Elementarfunktionen des Neurons[79]

Betrachten wir ein gewichtetes, neuronales Netz unendlich lange, so bemerken wir, daß sich dessen Werte ab und an ändern. Ein wirklich "statisches" Netz, dessen Werte gelernt werden könnten, existiert folglich nur in Idealisierung. Ordnen wir den Signalwerten Zeitpunkte zu, so fällt uns auf, daß auch jedes statische Netz als dynamisches angesehen werden muß: Gewichtete, neuronale Netze, wie Zustandsautomaten (state machines) oder digitale Filter ordnen sich Interferenznetzen unter, sind spezielle Strukturen von Interferenznetzen mit einer Möglichkeit der (quasi-statischen) Betrachtung.

Bezieht man nadelartige Impulse und langsame Leitgeschwindigkeiten in die funktionelle Untersuchung von Neuronen mit ein, kommt man schnell zum Schluß, daß Neuronen völlig anders funktionieren müssen, als bisher angenommen.

Beschäftigte sich das Neurocomputing bislang wesentlich mit der Rolle synaptischer Stärken (weights, Gewichte oder Schwellwerte genannt) deren Training von entscheidender Bedeutung für die Lernfähigkeit angesehen wurde, kommt die biologische Implementation um die Anerkennung eines wesentlichen Umstandes nicht umhin: sämtliche Signale sind schmale Nadeln (Spikes oder Impulse), deren geometrische Impulslänge s in Analogie zur physikalischen Wellenlänge $s = vt$ bei Leitgeschwindigkeiten v im Bereich Mikrometer bis Millimeter pro Sekunde von Mikrometern bis zu Millimetern reicht.

Ein Gatter oder Neuron aber, dessen Eingänge nicht zur selben Zeit ein Aktionspotential besitzen, arbeitet recht unzuverlässig. Also haben wir - *vor allen anderen* - die Frage nach Gleichzeitigkeit zu stellen. Wir haben nach Orten von Interferenz von Impulsen zu fragen.

Dem untergeordnet erscheint dann erst die Frage nach dem Gewicht: Klassisch neuronale Netze implizieren häufig eine Gleichzeitigkeit an Gattereingängen, die im biologischen System nicht vorausgesetzt werden kann.

Wir entdecken, daß bislang unbekannte, neuronale Grundfunktionen schon durch eine einfache Verfolgung von Verzögerungszeiten (Delays) beobachtbar sein müssten.

Das (lineare) Separationsproblem (Minsky/Papert "Perceptrons", 1969) und dessen mathematische Behandlung durch Kolmogorov 1983 ordnen sich durchaus hier ein. Hopfields "Backpropagation" Algorithmus bereitet hingegen schon eher Schwierigkeiten, da unzutreffende Voraussetzungen dessen Anwendung zunächst nahezu unmöglich erscheinen lassen.

Zum Verständnis des Folgenden sei angenommen, daß (sofern nicht anders erwähnt) multiplikative (AND) Verknüpfung vorliegen möge, daß ein Gatter oder Neuron also nur feuern kann, wenn alle Eingänge zeitgleich eine Pulsspitze erhalten.

[79] Anm. des Authors: Der Titel des Originals 1993 dieses Kapitels hieß: Interferenzlogik und Impulslogik. Eine kritische Aufarbeitung führte 1995 zur kompletten Neufassung des Kapitels.

Dynamische Codegenerierung (Bursts)

An verschiedenen Stellen des Nervensystems können wir beobachten, daß Pulse zu Bursts (d) gruppiert erscheinen. Andererseits ist bekannt, daß Nervenzellen baumartig vernetzt (b) ineinander greifen. Berücksichtigen wir noch, daß bekannt ist, daß die dünnsten Nervenfasern die geringsten Leitgeschwindigkeiten (also die relativ höchsten Delays) aufweisen, läßt sich eine nervliche Verbindung dynamisch mit einer einfachen Ersatzschaltung (a) modellieren. Diese Ersatzschaltung besitzt bemerkenswerte Eigenschaften: Für ein niedrig gewähltes Bias vom logischen OR-Typ gibt sie auf einen einzelnen Eingabeimpuls x(t) einen Burst y(t) mit dem Delay-Vektor M aus. Wir nennen diese Elementareigenschaft des Neurons 'Codegenerierung'. (Erstveröffentlichung im SAMS-Aufsatz Heinz 1994, siehe Veröffentlichungsliste).

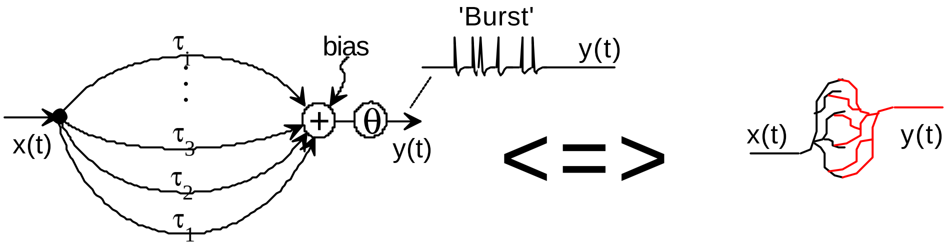

Bild: links: Einfachste Ersatzschaltung einer Koppelstelle zweier Neuronen; rechts: Nervenmodell

Dynamische Codedetektion (Burstdetektion)

Wird das Bias sehr hoch gewählt (AND-Typ), so reagiert die Schaltung nicht mehr auf Einzelimpulse. Statt dessen detektiert sie nur noch eine einzige Impulsgruppe mit definiertem Vektor M*. Wir nennen diese zweite Elementareigenschaft des Neurons 'Codedetektion'.

Adresse eines Burst (Dynamische Adressierung)

Betrachten wir die Neuronen im Beispielbild. Die Neuronen 1 bis 3 mögen verschieden Verzögerungsvektoren M1 bis M3 besitzen. Dazu mögen paarweise inverse Vektoren M1* bis M3* existieren.

Dann springt (abgesehen von nicht betrachteten Permutationen) die Erregung i.a. nur von M1 auf M1*, von M2 auf M2* und von M3 auf M3* über und visa versa.

Finden wir nun zwei Nervenzellen, deren Verzögerungsvektoren dem Zusammenhang

$M + M^* = T$

$$M = \begin{pmatrix} \tau_1 \\ \tau_2 \\ \vdots \\ \tau_i \end{pmatrix}; \quad T = \tau \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}; \quad M* = \begin{pmatrix} \tau - \tau_1 \\ \tau - \tau_2 \\ \vdots \\ \tau - \tau_i \end{pmatrix},$$

$$\overset{n}{\underset{i=1}{\forall}} \tau_j + \tau_j^* = const. = '$$

mit $T = \tau$ {1} und mit dem Einheitsvektor {1} genügen, so sendet die eine einen Burst y(t) aus, die nur die andere empfangen kann, und den sie zu einem Einzelimpuls z(t) zurückverwandelt. Senden mehrere Nervenzellen, die unterschiedliche Verzögerungsvektoren besitzen, auf eine gemeinsame Übertragungsleitung, und existieren auf der Empfangsseite Nervenzellen, deren jede nur einen spezifischen Burst selektieren kann, so finden wir ein Übertragungsprinzip, daß die Übertragung verschiedener Datenströme auf einer gemeinsamen Leitung erklären kann. Dieses Prinzip sollte 'Datenadressierung' genannt werden.

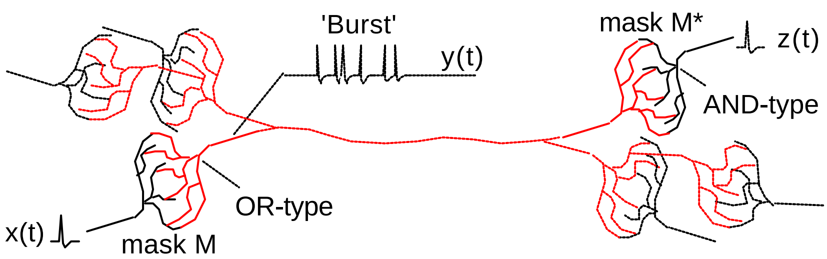

Bild: Codegenerierung ('Burst') und Codedetektion an zwei verkoppelten Neuronen
Ein ankommender Einzelimpuls (links) erzeugt u.U. einen Burst (mittig),
dieser wird u.U. in einen Einzelimpuls zurückgewandelt (rechts)

Dynamische Nachbarschaftsinhibition

Dasselbe Prinzip verhindert auch ein Überspringen von Erregung zwischen benachbarten Neuronen. Besitzen inkremental sehr eng benachbarte Neuronen Verbindungsstellen sowie identische Struktur, so besitzen sie denselben Verzögerungsvektor M. Da für eine Adressierung aber M + M* = T erfüllt sein muß und folglich M* = T - M gilt, die geometrische Struktur allerdings M* = M fordert, gibt es für M und M* nur die trivialle Lösung tau/2 {1}, die bedeutet, daß sich alle Synapsen beider Neurone im gleichen Radius r = tau/2 um das Soma herum befinden. Somit erklärt dieses Prinzip auf Grundlage von Delays, unabhängig von z.B. inhibitorisch wirkenden Synapsenstärken eine dynamisch wirkende 'Nachbarschaftsinhibition'. Man könnte sagen, die Erregung kann von ihrer dynamischen Natur her nicht zwischen zwei identischen, an gleiche Ortsknoten angeschlossenen Neuronen überspringen.

Dynamische Pegelgenerierung

Besitzt das detektierende Neuron zusätzlich eine integrierende Eigenschaft, so kann ein detektierter Burst zu einer Potentialanhebung führen. Wir nennen diese Eigenschaft Pegelgenerierung. Ist die integrierende Wirkung stark genug, so können auf diese Weise gleitende Hintergrundpotentiale geschaffen werden, die ihrerseits wieder benötigt werden, um Zoom- oder Move-Aktionen zu steuern. Ich persönlich glaube, wir messen genau diese Steuerpotentiale im EEG. Eigene

EEG-Experimente (siehe dort) erbrachten keine hinreichend gute, interferenzielle Rekonstruierbarkeit. Dies würde dazu passen.

Simulative Verifikation

Das Experiment von Heinz, Puschmann und Schoel (1994): siehe virtuelle Experimente.

Simulationsergebnisse sind z.B. zu finden im Postscript-File 'Projections and Coding in Pulspropagating Networks - Virtual Experiments' Seiten 18 und 23, zu laden auf der Seite der Veröffentlichungen, oder auf der Seite 'Projektionen und Interferenzmuster'.

Zusammengefaßt

Als dynamisch wirkende, neuronale Elementarfunktionen können wir am elementaren Interferenzkreis beobachten und simulativ verifizieren:

- Codegenerierung
- Codedetektion
- Datenadressierung
- Nachbarschaftsinhibition
- Pegelgewinnung

Die Eigenschaften konnten in der Simulation erstmals bestätigt werden mit dem Simulator 'Neuronet' an der FHTW Berlin am 20.10.1994 von Puschmann, Schoel und Heinz.