CHEBYSHEV POLYNOMIAL APPROXIMATION FOR ACTIVATION SIGMOID FUNCTION

*Miroslav Vlček**

Abstract: An alternative polynomial approximation for the activation sigmoid function is developed here. It can considerably simplify the input/output operations of a neural network. The recursive algorithm is found for Chebyshev expansion of all constituting polynomials.

Key words: *Sigmoid function, Chebyshev polynomials, recursive algorithms*

Received: January 30, 2012
Revised and accepted: August 3, 2012

1. Introduction

We assume a single neural network consisting of a distinct number of input nodes and one output node [6]. In order to evaluate the output we have to apply the standard activation sigmoid function $\sigma(y)$ to the sum $y = \sum_{i=1}^{N} w_i x_i$, where x_i are the values computed by the node's predecessor, and w_i are the weights of the corresponding edges. As the activation sigmoid function $\sigma(y)$ of a neural network satisfies the Riemann integrable condition, it can be approximated by the Chebyshev series. We present the recursive algorithm [8] for Chebyshev approximation of the activation sigmoid function and its natural generalization to a multiple number of inputs. These results can be applied in several neural networks [5], such as multilayer perceptron (MLP), wavelet networks, radial basis function networks (RBFN), piecewise smooth networks (PWSN), the time delay input multilayer perceptron, general regression neural networks (GRNN), recurrent neural networks, and the unified model UM. A common activation sigmoid function $\sigma(y)$ is usually represented through

$$\sigma(y) = \frac{e^y - e^{-y}}{e^y + e^{-y}} = \tanh y, \tag{1}$$

$$\sigma(y) = \frac{1}{1 + e^{-y}}. \tag{2}$$

*Miroslav Vlcek
Czech Technical University in Prague, Faculty of Transportation Sciences, Department of Applied Mathematics, E-mail: vlcek@fd.cvut.cz

There are different approaches for evaluating these functions when using digital implementations, such as a truncated series expansion [5], look-up tables,

$$\sigma(y) = \tanh y \approx y - \frac{y^3}{3} + \frac{2y^5}{15}, \tag{3}$$

or linear piecewise approximation [3], [4]

$$\sigma(y) = \frac{1}{1+e^{-y}} \approx c_1 y + c_2. \tag{4}$$

In the following sections we present an efficient algorithm for coefficients $a(n)$ for Chebyshev representation of the activation sigma function

$$\sigma(y) = \sum_{n=0}^{N} a(n) T_n(w), \tag{5}$$

and develop an explicit expansion of $\sigma(x+y)$ in bilinear form employing Chebyshev polynomials.

2. Sigmoid Function and Polynomial Representation

The Bernstein basis functions of degree n on $t \in (0,1)$ are defined in [2] by

$$b_{n,k}(t) = \left(\begin{array}{c} n \\ k \end{array} \right) t^k \, (1-t)^{n-k}. \tag{6}$$

By simple transformation $w = 2\,t - 1$ we obtain the basis functions on interval $w \in (-1,1)$ with

$$b_{n,k}(w) = 2^{-n} \left(\begin{array}{c} n \\ k \end{array} \right) (1+w)^k \, (1-w)^{n-k}. \tag{7}$$

In [8] we have recognized that the integral of a normalized Bernstein basis function

$$C_{p,q}(w) = \frac{p+q+1}{2} \int dw \left(\begin{array}{c} p+q \\ q \end{array} \right) \left(\frac{1+w}{2} \right)^q \left(\frac{1-w}{2} \right)^p \tag{8}$$

gives the maximally flat step function in the form

$$C_{p,q}(w) = \left(\frac{1+w}{2} \right)^{q+1} \sum_{\mu=0}^{p} \left(\begin{array}{c} \mu+q \\ \mu \end{array} \right) \left(\frac{1-w}{2} \right)^{\mu}. \tag{9}$$

We have also derived the differential equation

$$(1-w^2) C''_{p,q}(w) + [p-q+(p+q)w] C'_{p,q}(w) = 0 \tag{10}$$

from which the expansion of $C_{p,q}(w)$ in Chebyshev polynomials follows

$$C_{p,q}(w) = \sum_{n=0}^{N} a(n) T_n(w), \tag{11}$$

where $N = p + q + 1$. As $\frac{d}{dw} T_n(w) = n U_{n-1}(w)$ we can write the Chebyshev expansion of the first derivative $\frac{d}{dw} C_{p,q}(w)$

$$\frac{d}{dw} C_{p,q}(w) = \sum_{n=1}^{N} \alpha(n) U_{n-1}(w). \tag{12}$$

A maximally flat step function is a continuous function with all consecutive deriva-

given	p and q, $N = p + q + 1$
initialization	$\alpha(N) = (-1)^p \, 2^{-2(p+q)} \, \frac{N}{2} \begin{pmatrix} p+q \\ q \end{pmatrix}$
	$\alpha(N-1) = 2\,(q-p)\alpha(N)$
body	
(*for* $\mu = N-1$ *to* 2)	
	$\frac{N+1-\mu}{2}\alpha(\mu-1) = (q-p)\alpha(\mu) - \frac{N+1+\mu}{2}\alpha(\mu+1)$
(*end loop on* μ)	
(*for* $\mu = N$ *to* 1)	
	$a(\mu) = \frac{\alpha(\mu)}{\mu}$
(*end loop on* μ)	
	$a(0) = 1 - \sum_{n=1}^{N} a(n)$

Tab. I *Recursive algorithm for the evaluation of coefficients* $\alpha(n)$ *and* $a(n)$.

tives, and it can be easily identified as a sigmoid function

$$\sigma(w) \equiv C_{p,q}(w) = \left(\frac{1+w}{2}\right)^{q+1} \sum_{\mu=0}^{p} \begin{pmatrix} \mu+q \\ \mu \end{pmatrix} \left(\frac{1-w}{2}\right)^{\mu} = \sum_{n=0}^{N} a(n) T_n(w). \tag{13}$$

This sigmoid function is confined to the interval $(-1, 1)$. The study of a standard representation of the sigmoid function is typically limited to the range $(-8, 8)$ [3]. We can compare our definition with the standard sigmoid function $\sigma(w) = \frac{1}{1+e^{-8w}}$ reduced to this interval and conclude that the step function $C_{11,11}(w)$ has equivalent properties with the difference $\Delta(w) = C_{11,11}(w) - \sigma(w)$ of 2%. Using different values for p, q, we can move the origin of the switching process along the w axis – see Fig. 3.

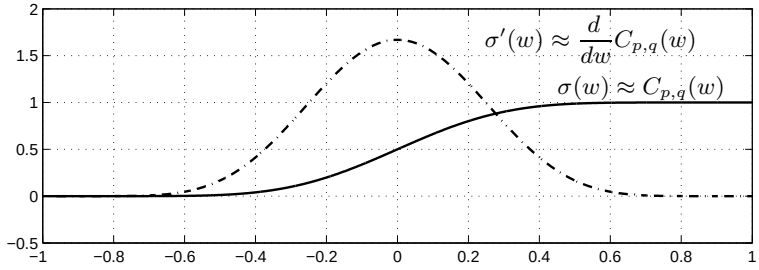

Fig. 1 *Step function* $C_{8,8}(w)$ *and its derivative related to a sigmoid function.*

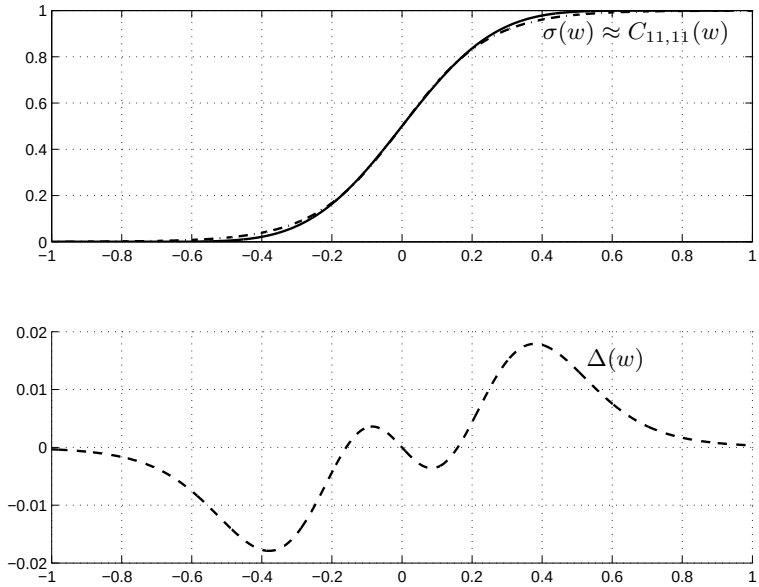

Fig. 2 *Step function* $C_{11,11}(w)$*, sigmoid function* $\sigma(w) = \dfrac{1}{1+e^{-8w}}$*, and their difference* $\Delta(w)$*.*

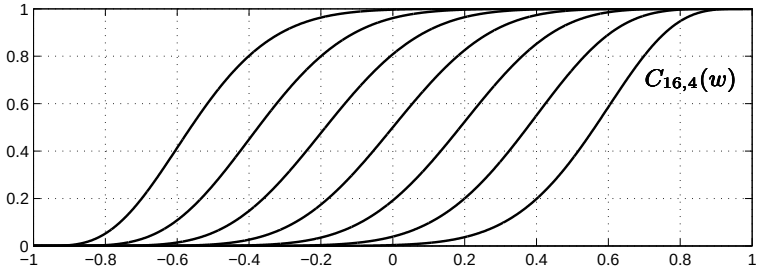

Fig. 3 *Step functions* $C_{4,16}(w)C_{6,14}(w), C_{8,12}(w), C_{10,10}(w), C_{12,8}(w), C_{14,6}(w), C_{16,4}(w)$*.*

3. Input/Output Operation of a Simple Neural Network

For a neural network it is important to represent consistently the addition of arguments $w = x + y$. We have developed addition and multiplication theorems for Chebyshev polynomials [7] in alternative forms

$$T_n(x+y) = \frac{n}{2} \sum_{k=0}^{n} \sum_{\ell=0}^{k} \frac{1}{k} \begin{pmatrix} k \\ \ell \end{pmatrix} C_{n-k}^{k}(x)\, T_{k-2\ell}(y), \tag{14}$$

where $C_{n-k}^{k}(x)$ are ultraspherical polynomials [1]. Formula (14) provides a decomposition of the argument $w = x + y$ between two types of orthogonal polynomials, Chebyshev and ultraspherical. In order to make the polynomial representation uniform we can assume eq. (14) in the form

$$T_n(x+y) = \sum_{k=0}^{n} \sum_{\ell=0}^{n} a_{k,\ell}(n) T_k(x) T_\ell(y) \tag{15}$$

and use the standard recursive formula for Chebyshev polynomials

$$T_{n+1}(x+y) + T_{n-1}(x+y) = 2\,(x+y)\, T_n(x+y). \tag{16}$$

Formula (16) is used to develop an algorithm for the matrix $a_{k,\ell}(n)$

$$\begin{aligned}
2\,(x+y)\, T_n(x+y) &= 2\,x\, T_n(x+y) + 2\,y\, T_n(x+y) \\
&= \sum_{k=0}^{n} \sum_{\ell=0}^{n} a_{k,\ell}(n) 2\,x\, T_k(x) T_\ell(y) + \sum_{k=0}^{n} \sum_{\ell=0}^{n} a_{k,\ell}(n) T_k(x) 2\,y\, T_\ell(y) \\
&= \sum_{k=0}^{n} \sum_{\ell=0}^{n} a_{k,\ell}(n) \left(T_{k+1}(x) + T_{k-1}(x)\right) T_\ell(y) \\
&\quad + \sum_{k=0}^{n} \sum_{\ell=0}^{n} a_{k,\ell}(n) T_k(x) \left(T_{\ell+1}(y) + T_{\ell-1}(y)\right) \\
&= \sum_{k=0}^{n} \sum_{\ell=0}^{n} a_{k,\ell}(n) T_{k+1}(x) T_\ell(y) + \sum_{k=0}^{n} \sum_{\ell=0}^{n} a_{k,\ell}(n) T_{k-1}(x) T_\ell(y) \\
&\quad + \sum_{k=0}^{n} \sum_{\ell=0}^{n} a_{k,\ell}(n) T_k(x) T_{\ell+1}(y) + \sum_{k=0}^{n} \sum_{\ell=0}^{n} a_{k,\ell}(n) T_k(x) T_{\ell-1}(y).
\end{aligned} \tag{17}$$

By replacing the summation indexes in a following way $p = k+1, q = \ell$, $p = k-1, q = \ell$, $p = k, q = \ell+1$ and $p = k, q = \ell-1$, we obtain a new set of equations

$$\begin{aligned}
2\,(x+y)\, T_n(x+y) &= \sum_{p=1}^{n+1} \sum_{q=0}^{n} a_{p-1,q}(n) T_p(x) T_q(y) + \sum_{p=-1}^{n-1} \sum_{q=0}^{n} a_{p+1,q}(n) T_p(x) T_q(y) \\
&\quad + \sum_{p=0}^{n} \sum_{q=1}^{n+1} a_{p,q-1}(n) T_p(x) T_q(y) + \sum_{p=0}^{n} \sum_{q=-1}^{n-1} a_{p,q+1}(n) T_p(x) T_q(y).
\end{aligned}$$

Comparing the coefficients with the same degree of polynomials $T_p(x)T_q(y)$, we arrive at a compact formula for the matrix $a_{p,q}(n)$

$$a_{p,q}(n+1)+a_{p,q}(n-1) = (1+\delta_{p,1})\, a_{p-1,q}(n)+a_{p+1,q}(n)+(1+\delta_{1,q})\, a_{p,q-1}(n)+a_{p,q+1}(n). \tag{18}$$

The Kronecker delta $\delta_{p,1}$ appears here due to the fact that $T_{-1}(x) = T_1(x)$ and it contributes to the index $p = 1$ twice. The algorithm produces computation of $T_n(x + y)$ as a bilinear form, for example

$$T_6(x+y) = \begin{bmatrix} 1 \\ T_1(x) \\ T_2(x) \\ T_3(x) \\ T_4(x) \\ T_5(x) \\ T_6(x) \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} 109 & 0 & 138 & 0 & 30 & 0 & 1 \\ 0 & 348 & 0 & 132 & 0 & 12 & 0 \\ 138 & 0 & 168 & 0 & 30 & 0 & 0 \\ 0 & 132 & 0 & 40 & 0 & 0 & 0 \\ 30 & 0 & 30 & 0 & 0 & 0 & 0 \\ 0 & 12 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ T_1(y) \\ T_2(y) \\ T_3(y) \\ T_4(y) \\ T_5(y) \\ T_6(y) \end{bmatrix}, \tag{19}$$

where $\mathbf{v}^{\mathrm{T}}$ denotes transposition of vector $\mathbf{v}$. It is worth noting that the off-diagonal contains the binomial coefficients. In this example $n = 6$ and they are $\binom{n}{0} \equiv \binom{n}{n} = 1$, $2\binom{n}{1} \equiv 2\binom{n}{n-1} = 12$, $2\binom{n}{2} \equiv 2\binom{n}{n-2} = 30$, and $2\binom{n}{3} = 40$.

4. Conclusion

We have developed a polynomial approximation for the activation sigmoid function. The algorithm is based on Chebyshev expansion for all constituting polynomials. Combining equations (13) and (15), we obtain

$$\sigma(x+y) = \sum_{n=0}^{N} a(n) T_n(x+y) = \sum_{n=0}^{N} a(n) \sum_{k=0}^{n} \sum_{\ell=0}^{n} a_{k,\ell}(n) T_k(x) T_\ell(y). \tag{20}$$

The main advantage of representation (20) over the standard sigmoid activation function $\sigma(y)$ consists of decoupling of the *nonlinear switching process,* which is now hidden in coefficients $a(n)$, and the *weighting of the various inputs,* embedded in coefficients $a_{k,\ell}(n)$. This approach enables us to state, that a major objective of future research will concern of finding the uniform and finite input/output operations for a forward neural network as a generalization of a bilinear form

$$\sigma(w_1 x_1 + w_2 x_2) = T_k(x_1) \mathbf{A}_{k,\ell} T_\ell(x_2). \tag{21}$$

We have also arrived at a rather compact form for the derivative of the sigmoid function which mimics eq. (20)

$$\frac{d}{dw}\sigma(w)|_{w=x+y} = \sum_{n=1}^{N} \alpha(n)\, U_{n-1}(x+y) = \sum_{n=1}^{N} \alpha(n) \sum_{k=0}^{n-1} \sum_{\ell=0}^{n-1} \alpha_{k,\ell}(n) U_k(x) U_\ell(y). \tag{22}$$

Acknowledgement

The author wishes to thank the Grant Agency of the Czech Republic for supporting this research through project GAP102/11/1795: *Novel selective transforms for non-stationary signal processing.*

References

[1] Abramowitz M., Stegun I.: Handbook of Mathematical Functions, Dover Publications, New York Inc., 1972.

[2] Meinardus G.: Approximation von Funktionen und ihre numerische Behandlung, Springer-Verlag, Berlin, Goettingen, Heidelberg, New York, 1964.

[3] Armato A., Fanucci L., Pioggia G., De Rossi D.: Low-error approximation of artificial neuron sigmoid function and its derivative, Electronic Letters, **45**, 21, October 2009, pp. 1082–1084.

[4] Basterretxea K., Tarela J. M., del Campo I.: Approximation of sigmoid function and the derivative for hardware implementation of artificial neurons, IEE Proc.-Circuits Devices Syst., **151**, 1, February 2004, pp. 18–24.

[5] Lee T. T., Jeng J. T.: The Chebyshev-Polynomial-Based-Unified Model Neural Network for Function Approximation, IEEE Transactions on Systems, Man, and Cybernetics, **28**, 6, December 1998, pp. 925–935.

[6] Duch W., Jankowski N.: Survey of Neural Transfer Functions, Neural Computing Surveys, 2, 1999, pp. 163–212.

[7] Vlcek M., Hoa P. B.: McClellan Transformation through the Addition and Multiplication Theorem for Chebyshev Polynomials, Proc. MICROCOLL'86, Budapest 1986, pp. 183–184.

[8] Vlcek M., Zahradnik P., Unbehauen R.: Analytic Design of FIR Filters, IEEE Transactions on Signal Processing, **48**, 9, September 2000, pp. 2705–2709.