RESEARCHERID
THOMSON REUTERS

[Iliev* 4(10): October, 2017] ISSN 2349-4506
Impact Factor: 2.785

Global Journal of Engineering Science and Research Management

SOME NOTES ON THE FAST ADAPTIVE NEURAL NETWORK SOLVERS

Anna Malinova, Angel Golev, Anton Iliev*, Nikolay Kyurkchiev

* Faculty of Mathematics and Informatics, University of Plovdiv Paisii Hilendarski, 24, Tzar Asen Str., 4000 Plovdiv, Bulgaria

Faculty of Mathematics and Informatics, University of Plovdiv Paisii Hilendarski, 24, Tzar Asen Str., 4000 Plovdiv, Bulgaria

Faculty of Mathematics and Informatics, University of Plovdiv Paisii Hilendarski, 24, Tzar Asen Str., 4000 Plovdiv, Bulgaria

Faculty of Mathematics and Informatics, University of Plovdiv Paisii Hilendarski, 24, Tzar Asen Str., 4000 Plovdiv, Bulgaria

DOI: 10.5281/zenodo.1034497

KEYWORDS: solving polynomial systems of equations, Weierstrass method, non-attractive set of initial approximations.

ABSTRACT

With this paper, we discuss some important aspects related to the iterative solution of two classes of polynomials, nonlinear systems of equations, and the adapted to them – "FAST adaptive neural solver" (FANS). The crucial issue of choosing initial approximations (separation of unattractive networks of initial data) and the possibility of minimizing CPU–time with the use of existing FANS is discussed.

INTRODUCTION

Polynomial systems of equations are of major interest and they are heavily used in any discipline of sciences such as mathematics, physics, chemistry and engineering.

According to [1], [2], the approaches for solving polynomial systems of equations can be classified in two categories as follows:

"1. Symbolic methods that stem from algebraic geometry;

2. Numerical methods, based on iterative procedures. These methods are suitable for local analysis only and perform well only if the initial guess is good enough, a condition that generally is rather difficult to satisfy."

In [1] the authors considered the neural network architecture for the system of polynomials

$$f_i(x) = \sum_{j=1}^{k_i} \left(a_{ij} \prod_{l=1}^{n} x_l^{e_{lj}^i} \right) - \beta_i = 0;\ i = 1,2,\dots,n, \quad (1)$$

where in every exponent e^i the superscript i denotes the equation, the first subscript j denotes the factor of the summation in equation i and the second subscript l denotes the corresponding unknown x.

Introduced in [1] neural solvers gave good results for polynomial systems associated with chemical engineering applications.

For other results, see [4]–[9].

The point $\tilde{x} \in R^n$ is an equilibrium point for the differential equation

$$\frac{dx}{dt} = g(t,x)$$

if $g(t,\tilde{x}) = 0$ for all t.

Of course, as we have already mentioned, it remains of crucial importance to choose initial approximations for which researched numerical method or iterative procedure is convergent.

The article is structured as follows. Firstly in **"MAIN RESULTS"** we explore one special class of polynomial systems and give an analytical description of the non-attractive sets to be considered by the specialists working in the direction - generating of FANS.

Similar issues are discussed in "**Solving the more general polynomial systems class of the type (1)"** - to solve an arbitrary nonlinear system based on the use of fast-acting iterative procedures with order of convergence t

(appearing as modification of classical Newton-Broyden method).

MAIN RESULTS

We will illustrate the said for a special class of polynomial systems.

Many problems in mathematics and other natural sciences and techniques reduce themselves to determining all roots of a system of equations:

$$\begin{vmatrix} f_1(x) &= x^{d_1} + a_{1,1}x^{d_1-1} + \cdots + a_{1,d_1} = 0, \\ f_2(x) &= x^{d_2} + a_{2,1}x^{d_2-1} + \cdots + a_{2,d_2} = 0, \\ \cdots & \\ f_n(x) &= x^{d_n} + a_{n,1}x^{d_n-1} + \cdots + a_{n,d_n} = 0, \end{vmatrix} \quad (2)$$

where $f_j(x)$ are polynomials of degree d_j, $1 \le j \le n$ with simple zeros. Let $x_{i,j}^k, i = 1,2,\dots,n;\ j = 1,2,\dots,d_i$, be distinct reasonably close approximations of these zeros. Usually, the Weierstrass procedure is used to solve the problem [45]:

$$x_{i,j}^{k+1} = x_{i,j}^k - \frac{f_i(x_{i,j}^k)}{\prod_{l\neq s}^{d_i}(x_{i,l}^k - x_{i,s}^k)};\ j = 1,2,\dots,d_i;\ \ i = 1,2,\dots,n.$$

Finding the zeros of the polynomial system (2) is related closely to research in the area of "chemical equilibrium applications", "kinematic applications" and others. Following the ideas given in [34], [30] we obtain

$$\begin{aligned} &\textstyle\sum_{p=1}^{d_r} x_{r,p}^{k+1} = -a_{r,1}, \\ &\textstyle\sum_{p=1}^{d_r} x_{r,p}^{k+1} \sum_{q\neq p}^{d_r} x_{r,q}^k = \sum_{l<s}^{d_r} x_{r,l}^k x_{r,s}^k + a_{r,2}, \\ &\textstyle\sum_{p=1}^{d_r} x_{r,p}^{k+1} \sum_{l<s;l,s\neq p} x_{r,l}^k x_{r,s}^k = 2\sum_{l<s<t}^{d_r} x_{r,l}^k x_{r,s}^k x_{r,t}^k - a_{r,3}, \\ &\dots \\ &\textstyle\sum_{p=1}^{d_r} x_{r,p}^{k+1} \prod_{q\neq p}^{d_r} x_{r,q}^k = (d_r - 1)\prod_{q=1}^{d_r} x_{r,q}^k + (-1)^{d_r} a_{r,d_r},\ \ r = 1,2,\dots,n. \end{aligned} \quad (3)$$

The resulting systems of equations (3) can be written in vector form as:

$$A_{d_r}^r x_{rd_r}^{k+1} = b_{d_r}^r;\ \ r = 1,2,\dots,n,$$

where

$$A_{d_r}^r := \begin{pmatrix} 1 & 1 & \dots & 1 \\ \sum_{q\neq 1} x_{r,q}^k & \sum_{q\neq 2} x_{r,q}^k & \dots & \sum_{q\neq d_r} x_{r,q}^k \\ \vdots & \vdots & & \vdots \\ \prod_{q\neq 1} x_{r,q}^k & \prod_{q\neq 2} x_{r,q}^k & \dots & \prod_{q\neq d_r} x_{r,q}^k \end{pmatrix},$$

$$detA_{d_r}^r = \prod_{i<j}^{d_r} (x_{r,i}^k - x_{r,j}^k) \neq 0,$$

$$x_{rd_r}^{k+1} := \begin{pmatrix} x_{r,1}^{k+1} \\ x_{r,2}^{k+1} \\ \vdots \\ x_{r,d_r}^{k+1} \end{pmatrix},$$

$$b_{d_r}^r := \begin{pmatrix} b_{r,1}^{d_r} \\ b_{r,2}^{d_r} \\ \vdots \\ b_{r,d_r}^{d_r} \end{pmatrix} = \begin{pmatrix} -a_{r,1} \\ \sum_{l<s}^{d_r} x_{r,l}^k x_{r,s}^k + a_{r,2} \\ \vdots \\ (d_r - 1)\prod_{q=1}^{d_r} x_{r,q}^k + (-1)^{d_r} a_{r,d_r} \end{pmatrix},\ \ r = 1,2,\dots,n.$$

We shall use also the notations

$$S_m^{r,d_r} := \textstyle\sum_{1\le i_1<i_2<\cdots<i_m\le d_r} x_{r,i_1}^k \dots x_{r,i_m}^k,\ \ r = 1,2,\dots,n;\ \ m = 1,2,\dots,d_r.$$

Then

$$b_{r,m}^{d_r} = (m-1)S_m^{r,d_r} + (-1)^m a_{r,m},\ r = 1,2,\dots,n;\ \ m = 1,2,\dots,d_r.$$

For any given $1 \le i < j \le d_r, r = 1,2,\dots,n$, we define the polynomials:

$$\omega_{ij}^{r,d_r}(x) = (x - x_{r,1}^k)\dots(x - x_{r,i-1}^k)(x - x_{r,i+1}^k)\dots(x - x_{r,j-1}^k)(x - x_{r,j+1}^k)\dots(x - x_{r,d_r}^k).$$

We have the following theorem.

Theorem A. [37] Suppose that for some $1 \le i < j \le d_r,\ r = 1,2,\dots n$, the sequence of approximations $x_{r,1}^k, \dots, x_{r,d_r}^k$, $r = 1,2,\dots,n$, satisfies the conditions

$$\sum_{m=1}^{d_r} (-1)^m b_{r,m}^{d_r} \left[\omega_{ij}^{r,d_r}(x_{r,j}^k)\left(x_{r,i}^k\right)^{d_r-m} + \omega_{ij}^{r,d_r}(x_{r,i}^k)\left(x_{r,j}^k\right)^{d_r-m}\right] = 0, \quad (4)$$

$$r = 1,2,\dots,n.$$

Then $x_{r,i}^{k+1} = x_{r,j}^{k+1}, r = 1,2,\dots,n$, and thus, the ($k$+2)-th step of the Weierstrass method cannot be performed.

The set $D_{f[f_1,\dots,f_n]}$ of the non-attractive starting points is the set of points satisfying equations (4).

Example 1. We consider the system (Mamat et al. [3], Goulianis et al. [1]):

$$f_i(x) = x_i^2 + x_i - 2 = 0, i = 1,2,\dots,n. \quad (5)$$

Let $n = 2$. The non-attractive set $D_{f[f_1,f_2]}$ is given by (see, (4))

$$D_{f[f_1,f_2]} = \cup_{i=1}^2 D_{f_i},$$

where

$$D_{f_1}: 2x_{1,1}^k x_{1,2}^k + x_{1,1}^k + x_{1,2}^k - 4 = 0,$$
$$D_{f_2}: 2x_{2,1}^k x_{2,2}^k + x_{2,1}^k + x_{2,2}^k - 4 = 0.$$

The divergent set of the system (5) consists of the union of the hyperbolas of type (see Fig.1)

$$y = \frac{4-x}{2x+1}.$$

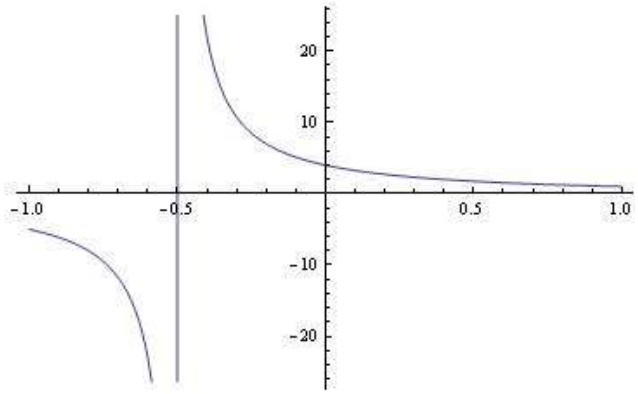

Figure 1: The hyperbola $y = \frac{4-x}{2x+1}$.

Example 2. We consider the system

$$\begin{cases} f_1(x) = x^2 - 1 = 0, \\ f_2(x) = x^2 - x - 2 = 0, \\ f_3(x) = x^3 + 1 = 0 \end{cases} \quad (6)$$

which has root -1. The non-attractive set $D_{f[f_1,f_2,f_3]}$ is given by (see, (4))

$$D_{f[f_1,f_2,f_3]} = \cup_{i=1}^{3} D_{f_i},$$

where

$$D_{f_1}: x_{1,1}^k x_{1,2}^k - 1 = 0,$$
$$D_{f_2}: x_{2,2}^k\left(2x_{2,1}^k - 1\right) = 4 + x_{2,1}^k,$$
$$D_{f_3}: (x_{3,1}^k - x_{3,2}^k)^2 (x_{3,3}^k)^2 + (2 + x_{3,1}^k (x_{3,2}^k)^2 + (x_{3,1}^k)^2 x_{3,2}^k) x_{3,3}^k$$
$$-(2(x_{3,1}^k)^2 (x_{3,2}^k)^2 + x_{3,1}^k + x_{3,2}^k) = 0.$$

The divergent set of the system (6) consists of the union of the hyperbolas (see Fig.2–Fig.3)

$$x_{1,2}^k = \frac{1}{x_{1,1}^k}, \quad x_{2,2}^k = \frac{4 + x_{2,1}^k}{2x_{2,1}^k - 1}$$

and surface (see Fig. 4).

$$z = \frac{-(2+xy^2+x^2y) + \varepsilon\sqrt{(2+xy^2+x^2y)^2 + 4(x-y)^2(2x^2y^2+x+y)}}{2(x-y)^2}, \quad \varepsilon = \pm 1,$$

$$(z = x_{3,3}^k; \ x = x_{3,1}^k; \ y = x_{3,2}^k).$$

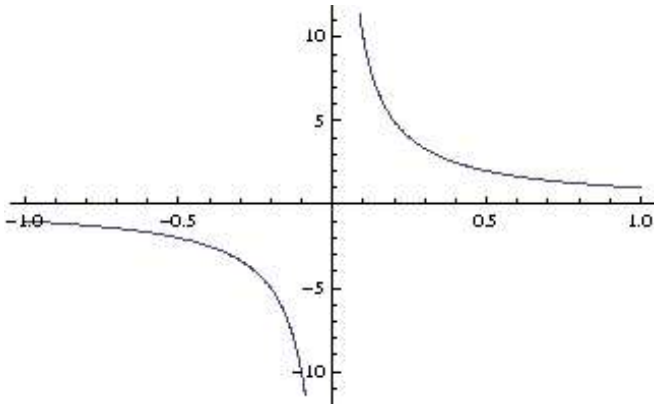

Figure 2: The hyperbola $x_{1,2}^k = \frac{1}{x_{1,1}^k}$.

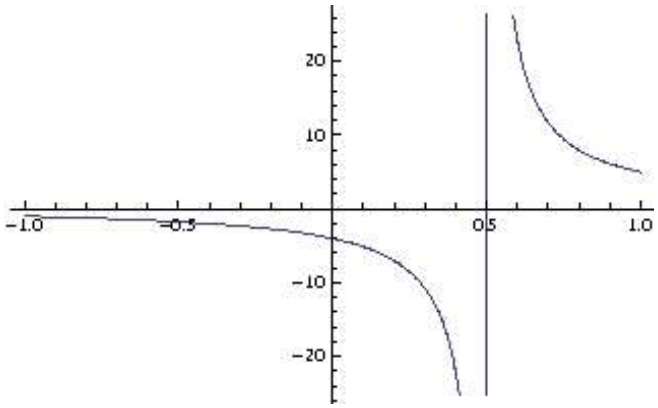

Figure 3: The hyperbola $x_{2,2}^k = \frac{4 + x_{2,1}^k}{2x_{2,1}^k - 1}$.

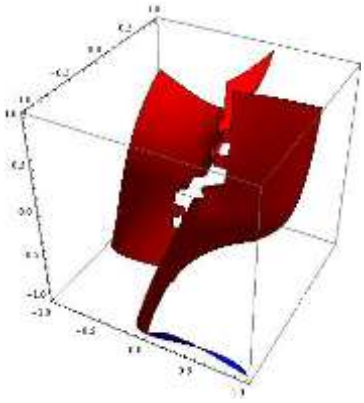

Figure 4: The surface z.

Remarks

In the Example 1, the Newton's procedure does not converge for $n > 2$ and Weierstrass procedure fails for the non-attractive set $D_{f[f_1,f_2]}$ of initial approximations.

The Example 2 is very instructive and divergent set of initial approximations $D_{f[f_1,f_2,f_3]}$ is very complicated.

In studying the equilibrium state of some classes of differential systems that require the study of polynomial systems of the type (2) it is extremely important to "separate" the non-attractive initial approximations to the roots.

For an arbitrary n the description of conditions (4) it is not difficult due to the existing symmetry.

You will explicitly note that for all known classic and newer algorithms like Chebishev, Halley, Ehrlich, Abert, Nourein, Dvorchuk, Petkovic, Kyurkchiev, Iliev, Proinov and others(see, [29], [31]–[33], [35]–[36], [38]–[43]) to solve the assigned task are derived and described in details respective non-attractive sets of initial data that are permanently present in the implemented scientific platform (intellectual property, see, for instance [44]).

This is a very important element in dealing with issues of the mentioned issues.

First of all, the user of such algorithms usually is naturalist and a priori is not supposed to be a specialist in "Applied Mathematics".

In this sense, the requirements to user of choosing initial approximations by using different subroutines and products included in program environments and platforms are highly restrictive.

Program modules in contemporary environments must be so set to automatically report ”non-attractive sets” (as we have noted - it is not difficult their description) and to provide the user reliable solution to the problem.

Goulianis et al. [1] showed that the proposed adaptive neural network algorithm is characterized by fast convergence for high–dimensional systems.

Of course analogous questions regarding the appropriate choice of initial approximations can be found in the existing and described in the literature adaptive neural network solvers.

Inevitably that will mean – ”better CPU–time”.

Solving the more general polynomial systems class of the type (1)

General iteration process, which possesses order of convergence t is constructed in [32], [33]. From this process, Newton's iteration formula ($t = 2$) and Halley's iteration formula ($t = 3$) are received as particular cases. The used technique is based on generalized Taylor's formula.

Let the system of equations

$$f_i(\vec{x}_k) = 0, i = 1,2,\dots,n \tag{7}$$

be given.

Supposed that f_i and the partial derivatives of these functions of sufficiently high order are continuous in the neighborhood of solution $\vec{\xi}(\xi_1, \xi_2, \dots, \xi_n)$.

Both Newton's and Halley's iteration formula can be generalized when in Taylor's formula all the terms up to

degree $t-1$ with respect of h^s will be included. Taylor's formula is of the form

$$0 = f_i + f_{ii_1}^{(1)} h^{i_1} + \frac{1}{2!} f_{ii_1 i_2}^{(1)} h^{i_1} h^{i_2} + \frac{1}{3!} f_{ii_1 i_2 i_3}^{(1)} h^{i_1} h^{i_2} h^{i_3} + \ldots + \frac{1}{(t-1)!} f_{ii_1 i_2 \ldots i_{t-1}}^{(1)} h^{i_1} h^{i_2} \ldots h^{i_{t-1}} + \bar{O}(\varepsilon^t). \tag{8}$$

Following the scheme by which Newton's and Halley's iteration formulae were developed we suppose that in the above consideration we have determined

$$h^j = H_{t-2}^j + O^j(\varepsilon^{t-1}) \tag{9}$$

and

$$\|O_i(\varepsilon^{t-1})\| \le M_{t-1}^* \varepsilon^{t-1}. \tag{10}$$

Then, after replacing h^j from (10) in (9) we receive

$$\begin{aligned} 0 = f_i &+ \Big(f_{ii_{(1)}}^1 + \frac{1}{2!} f_{ii_1 i_2}^{(1)} H_{t-1}^{i_2} + \frac{1}{3!} f_{ii_1 i_2 i_3}^{(1)} H_{t-1}^{i_2} H_{t-1}^{i_3} \\ &+ \ldots + \frac{1}{(t-1)!} f_{ii_1 i_2 \ldots i_{t-1}}^{(1)} H_{t-1}^{i_2} H_{t-1}^{i_3} \ldots H_{t-1}^{i_{t-1}} \Big) h^{i_1} + \frac{1}{2!} f_{ii_1 i_2}^{(1)} O^{i_2}(\varepsilon^{t-1}) h^{i_1} \\ &+ \frac{1}{3!} f_{ii_1 i_2 i_3}^{(1)} \Big(O^{i_2}(\varepsilon^{t-1}) H_{t-1}^{i_3} + O^{i_3}(\varepsilon^{t-1}) H_{t-1}^{i_2} + O^{i_2}(\varepsilon^{t-1}) O^{i_3}(\varepsilon^{t-1}) \Big) h^{i_1} \\ &+ \ldots + \frac{1}{(t-1)!} f_{ii_1 i_2 \ldots i_{t-1}}^{(1)} \big(O^{i_2}(\varepsilon^{t-1}) H_{t-1}^{i_3} H_{t-1}^{i_4} \ldots H_{t-1}^{i_{t-1}} \\ &+ \ldots + H_{t-1}^{i_2}(\varepsilon^{t-1}) H_{t-1}^{i_3}(\varepsilon^{t-1}) \ldots O^{i_{t-1}}(\varepsilon^{t-1}) \\ &+ O^{i_2}(\varepsilon^{t-1}) O^{i_3}(\varepsilon^{t-1}) \ldots \big) h^{i_1} + \bar{O}_i(\varepsilon^t). \end{aligned} \tag{11}$$

The system (11) we write in the form

$$0 = f_i + f_{ii_1}^{(t)} h^{i_1} + O_i(\varepsilon^t), \tag{12}$$

where we substitute

$$f_{ii_1}^{(t)} = f_{ii_1}^1 + \frac{1}{2!} f_{ii_1 i_2}^{(1)} H_{t-1}^{i_2} + \ldots + \frac{1}{(t-1)!} f_{ii_1 i_2 \ldots i_{t-1}}^1 H_{t-1}^{i_2} H_{t-1}^{i_3} \ldots H_{t-1}^{i_{t-1}}$$

and

$$\begin{aligned} O_i(\varepsilon^t) = &\frac{1}{2!} f_{ii_1 i_2}^{(1)} O^{i_2}(\varepsilon^{t-1}) h^{i_1} + \frac{1}{3!} f_{ii_1 i_2 i_3}^{(1)} \big(O^{i_2}(\varepsilon^{t-1}) H_{t-1}^{i_3} + O^{i_3}(\varepsilon^{t-1}) H_{t-1}^{i_2} \\ &+ O^{i_2}(\varepsilon^{t-1}) O^{i_3}(\varepsilon^{t-1}) \big) h^{i_1} + \ldots + \frac{1}{(t-1)!} \big(O^{i_2}(\varepsilon^{t-1}) H_{t-1}^{i_3} H_{t-1}^{i_4} \ldots H_{t-1}^{i_{t-1}} \\ &+ \ldots + H_{t-1}^{i_2}(\varepsilon^{t-1}) H_{t-1}^{i_3}(\varepsilon^{t-1}) \ldots O^{i_{t-1}}(\varepsilon^{t-1}) + \ldots \big) h^{i_1} + \bar{O}_i(\varepsilon^t). \end{aligned}$$

The elements of reciprocal matrix of $\left\{ f_{ii_1}^{(t)} \right\}$ we denote with $f_{(t)}^{ii_1}$. From the system (12) we determine h^{i_1}

$$h^{i_1} = -f_{(t)}^{i_1 s} f_s + f_{(t)}^{i_1 s} O_s(\varepsilon^t). \tag{13}$$

We substitute $H_{t-1}^{i_1} = -f_{(t)}^{i_1 s} f_s$.

The following iteration formula can be formed:

$$x_{k+1}^s = x_k^s + H_{t-1}^s, \tag{14}$$

where t denotes order of convergence.

Let us form the expression

$$\xi^s - x_{(k)}^s - H_t^s = \xi^s - x_{(k+1)}^s = h^s - H_t^s = O^s(\varepsilon^t).$$

We have

$$\|O_i(\varepsilon^t)\| \le \frac{1}{2!} \left\| f_{ii_1 i_2}^{(1)} \right\| \|O^{i_2}(\varepsilon^{t-1})\| \|h^{i_1}\| + \frac{1}{3!} \Big(\left\| f_{ii_1 i_2 i_3}^{(1)} \right\| \|O^{i_2}(\varepsilon^{t-1})\| \|h^{i_1}\| + \left\| f_{ii_1 i_2 i_3}^{(1)} \right\| \|O^{i_3}(\varepsilon^{t-1})\| \|h^{i_1}\| + \ldots \Big)$$

$$+\ldots+\frac{1}{(t-1)!}\left(\left\|f^{(1)}_{ii_1i_2\ldots i_{t-1}}\right\|\|O^{i_2}(\varepsilon^{t-1})\|\|H^{i_3}_{t-1}\|\ldots\|H^{i_{t-1}}_{t-1}\|\right.$$

$$\left.+\ldots+\left\|f^{(1)}_{ii_1i_2\ldots i_{t-1}}\right\|\|H^{i_2}_{t-1}\|\|H^{i_3}_{t-1}\|\ldots\|O^{i_{t-1}}(\varepsilon^{t-1})\|+\ldots\right)+\|\bar{O}(\varepsilon^t)\|$$

and

$$\|\bar{O}_i(\varepsilon^t)\|=\left\|f^{(1)}_{ii_1i_2\ldots i_t}h^{i_1}h^{i_2}\ldots h^{i_t}\right\|\le\left\|f^{(1)}_{ii_1i_2\ldots i_t}\right\|\|h^{i_1}\|\|h^{i_2}\|\ldots\|h^{i_t}\|\le\left\|f^{(1)}_{ii_1i_2\ldots i_t}\right\|\varepsilon^t,$$

where all summands which contained multipliers ε^s are dropped, when $s>t$.
Taking into account the fact that functions f_i are sufficiently smooth and that in above estimates all summands are of order $O_i(\varepsilon^t)$.

We conclude that the following upper bound is valid

$$\|O_i(\varepsilon^t)\|\le M^*_t\varepsilon^t, \tag{15}$$

where M^*_t is a positive constant.

Using (15) we receive the inequalities

$$\left\|\xi^s-x^s_{(k+1)}\right\|\le\|O^s(\varepsilon^t)\|\le M^*_t\varepsilon^t\left\|f^{i_1l}_t\right\|.$$

Example 3. We consider the system

$$\begin{cases}f_1(x)=0.25x_1^2+x_2^2-1=0,\\ f_2(x)=x_1^2-2x_1+x_2^2=0,\end{cases} \tag{16}$$

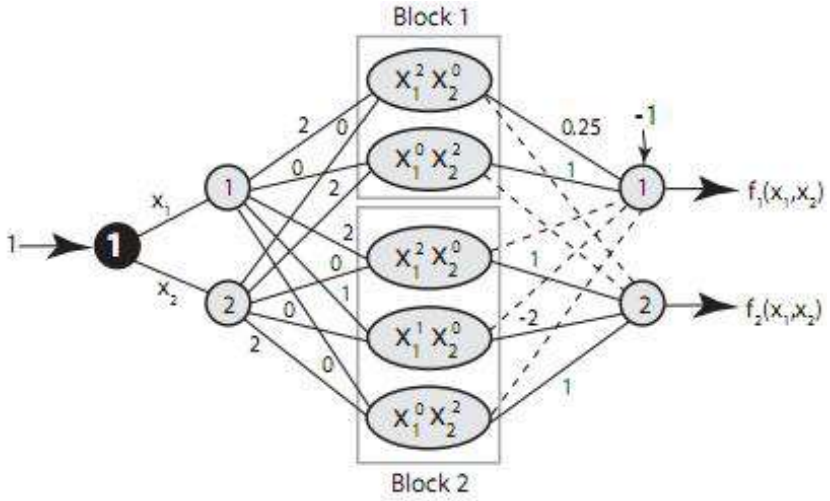

Figure 5: The structure of neural solver for Example 3. [1].

When we apply formula (14) to solve system from Example 3 for $t=3$ with initial approximations $(1;-1)$, $(1;1)$ and $(3;1)$ with accuracy 15 decimal digits the respectively results are

$$(0.666666666666667;-0.942809041582063),$$

$$(0.666666666666667;0.942809041582063)$$

and

$$(2.000000000000000;0.000000000000001)$$

for 4 iterations only.

For specifying the double root (2,0), appropriate modification of the method outlined above is used.

Example 4. We consider the system

$$\begin{cases} f_1(x) = x_1x_2^2 - 3x_1^3x_2^2 - 5 = 0, \\ f_2(x) = 2x_1^3x_2^2 + 2x_1^4x_2^4 - 4x_1^2x_2^4 + 3 = 0, \end{cases} \tag{17}$$

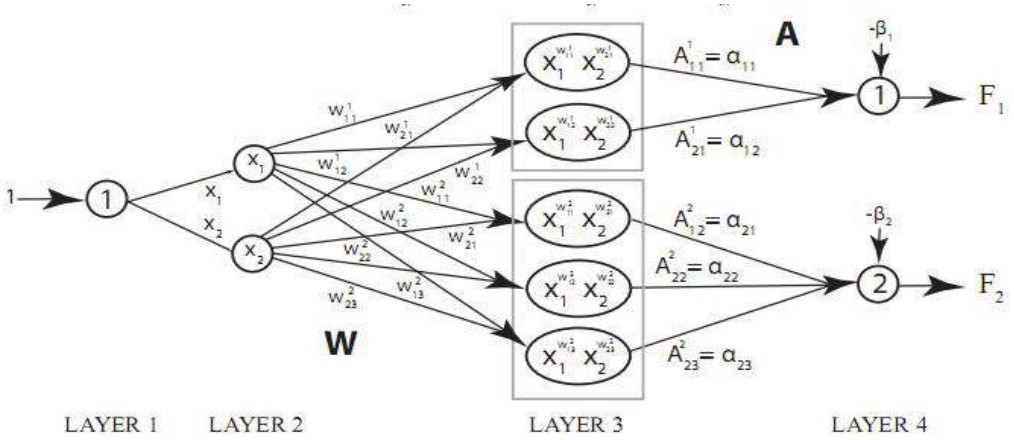

Figure 6: The structure of neural solver for Example 4. [1].

We have used the formula (14) for finding solutions of system from Example 4 for $t = 3$ with initial approximations $(-1.9; 0.7)$, $(-1.9; -0.7)$, $(-3.6; 0.2)$ and $(-3.6; -0.2)$ with accuracy 15 decimal digits and the respectively results are

$$(-1.70807408954102; 0.614482226647498),$$

$$(-1.70807408954102; -0.614482226647499),$$

$$(-3.32903633273033; 0.215813224817305)$$

and

$$(-3.32903633273034; -0.215813224817305)$$

for 4 iterations only.

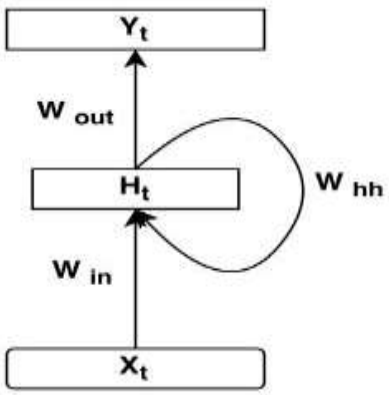

Figure 7: Higher order recurrent neural networks.

CONCLUDING REMARKS

Part of the notes in **"Remarks"** remain valid for solving arbitrary systems of nonlinear equations.
Here is also interesting the question of isolating the non-attractive initial approximations using different modifications of the classic Newton–Broyden type methods with high order of convergence t.
The much broader issue of selecting initial data for solving polynomial systems of type (1) can be considered as open.
Sigmoidal functions (also known as "activation functions") find multiple applications to neural networks [10]–[15].
In conclusion, we will note that the newly constructed recurrently generable families of sigmoidal and activation functions (see, for instance [16]–[28]) can be used with success in creating a new higher order recurrent neural networks (Fig. 7).
Of course the specialists working in this important area also have the task of exploring eventual possibility for minimizing CPU–time using the existing and new ones FANS.

ACKNOWLEDGMENTS

This work has been supported by the project FP17–FMI–008 of Department for Scientific Research, Paisii Hilendarski University of Plovdiv.

REFERENCES

1. K. Goulianis, A. Margaris, I. Refandis, K. Diamantaras, Solving polynomial systems using a fast adaptive back propagation–type neural network algorithm, Euro. Jnl of Applied Mathematics, 2017, 1–37.
2. D. Manocha, Solving systems of polynomial equations, IEEE Comput. Graph. Appl., 14 (2), 1994, 46–55
3. M. Mamat, K. Muhamad, M.Y. Waziri, Trapezoidal Broyden's method for solving systems of nonlinear equations, Appl. Math. Sci. 8 (6), 2014, 251–260
4. T. Nguyen, Neural network architecture of solving nonlinear equation systems, Electron. Lett., 29 (16), 1993, 1403–1405
5. K. Mathia, R. Saeks, Solving nonlinear equations using recurrent neural networks, In: World Congress on Neural Networks (WCNN' 95), Washington DC, pp. 76–80
6. X. Guo, Z. Zeng, The neural–network approaches to solve nonlinear equation, J. Comput., 5 (3), 2010, 410–416
7. A. Margaris, M. Adamopoulos, Solving nonlinear algebraic systems using artificial neural networks, In: Proceedings of the 10th international conference of engineering applications of artificial neural networks, Thessaloniki, Greece, 2007
8. N. Hahm, B.I. Hong, A note on neural network approximation with a sigmoidal function, Appl. Math. Sci., 10 (42), 2016, 2075–2085
9. K. Goulianis, A. Margaris, I. Refandis, K. Diamantaras, T. Papadimitriou, A back propagation–type neural network architecture for solving the complete $n \times n$ nonlinear algebraic system of equations, Adv. in Pure Math., 6, 2016, 455–480.
10. D. Costarelli, R. Spigler, Approximation results for neural network operators activated by sigmoidal functions, Neural Networks 44, 101–106 (2013).
11. D. Costarelli, G. Vinti, Pointwise and uniform approximation by multivariate neural network operators of the max-product type, Neural Networks, 2016; doi:10.1016/j.neunet.2016.06.002; http://www.sciencedirect.com/science/article/pii/S0893608016300685.
12. D. Costarelli, R. Spigler, Solving numerically nonlinear systems of balance laws by multivariate sigmoidal functions approximation, Computational and Applied Mathematics 2016; DOI:10.1007/s40314-016-0334-8 http://link.springer.com/article/10.1007/s40314-016-0334-8
13. D. Costarelli, G. Vinti, Convergence for a family of neural network operators in Orlicz spaces, Mathematische Nachrichten 2016; doi:10.1002/mana.20160006
14. N. Gyliyev, V. Ismailov, On the approximation by single hidden layer feed forward neural networks with fixed weights, arXiv:1708.06219v1 [cs. NE] 21 August (2017)
15. N. Guliyev, V. Ismailov, A single hidden layer feedforward network with only one neuron in the hidden layer san approximate any univariate function, Neural Computation, 28 2016, 1289–1304.

16. N. Kyurkchiev, A. Iliev, S. Markov, Some techniques for recurrence generating of activation functions, LAP LAMBERT Academic Publishing, 2017; ISBN 978-3-330-33143-3.
17. Malinova, A., A. Golev, A. Iliev, N. Kyurkchiev, A Family of Recurrence Generating Activation Functions Based on Gudermann Function, International Journal of Engineering Researches and Management Studies, 4(8), 2017, 38-48, ISSN: 2394-7659
18. N. Kyurkchiev, A. Iliev, A note on the new Fibonacci hyperbolic tangent activation function, Int. J. of Innovative Science, Engineering and technology, 4 (5), 2017.
19. Golev, A., A. Iliev, N. Kyurkchiev, A Note on the Soboleva' Modified Hyperbolic Tangent Activation Function, International Journal of Innovative Science Engineering and Technology, 4 (6), 2017.
20. N. Kyurkchiev, S. Markov, Sigmoid functions: Some Approximation and Modelling Aspects. (LAP LAMBERT Academic Publishing, Saarbrucken, 2015); ISBN 978-3-659-76045-7.
21. V. Kyurkchiev, A. Iliev, N. Kyurkchiev, On some families of recurrence generated activation functions, Int. J. of Sci. Eng. And Appl. Sci., 3 (3), 2017; ISSN:2395-370
22. N. Kyurkchiev, A family of recurrence generated sigmoidal functions based on the Verhulst logistic function. Some approximation and modeling aspects, Biomath Communications, 3 (2), (2016); 18 pp., http://www.biomathforum.org/biomath/index.php/conference/article/view/789/873
23. N. Kyurkchiev, A note on the Volmer's activation function, Compt. rend. Acad. bulg. Sci., 70 (6), 2017, 769-776
24. A. Iliev, N. Kyurkchiev, S. Markov, A family of recurrence generated parametric activation functions with applications to neural networks, International Journal on Research Innovations in Engineering Science and Technology, 2 (1), 60–68, 2017.
25. N. Kyurkchiev, S. Markov, Hausdorff Approximation of the Sign Function by a Class of Parametric Activation Functions, Biomath Communications, 3 (2), (2016); 14 pp., http://dx.doi.org/10.11145/bmc.2016.12.217
26. N. Kyurkchiev, A. Iliev, S. Markov, Families of recurrence generated three and four parametric activation functions, Int. J. Sci. Res. and Development, 4 (12), 746-750, 2017.
27. V. Kyurkchiev, N. Kyurkchiev, A family of recurrence generated functions based oh Half-hyperbolic tangent activation functions, Biomedical Statistics and Informatics, 2 (3). 87-94, 2017,
28. Iliev, A., N. Kyurkchiev, S. Markov, A Note on the New Activation Function of Gompertz Type, Biomath Communications, 4(2), 2017 (accepted).
29. Petkovic, M., D. Herceg, S. Ilic, Point Estimation Theory and its Applications, Institute of Mathematics, Novi Sad, 1997
30. Hristov V., N. Kyurkchiev, A note on the globally convergent properties of the Weierstrass-Dochev method, Approximation Theory, A volume dedicated to Bl. Sendov (B. Boyanov Ed.), DARBA, Sofia, 2002, 231-240.
31. Kyurkchiev N., Initial approximations and root finding methods, Mathematical Research Vol. 104, Wiley-VCH Verlag Berlin GmbH, Berlin, 1998.
32. Iliev, A., I. Iliev. Numerical method with order t for solving system nonlinear equations, Collection of scientific works, 30 years FMI, Plovdiv, 2000, 105-112.
33. A. Iliev, N. Kyurkchiev, Nontrivial methods in numerical analysis, Selected topics in numerical analysis, LAP LAMBERT Academic Publishing, GmbH Co. KG, Saarbrucken, 2010.
34. Kyurkchiev N., Some remarks on Weierstrass root-finding method, Compt. rend. Acad. bulg. Sci., 46, No 8, 1993, 17–20.
35. Petkovic, M., N. Kyurkchiev, A note on the convergence of the Weierstrass SOR method for polynomial roots, J. Comput. Appl. Math., 80 (1), 1997, 163–168
36. Valchanov, N., A. Golev, A. Iliev, On the critical points of Kyurkchiev's method for solving algebraic equations, Serdica J. Computing, 9 (1), 2015, 27–34
37. Hristov, V., N. Kyurkchiev, A. Iliev, On the solutions of polynomial systems obtained by Weierstrass method, C. R. Acad. Bulg. Sci., 62, No 11, 2009, 1371–1376
38. P. Proinov, New general convergence theory for iterative processes and its applications to Newton–Kantorovich type theorems, Journal of Complexity 26, 2010, 3–42.
39. P. Proinov, General local convergence theory for a class of iterative processes and its applications to Newton's method, Journal of Complexity, 25(1), 2009, 38-62.

40. P. Proinov, M. Vasileva, On the convergence of high-order Ehrlich-type iterative methods for approximating all zeros of a polynomial simultaneously, Journal of Inequalities and Applications, 336, 2015, doi:10.1186/s13660-015-0855-5
41. P. Proinov, S. Cholakov, Semilocal convergence of Chebyshev-like root-finding method for simultaneous approximation of polynomial zeros, Applied Mathematics and Computation, 236, 2014, 669-682.
42. P. Proinov, S. Ivanov, On the Convergence of Halley's Method for Multiple Polynomial Zeros, Mediterranean Journal of Mathematics, 2015, 12(2), 555–572.
43. P. Proinov, M. Petkova, Convergence of the Weierstrass method for simultaneous approximation of polynomial zeros, C. R. Acad. Bulg. Sci., 66(6), 2013, 809-818.
44. Valchanov, N., Modeling and Implementation of Extensible Modular Information Systems, PhD Thesis, University of Plovdiv, Plovdiv, 2012
45. Weierstrass K., Neuer Beweis des Satzes, dass jede ganze rationale Funktion einer Veranderlichen dargestellt werden kann als ein Product aus linearen Funktionen derselben Veranderlichen, Ges. Werke, 3, 1903, 251-269 (Johnson Reprint Corp., New York 1967).