Back Propagation in a Clifford Algebra

J.K.Pearson and D.L.Bisset

Electronic Engineering Laboratories, The University, Canterbury, Kent CT2 7NT, United Kingdom

Abstract

Recent work has shown that in some cases the phase information of synaptic signal is important in the learning and representation capabilities of networks. Modelling such information with complex valued activation signals is possible, and indeed complex back-propagation algorithms have been derived [3]. Clifford algebras give a way to generalise complex numbers to many dimensions. This paper presents a back propagation for feed-forward networks with Clifford activation values.

1 Definition of a Clifford Algebra.

Clifford algebras have been rediscovered many times see [4] for examples and applications. Part of the success of Clifford algebras is due to the geometric character of the definition. In this paper the geometric content of the algebra will not be relevant. Although in individual applications a Clifford net could be used to process geometric information.

Let $\mathbb{R}^n$ denote the real n-dimensional vector space. We shall denote the Universal Clifford Algebra derived from $\mathbb{R}^n$ by $C_{n,s}$, where $1 \leq s \leq n$. The Universal Clifford Algebra is defined as a relation on basis vectors of $\mathbb{R}^n$. Let $e_1, e_2, \ldots, e_n$ be a set of basis vectors for $\mathbb{R}^n$, then $C_{n,s}$ is a 2^n dimensional vector space with basis elements:

$$\{e_A = e_{h_1 \ldots h_r} | A = (h_1, \ldots, h_r) \in \mathcal{P}(\mathbb{N}), 1 \leq h_1 < \ldots < h_r \leq n\}.$$

To give the vector space an algebraic structure the following multiplication rules are used:

$$e_i^2 = 1 \quad , \quad i = 1, \ldots, s \tag{1}$$

$$e_i^2 = -1 \quad , \quad i = s+1, \ldots, n \tag{2}$$

$$e_i e_j = -e_j e_i \quad , \quad i \neq j. \tag{3}$$

With for $1, \leq h_1, < \ldots h_r \leq n$, $e_{h_1} \cdot e_{h_2} \cdots e_{h_r} = e_{h_1 \ldots h_r}$.

This can be expressed more compactly in the following way,

$$e_A e_B = (-1)^{\#((A \cap B) \setminus S)} (-1)^{p(A,B)} e_{A \Delta B}, \tag{4}$$

where S stands for the set $1, \ldots s$, $p(A, B) = \sum_{j \in B} p'(A, j)$, $\quad p'(A, j) = \#\{i \in A | i > j\}$, Δ symmetric difference and the sets A, B and $A \Delta B$ are ordered in the prescribed way.

In this paper Clifford numbers are denoted as,

$$x = \sum_A x_A e_A, \tag{5}$$

with x_A a real number e_A a formal symbol. In general A,B and C will be used to range over sets of the form given in (1).

Also the notation $[x]_A$ will denote the A'th part of the Clifford number x, for example if $x = 1 + 2e_1 + 10e_12$ in the algebra $C_{2,2}$ then $[x]_\emptyset = 1$, $[x]_{\{2\}} = 0$ and $[x]_{\{1,2\}} = 10$ etc. For a notational convinence, $[x]_\emptyset$ will be written $[x]_0$, $[x]_{\{2\}}$ will be $[x]_2$ etc.

2 Clifford Back Propagation

A Clifford network is defined in a similar way to a conventional feed forward network. The only difference being all weights and activation values are Clifford numbers from some particular Clifford algebra.

An activation function $f(x)$ defined on a Clifford algebra can be written as,

$$f(x) = \sum_A u_A e_A. \tag{6}$$

Where each u_A are mappings from the Clifford algebra to the real numbers.

Thus the output o_j of the j'th neuron can be written as,

$$o_j = f(x_j) = \sum_A u_A^j e_A, \tag{7}$$

with,

$$x_j = \sum_A x_A^j = \sum_{l=1}^{n} W_{jl} X_{jl}. \tag{8}$$

It is important to notice that since the algebra is non commuative the order of multiplication in the above equation is important.

The error measure E of the net is defined in a similar way to the conventional case,

$$E = \frac{1}{2} \sum_k \sum_i [d_k - o_k]_i^2. \tag{9}$$

Where d_k is the desired output of neuron k, o_k is the actual output of neuron k, and $[x]_i$ represents the i'th part of the Clifford number x.

To minimise the error function E it is necessary to minimise E with respect to each element of each weight vector. To take into account the functional dependencies the following equations is derived (W_i^{jl} is the i'th component of the weight W_{jl})

$$\frac{\partial E}{\partial W_i^{jl}} = \sum_A \left(\frac{\partial E}{\partial u_A^j} \left(\sum_B \frac{\partial u_A^j}{\partial x_B^j} \frac{\partial x_B^j}{W_i^{jl}} \right) \right). \tag{10}$$

The partial derivatives $\frac{\partial x_B}{\partial W_A^{jl}}$ are as follows,

$$\frac{\partial x_B}{\partial W_A^{jl}} = \frac{[W_{jl} X_{jl}]_B}{\partial W_A^{jl}} = (-1)^k X_C^{jl}, \tag{11}$$

where, C, is a set satisfying the following condition $A \Delta C = B$ and

$$k = \#((A \cap C) \setminus S)(-1)^{p(A,C)}, \tag{12}$$

with the function p defined as above. As an example in the algebra $C_{2,2}$,

$$\begin{array}{||c|cccc||}
\hline
\frac{\partial x_\alpha}{\partial W_\beta^{jl}} & \alpha = 0 & 1 & 2 & 12 \\
\hline
\beta = 0 & X_0^{jl} & X_1^{jl} & X_2^{jl} & X_{12}^{jl} \\
1 & X_1^{jl} & X_0^{jl} & X_{12}^{jl} & X_2^{jl} \\
2 & X_2^{jl} & -X_{12}^{jl} & X_0^{jl} & -X_1^{jl} \\
12 & -X_{12}^{jl} & X_2^{jl} & -X_1^{jl} & X_0^{jl} \\
\hline
\end{array} \tag{13}$$

The error derivative $\frac{\partial E}{\partial u_i^j}$ is now easy to calculate. If j is an output neuron then,

$$\frac{\partial E}{\partial u_i^j} = -([d_k]_i - u_i^j). \tag{14}$$

If the neuron is not an output unit the chain rule must be used. The net input to neuron k is therefore,

$$x_k = \sum_A x_A^k e_A = \sum_l W_{kl} X_{kl},$$

with,

$$X_{kl} = \sum_A u_A^{kl} e_A$$

and

$$W^{kl} = \sum_A W_A^{kl} e_A.$$

The terms $\frac{\partial x_\alpha}{\partial u_\beta}$ can be calculated as before, and will be simler in form to (11).

Therefore using the chain rule,

$$\frac{\partial E}{\partial u_i^j} = \sum_k \frac{\partial E}{\partial u_i^k} \left(\sum_{A,B} \frac{\partial u_\alpha^k}{\partial x_\beta^k} \frac{\partial x_\beta^k}{\partial u_i^j} \right) \tag{15}$$

With k running over the neurons that receive input from neuron j.

3 Choice of Activation function.

As has been observed in [3], in the complex domain analytic functions are not suitable activation functions. In the Clifford domain the situation is even worse, because there is no suitable notion of an analytic function. Further because a Clifford algebra is not in general a division algebra, it is not possible to define higher dimensional analogues of the sigmoid function.

The function,

$$f(x) = \frac{x}{c + \frac{1}{r}|x|} \tag{16}$$

where c and r are real positive constants, and $|x|$ is the norm,

$$|x| = \sqrt{\sum_A [x]_A^2} \tag{17}$$

will be a sutiable activation function $f(x)$ and will have all the suitable properties, as stated in [3] in the Clifford domain as it does in the complex domain.

The partial derivatives as follows, for $A = B$,

$$\frac{\partial u_A}{x_B} = \begin{cases} \frac{1}{c+\frac{1}{r}|x|} - \frac{x_A^2}{(c+\frac{1}{r}|x|)^2 r|x|} & \text{if } |x| > 0 \\ \frac{1}{c} & \text{if } |x| = 0. \end{cases} \tag{18}$$

If $A \neq B$ then,

$$\frac{\partial u_A}{x_B} = \begin{cases} -\frac{x_A x_B}{(c+\frac{1}{r}|x|)^2 r|x|} & \text{if } |x| > 0 \\ 0 & \text{if } |x| = 0 \end{cases} \tag{19}$$

4 Applications

If phase information is important in biological systems then complex domain networks are of significance. Although Clifford networks have no direct biological basis, they provide the ability to represent multidimensional signals in a coherent way. The Authors are currently engaged in experimental work with Clifford networks and results will be reported in further publications.

5 Acknowledgements

The Authors wish to thank Prof. Roy Chisholm and Dr. Alan Common, for useful discussions on Clifford algebras, and acknowledge the financial support of the SERC and the systems research division of British Telecom.

References

[1] G.E.Hinton D.E.Rumelhart and R.J.Williams. Learning internal representations by error propagation. In *Parallel Distributed Processing, Volume 1*, chapter 8. MIT Press, 1986.

[2] R.Delenghe F.Brackx and F.Sommen. *Clifford Analysis*. Research notes in mathematics; 76. Pitman Advanced Publishing Program, 1982.

[3] George M. Georgiou and Cris Koutsougeras. Complex domain backpropagation. To appear in the IEEE Transactions on Circuits and Systems, 1992.

[4] J.S.R.Chisholm and A.K.Common, editors. *Clifford Algebras and Their Applications in Mathematical Physics*. NATO ASI Series Reidel, Sept 1985. Proceedings of the NATO and SERC Workshop, Canterbury.