, 0, 0)^t$. For an initial guess we set $\mathbf{x}^{(0)} = (.1, .1, .1, .1)^t$ and $\omega = 0.5$ also for SOR-Secant method we set $\mathbf{x}^{(0)} = (.5, .5, .5, .5)^t$, $\mathbf{x}^{(1)} = (.1, .1, .1, .1)^t$. The numerical results are given in Table 2.

Table 2. Results for Example 2

Method	err (10^{-14})	CPU time (s)	Iter.
SOR-SN	4.2633	6.978867	10
SOR-N	5.6567	15.557740	35
SOR-St	6.4758	24.313022	34
SOR-Se	6.4758	18.594764	34

Example 3. Consider the following system

$$F_i(\mathbf{x}) = x_i^2 - 1, \quad i = 1, \dots, m.$$

Its exact solutions are $\mathbf{x} = (1, 1, \dots, 1)^t$ and $(-1, -1, \dots, -1)^t$ for different values of m . For an initial guess we set $\mathbf{x}^{(0)} = (1.5, 1.5, \dots, 1.5)^t$ and $\omega = 0.8$ also for SOR-Secant method we set $\mathbf{x}^{(0)} = (1, 1, \dots, 1)^t$, $\mathbf{x}^{(1)} = (1.5, \dots, 1.5)^t$. The numerical results are given in Table 3.

Table 3. Results for Example 3.

m	Method	err(10^{-14})	CPU time(s)	Iter.
4	SOR-SN	0	5.642484	6
	SOR-N	6.5725	10.627721	19
	SOR-St	2.2871	14.902830	20
	SOR-Se	7.0610	11.035609	19
5	SOR-SN	0	7.023182	6
	SOR-N	6.5725	13.328038	19
	SOR-St	2.2871	19.442190	20
	SOR-Se	7.061	14.01274	19
10	SOR-SN	0	14.554587	6
	SOR-N	6.5725	29.039288	19
	SOR-St	2.2871	41.446225	20
	SOR-Se	7.061	31.160055	19
20	SOR-SN	0	34.889920	6
	SOR-N	6.5725	77.345117	19
	SOR-St	2.2871	103.3088041	20
	SOR-Se	7.061	81.781951	19
40	SOR-SN	0	95.250077	6
	SOR-N	6.5725	250.734556	19
	SOR-St	2.2871	309.225352	20
	SOR-Se	7.061	255.216354	19
80	SOR-SN	0	323.583431	6
	SOR-N	6.5725	1058.228354	19
	SOR-St	2.2871	1212.605952	20
	SOR-Se	7.061	1046.439768	19

Example 4. Consider the following nonlinear system

$$F_i(\mathbf{x}) = \sin(x_i), \quad i = 1, \dots, m.$$

One of the exact solutions of the system is $\mathbf{x} = (0, 0, \dots, 0)^t$ for different values of m . For an initial guess we set $\mathbf{x}^{(0)} = (.1, \dots, .1)^t \times \frac{\pi}{180}$ also $\omega = 0.8$ and for SOR-Secant method we set

$$\mathbf{x}^{(0)} = (.5, \dots, .5)^t \times \frac{\pi}{180}, \quad \mathbf{x}^{(1)} = (.1, \dots, .1)^t \times \frac{\pi}{180}.$$

The numerical results are given in Table 4.

Table 4. Results for Example 4.

m	Method	err(10^{-14})	CPU time(s)	Iter.
4	SOR-SN	0	1.998834	2
	SOR-N	5.7191	8.319543	15
	SOR-St	5.7191	10.843830	15
	SOR-Se	5.7191	8.564604	15
5	SOR-SN	0	2.461118	2
	SOR-N	5.7191	10.177663	15
	SOR-St	5.7191	13.179991	15
	SOR-Se	5.7191	10.656945	15
10	SOR-SN	0	4.989769	2
	SOR-N	5.7191	22.646616	15
	SOR-St	5.7191	29.142497	15
	SOR-Se	5.7191	23.198631	15
20	SOR-SN	0	11.242210	2
	SOR-N	5.7191	60.061224	15
	SOR-St	5.7191	73.690408	15
	SOR-Se	5.7191e	61.808370	15
40	SOR-SN	0	27.044883	2
	SOR-N	5.7191	189.645564	15
	SOR-St	5.7191	218.421042	15
	SOR-Se	5.7191	193.010061	15
80	SOR-SN	0	97.220895	2
	SOR-N	5.7191	820.077416	15
	SOR-St	5.7191	878.713914	15
	SOR-Se	5.7191	821.963020	15

In the following example, we apply the new method to

solve a boundary value problem. The stopping criteria used in this case is $\|F(y_k)\|_\infty < 10^{-13}$, where y_k is the k th approximation of the solution.

Example 5 . We consider the following boundary value problem, which is given in[3]:

$$y'' = -(y')^2 - y + \ln x, \quad x \in [0, 2],$$

$$y(1) = 0, \quad y(2) = \ln 2.$$

The exact solution of this problem is $y(x) = \ln x$. To discretize the problem we use the second order finite difference method. The zeros of the following nonlinear functions will provide us an estimation of the solution of the problem:

$$F(y) = (f_1(y), f_2(y), \dots, f_{n-1}(y))^t,$$

where

$y = (y_1, y_2, \dots, y_{n-1})^t$ and for $i = 1, 2, \dots, n-1$ we have $f_i : \mathbb{R}^{n-1} \rightarrow \mathbb{R}$, such that

$$f_1 = 4 y_2 + y_2^2 + 4 y_1 (h^2 - 2) - 4 h^2 \ln x_2,$$

$$f_i = 4 (y_{i+1} + y_{i-1}) + (y_{i+1} - y_{i-1})^2 + 4 y_i (h^2 - 2) - 4 h^2 \ln x_{i+1},$$

for $i = 2, 3, \dots, n-2$,

$$f_{n-1} = 4 (\ln 2 + y_{n-2}) + (\ln 2 - y_{n-2})^2$$

$$+ 4 y_{n-1} (h^2 - 2) - 4 h^2 \ln x_n.$$

In the above system, the second order approximations are used for $y'(x_i)$ and $y''(x_i)$ by step $h = \frac{1}{n}$. By this step, we set the nodes x_i as:

$$x_i = 1 + ih, i = 0, 1, \dots, n.$$

Also, $y_i, i = 1, \dots, n-1$ denotes the unknown $y(x_i)$. In Table 5 some results can be observed. For any initial estimation, we analyze the number of iterations and CPU time needed to converge to the solution. The initial estimations used are:

$$p = (p_1, \dots, p_{n-1}), p_i = 0 + i \frac{\ln 2}{0.4n}, \quad i = 1, 2, \dots, n-1$$

$$q = (q_1, \dots, q_{n-1}), q_i = 1.2, \quad i = 1, 2, \dots, n-1$$

For SOR-secant in Table 5 we set $x^{(0)} = p, x^{(1)} = q$ and in

Table 6 we set $x^{(0)} = q, x^{(1)} = p$. And $\omega = 0.7$ in both cases. As we can see from Tables 5 and 6 for all cases the number of iterations and CPU time for our new method are less than number of iterations and CPU time for the other methods.

Table 5. Results for $p_i = i \frac{\ln 2}{.4n}$ in Example 5

m	Method	err(10^{-14})	CPUtime(s)	Iter.
4	SOR-SN	8.9965	33.073550	50
	SOR-N	8.1213	36.638949	102
	SOR-St	8.1213	60.880844	102
	SOR-Se	8.1213	45.860077	102
5	SOR-SN	8.1012	70.473741	81
	SOR-N	9.1226	75.927102	157
	SOR-St	9.1448	123.773890	157
	SOR-Se	9.1226	94.721393	157
6	SOR-SN	9.8865	127.758798	117
	SOR-N	9.4307	135.968814	224
	SOR-St	9.4251	222.942813	224
	SOR-Se	9.4043	168.422923	224

Table 6. Results for $q_i = 1.2$ in Example 5.

m	Method	err(10^{-14})	CPU time(s)	Iter.
4	SOR-SN	9.6090	32.068312	50
	SOR-N	8.1657	37.054225	103
	SOR-St	8.1657	61.235463	103
	SOR-Se	8.0991	46.156399	103
5	SOR-SN	8.8117	70.732677	81
	SOR-N	8.5008	77.345475	159
	SOR-St	8.5008	126.204815	159
	SOR-Se	8.4786	96.409719	159
6	SOR-SN	8.5543	126.523083	118
	SOR-N	8.8124	137.612224	227
	SOR-St	8.7930	225.340664	227
	SOR-Se	8.7930	169.037417	227

4. Conclusions

In this paper, we have presented SOR-SN algorithm to solve systems of nonlinear equations. We have shown that our method is convergent. In comparison with another SOR type methods, such as, SOR-Newton, SOR-Steffensen and SOR-Secant methods, SOR-SN algorithm were better in CPU time and number of iterations.

REFERENCES

- [1] J.M. Ortega, W.C. Rheinboldt. Iterative solution of nonlinear equations in several variables, Academic Press, 1970
- [2] D.K.R. Babajee, M.Z. Dauhoo. (2006) An analysis of the properties of the variants of Newton's method with third order convergence, Appl. Math. Comput., 183, 659-684

- [3] A. Cordero, J.R. Torregrosa. (2006) Variants of Newton's method for functions of several variables, *Appl. Math. Comput.*, 183, 199-208
- [4] J.A. Ezquerro, M.A. Hernandez. (2003) A uniparametric Halley-type iteration with free second derivative, *Int. J. Pure Appl. Math.*, 6, 103-114
- [5] J.A. Ezquerro, M.A. Hernandez. (2004) On Halley-type iterations with free second derivative, *J. Comp. Appl. Math.*, 170, 455-459
- [6] M. Grau-Sanchez. (2007) Improvements of the efficiency of some three-step iterative Newton-like methods, *Numer. Math.*, 107, 131-146
- [7] M. A. Hernandez. (2000) Second-derivative-free variant of the Chebyshev method for nonlinear equations, *J. Opt. Theo. Appl.*, 104, 501-515
- [8] H.H.H. Homeier. (2004) Modified Newton method with cubic convergence: the multivariate case, *J. Comput. Appl. Math.*, 169, 161-169
- [9] A. Cordero, J.L. Hueso, E. Martinez, J.R. Torregrosa. (2010) Accelerated methods of order $2p$ for systems of nonlinear equations, *J. Comput. Appl. Math.*, 233, 2696-2702
- [10] A. Cordero, J.L. Hueso, E. Martinez, J.R. Torregrosa. (2011) Efficient high-order methods based on golden ratio for nonlinear systems, *Appl. Math. Comput.*, 217, 4548-4556
- [11] M.T. Darvishi, A. Barati. (2007) A third-order Newton-type method to solve systems of nonlinear equations, *Appl. Math. Comput.*, 187, 630-635
- [12] M.T. Darvishi, A. Barati. (2007) A fourth-order method from quadrature formulae to solve systems of nonlinear equations, *Appl. Math. Comput.*, 188, 257-261
- [13] M.T. Darvishi, A. Barati. (2007) Super cubic iterative methods to solve systems of nonlinear equations, *Appl. Math. Comput.*, 188, 1678-1685
- [14] M.T. Darvishi. (2009) A two-step high order Newton-like method to solve systems of nonlinear equations, *Int. J. Pure and Appl. Math.*, 57(4), 485-498
- [15] M.T. Darvishi. (2009) Some three-step iterative methods free from second order derivative for finding solutions of systems of nonlinear equations, *Int. J. Pure and Appl. Math.*, 57(4), 499-516
- [16] M. Frontini, E. Sormani. (2004) Third-order methods from quadrature formulae for solving systems of nonlinear equations, *Appl. Math. Comput.*, 149, 771-782
- [17] D.K.R. Babajee, M.Z. Dauhoo, M.T. Darvishi, A. Barati. (2008) A note on the local convergence of iterative methods based on Adomian decomposition method and 3-node quadrature rule, *Appl. Math. Comput.*, 200, 452-458
- [18] H.G. Ren, Q.B. Wu, W.H. Bi. (2009) A class of two-step Steffensen type methods with fourth-order Convergence, *Appl. Math. Comput.*, 209, 206-210
- [19] M.E. Brewster, R. Kannan. (1984) Nonlinear successive over-relaxation, *Numer. Math.*, 44, 309-315