

# **Erkennen von Korona-Entladungen in der TE-Meßtechnik mittels neuronaler Netzwerke**

M. Lauersdorf, Universität Stuttgart,  
Institut für Energieübertragung und Hochspannungstechnik

## **Einleitung**

Die TE-Diagnostik ist inzwischen zu einem unverzichtbaren Mittel zur zerstörungsfreien Prüfung von Betriebsmitteln der Hochspannungstechnik geworden. Vor allem für die On-line-Überwachung von Isolationssystemen im Netzbetrieb sind bei der breitbandigen Teilentladungsmessung zwei Ziele von zentraler Bedeutung:

1. Unterdrückung der dem Meßsignal überlagerten Störungen.
2. Klassifizierung und Lokalisierung der erfaßten TE

In der Vergangenheit gab es z.T. recht erfolgreiche Versuche, die bei der Messung auftretenden bekanntesten Störer zu erkennen und zu unterdrücken. So ist es inzwischen möglich, periodische Sinusstörer zu bestimmen und digital aus dem Meßsignal herauszufiltern [1]. Netzsynchrone, zyklisch auftretende, TE-ähnliche Impulse können mittlerweile ebenso effizient erkannt und ausgeblendet werden [2]. Die Problematik der Erkennung von in Häufigkeit und Impulsform TE-ähnlicher, dem Nutzsignal überlagerter Störimpulse, wie es z.B. bei Koronastörern der Fall ist, führte zu einer vertieften Betrachtung der Idee einer "Musteridentifikation" d.h. einer Klassifikation von TE-ähnlichen Störsignalen aufgrund charakteristischer Merkmale. Bezüglich der Klassifikation von TE-Impulsen sind schon seit längerer Zeit die wesentlichsten Erscheinungsbilder von TE-Signalaufzeichnungen bekannt [4]. Dadurch erst konnte das neue Gebiet der 'TE-Mustererkennung' erschlossen werden [3, 5, 6, 7]. Die Automatisierung einer solchen Mustererkennung wurde mit der Entwicklung und Pflege von umfangreichen Expertensystemen möglich [z.B. 6]. Es hat sich gezeigt, daß ein TE-Muster, für jede Meßanordnung und jeden Prüfling spezifisch, entsprechend dem oszillographischen Erscheinungsbild schon durch wenige signifikante Parameter, über die gesamte Netzperiode betrachtet, klassifiziert werden kann.

## **1. Beschreibung von TE-Mustern**

Ein TE-Muster ist ein Impuls-Erscheinungsbild einer TE-Signalaufzeichnung, das sich über eine gesamte Netzperiode hinweg erstreckt. Man kann dabei auch von einem "Fingerabdruck" eines für die Meßanordnung und die Fehlstelle spezifischen Auftretens von Teilentladungen sprechen [3]. Zur Mustererkennung für eine bestimmte Anordnung ist es sinnvoll bzw. notwendig eine TE-Messung und -Aufzeichnung über einen geeignet längeren Zeitraum ununterbrochen vorzunehmen [3...7], um auch stochastisch und seltener auftretende Impulse zu erfassen und statistisch auszuwerten. So kann zur Bestimmung eines TE-Musters ein Betrachtungszeitraum von 100, 1000 oder noch mehr Netzperioden erforderlich sein. Fast immer wird dabei die Verteilung der Ladungswerte (Anzahl, Amplitude) über alle betrachteten Perioden hinweg auf einem Phasenbild ( $0-360^\circ$ ) dargestellt ("Phasenwinkelhistogramm"). Das so erhaltene 3-dimensionale Bild mit dessen Achsen "Phase", "Ladungsamplitude" und "Anzahl" kann zur Mustererkennung herangezogen werden. Die Werte für die Dimensionen "Phase" und "Ladung" können dabei beliebig fein aufgelöst sein. Expertensysteme sind dann in der Lage aus diesen Parametern eine Musterklassifikation durchzuführen [7].

Eine neue moderne Methode einer solchen Musterklassifikation besteht in der gezielten Anwendung neuronaler Netzwerke, die als Software z.B. auf einem PC implementiert werden können [z.B. 10]. Ein solches programmiertes Netz ist in der Lage, auf angebotene, theoretisch beliebig viele, verschiedene Muster eine Klassifikation in die jeweilig bekannte Kategorie vorzunehmen. Voraussetzung dabei ist, daß das neuronale Netz auf die jeweiligen Muster "trainiert" wurde, d.h. daß möglichst typische Repräsentanten der einzelnen Muster dem neuronalen Netz bekannt gemacht wurden. Die große Stärke eines neuronalen Netzes liegt dann in der Fähigkeit zu verallgemeinern, d.h. daß es auch ungelernte Muster schnell und richtig zu klassifizieren vermag.

## 2. Neuronale Netze

### 2.1 Grundlagen

Der Grundbaustein eines künstlichen neuronalen Netzes geht aus einem Neuronenmodell, dem 'Perceptron', wie es in Bild 1 zu sehen ist, hervor. Wie im Gehirn eines Menschen oder Tieres, wo sehr viele Neuronen über noch mehr Axonen bzw. Dendriten, den Leitungen zum Informationstransport, miteinander verbunden sind, hat auch das künstliche Perceptron die Aufgabe mehrere Eingangsinformationen verschieden gewichtet aufzusummieren um dann gemäß einer Aktivierungsfunktion einen Ausgangswert, d.h. eine Entscheidung, auszugeben [z.B. 8, 9]. Dieser Ausgangswert kann dann eine von vielen Eingangsinformationen weiterer solcher Neuronen darstellen. Auf diese Weise läßt sich ein Netz solcher Neuronen beliebig komplizierter Struktur aufbauen.

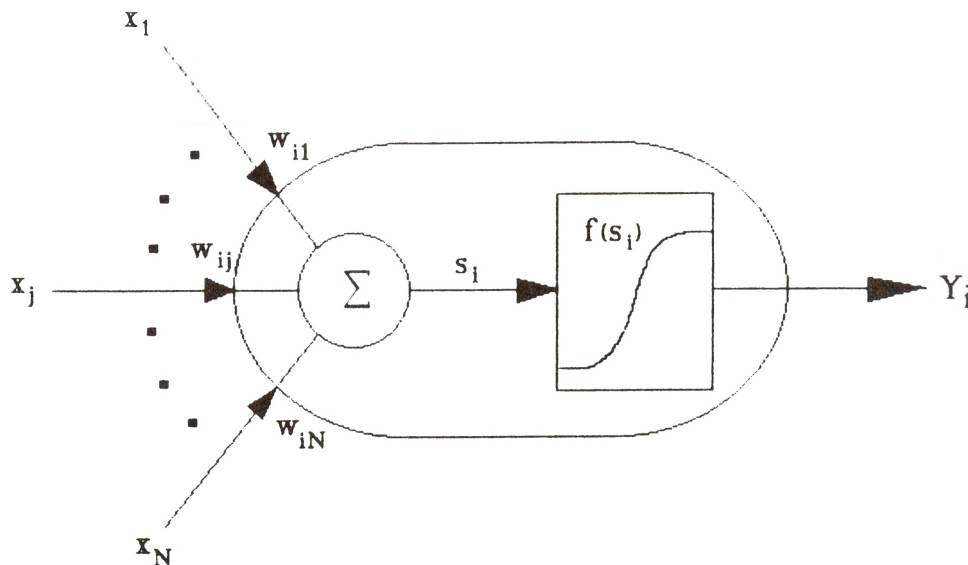


Bild 1: Häufig verwendete Form eines Modell-Neurons

Zur Kalkulation der Ausgangswerte  $y_i$  gilt dann:

$$s_i = \sum_j w_{ij} x_j; \quad (1 \text{ a})$$

$$y_i = f(s_i); \quad (1 \text{ b})$$

Die 'Aktivierungsfunktion'  $f(s_i)$  hat einen maßgebenden Einfluß auf das Verhalten des gesamten Netzes. Die Gewichtsparameter\*  $w_{ij}$  können beliebig unterschiedlich sein. Es ist einsichtig, daß durch die beliebige Wahl der Gewichtsparameter und der Aktivierungsfunktion  $f$  eine große Breite im Reaktionsverhalten des Perceptrons

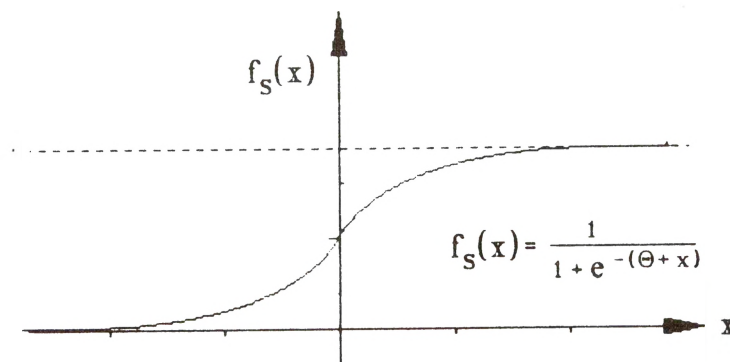
\* Die "Gewichtsparameter" werden in der Literatur, in Anlehnung an die Biologie, häufig auch als "synaptische Koeffizienten" bezeichnet.

erzielt werden kann. Vorzugsweise wird oft eine stark nichtlineare Aktivierungsfunktion mit nichtlinearem Anteil gegen "voll aktiv = 1" und "nicht aktiv = 0" als Entscheidungscharakteristik gewählt. Dies hat seinen Grund darin, daß oft vom gesamten Netzwerk, mit vielen solcher Neuronen eine extrem "nichtlineare Entscheidung", wie sie beispielsweise bei boole'schen Funktionen der Fall ist, abverlangt wird [8,10].

Eine derart nichtlineare Charakteristik wird z.B. erreicht durch die 'Sigmoid-Funktion':

$$f(s_i) = (1 + \tanh(\theta + s_i))/2 = 1/(\exp(-(\theta + s_i)) + 1); \quad (2)$$

In Bild 2 ist diese Funktion skizziert und deutet das konvergente Verhalten für  $x \rightarrow +\infty$  und  $x \rightarrow -\infty$  an. Mit dem in Gleichung (2) angegebenen Offsetwert  $\theta$  (auch als "Schwellwert" bezeichnet) kann zusätzlich noch eine große Variationsmöglichkeit im Konvergenzverhalten der Funktion erreicht werden.



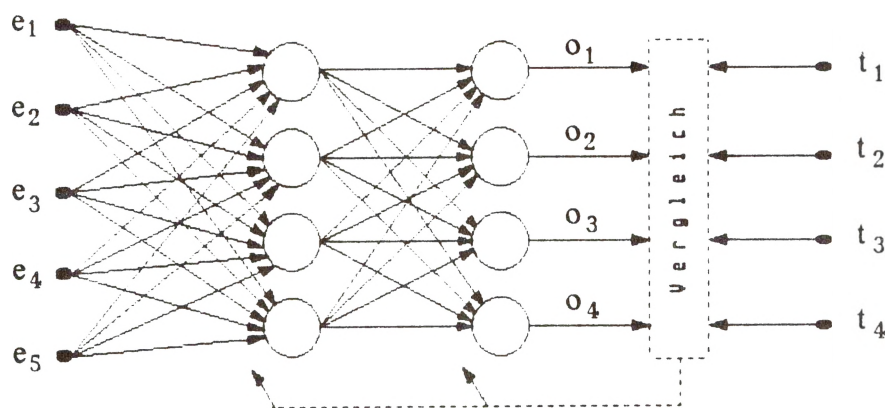
*Bild 2: Charakteristik der als Aktivierungsfunktion häufig verwendeten nichtlinearen 'Sigmoid-Funktion'.*

Ein neuronales Netz ist aus vielen einzelnen Bausteinen gemäß dem obigen Modellneuron aufgebaut. Es gibt beliebig viele Möglichkeiten, Verbindungen zwischen den Bausteinen eines Netzwerkes herzustellen. Ebenso sind die Verfahren, das Netzwerk ein bestimmtes Muster zu "lehren", d.h. es auf ein Muster zu trainieren, sehr unterschiedlich. Im folgenden wird eine von vielen möglichen Netzwerkstrukturen und ein spezielles, häufig angewandtes Lernverfahren vorgestellt.

## 2.2 Netzwerkstruktur und Lernregel

Bei der Anwendung neuronaler Netze sind sowohl der Netzwerkaufbau mit dessen Gewichtswerten, sowie die Art und Weise, wie das Netzwerk ein bestimmtes Muster erlernt (dazu gehört auch die Art der Aktivierungsfunktion), von entscheidender Bedeutung.

In der folgenden Beschreibung wird exemplarisch von nur einer (angewandten) Netzwerkategorie ausgegangen. Bei dieser Art von Netzwerken handelt es sich um sogenannte "feedforward"-Netze, bei denen eine strenge Informationsrichtung vom Eingang zum Ausgang hin, ohne irgendwelche Rückkopplungen, zu erkennen ist (siehe Bild 3).



*Bild 3: Feedforward-Netz mit zwei Neuronenschichten und dem Eingangsmuster  $e_i$ , dem Ausgangsmuster  $o_i$  und dem Sollmuster  $t_i$ . In der Trainingsphase wird das Ausgangsmuster dem Sollmuster angepaßt.*

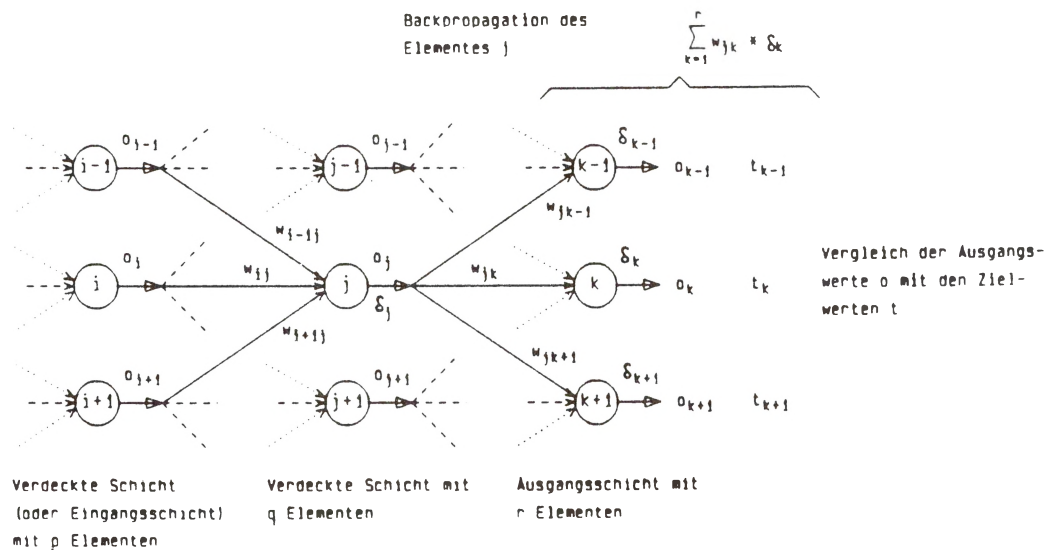
Für die Anwendung von neuronalen Netzen wird zwischen zwei Betriebsarten unterschieden:

1. **Lernen** eines bestimmten, angebotenen Datenmusters
2. **Erkennen** eines ungelernten, neuen Datenmusters durch Einordnen in eine vom Netzwerk gelernte Musterkategorie.

Bei der Betriebsart "Lernen", auch als "Training" bezeichnet, wird dem Netzwerk ein Eingabemuster und ein dazugehöriges Soll-Ausgabemuster angeboten. Die



Gewichtsfaktoren und Parameter der Aktivierungsfunktionen der Neuronen sind zunächst beliebig. Dem Netzwerk wird nur das Eingabemuster eingespeist und ein Ausgabemuster, als Ausgangsentscheidung, erzeugt. Durch Vergleich des Sollmusters mit dem tatsächlich erzeugten Ausgabemuster (=Istmuster) können die Gewichtsfaktoren der Neuronen korrigiert werden. Es gibt nun viele Möglichkeiten auf welche Art und Weise diese Korrektur vorgenommen werden kann [z.B. 8, 9, 10]. Eine bisher häufig und erfolgreich angewandte Methode eines Netzwerktrainings stellt das **"Error-Backpropagation"**-Verfahren für mehrschichtige Netze dar. Das Prinzip dieses Algorithmus' beruht darauf, daß der Entscheidungsfehler (die Differenz zwischen Soll- und Istmuster) als Grundlage zur Veränderung der Gewichtsfaktoren herangezogen wird. Dabei erfolgt die Korrektur der Faktoren vom Netzausgang zum Netzeingang hin so, daß die Faktoren der Ausgangsschicht zuerst und diejenigen der Eingangsschicht zuletzt abgeändert werden. Für die Kalkulation der Gewichtsfaktoren oder Offsetwerte der Aktivierungsfunktion nach der Backpropagation-Regel wird ein mehrschichtiges feedforward-Netz zugrundegelegt (siehe Bild 4). Bei der Berechnung der Gewichtsfaktoren muß zwischen dem Informationstransport der Neuronen der letzten verdeckten Schicht (= letzte "Zwischenschicht") zur Ausgangsschicht und den Neuronen einer verdeckten Schicht (oder Eingangsschicht) zur nächstfolgenden verdeckten Schicht unterschieden werden.



**Bild 4:** Mehrschichtiges Netz zur Veranschaulichung des "Error-Backpropagation"-Verfahrens.

Für die Bestimmung des Netz-Ausgabefehlers zur Verwendung im Backpropagation-Algorithmus wird das mittlere Fehlerquadrat aus den Ausgangswerten aller Neuronen der Ausgabeschicht herangezogen. Wenn  $t_k$  die Werte eines Sollmusters,  $o_k$  die des Ausgabemusters darstellt und der Parameter  $\mu$  sich über alle angebotenen Muster erstreckt, so berechnet sich der Entscheidungsfehler wie folgt:

$$E_{\mu} = 1/2 \sum_{\mu} \sum_k [t_k^{\mu} - o_k^{\mu}]^2 ; \quad (3)$$

$f(h_k^{\mu})$  ist die Aktivierungsfunktion der Ausgangsneuronen. Für die Ausgabewerte der Ausgangsneuronen gilt:

$$o_k = f(h_k) \quad \text{mit} \quad h_k = \sum_j w_{jk} o_j - \theta_k ; \quad (4)$$

und für Neuronen einer verdeckten Schicht:

$$o_j = f(h_j) \quad \text{mit} \quad h_j = \sum_i w_{ij} o_i - \theta_j ; \quad (5)$$

Die Beiwerte  $\theta_k$  bzw.  $\theta_j$  sind die Reaktionscharakteristik der Aktivierungsfunktion bestimmenden Schwellwerte, und geben den Beginn der Aktivität eines Neurons an. Für die Sigmoid-Funktion entspricht dies dem Offsetwert  $\theta$ .

Der obige Fehlerwert  $E_{\mu}$  sollte idealerweise ein globales Minimum besitzen. Ein solches Minimum wird erreicht durch Berechnung des Gradienten von  $E_{\mu}$ :

$$\begin{aligned} \delta w_{jk} &= - \epsilon \frac{\partial E}{\partial \theta_k} ; \\ &= - \epsilon \sum_{\mu} [t_k^{\mu} - f(h_k^{\mu})] f'(h_k^{\mu}) \frac{\partial h_k^{\mu}}{\partial w_{jk}} ; \\ &= \epsilon \sum_{\mu} \Delta_k^{\mu} o_j^{\mu} ; \end{aligned} \quad (6)$$

Entsprechend gelte für die Änderung der Offsetwerte  $\theta_k$  der Ausgabeschicht:

$$\begin{aligned}
 \delta\theta_k &= -\epsilon \frac{\partial E}{\partial \theta_k} ; \\
 &= -\epsilon \sum_{\mu} [t_k^{\mu} - f(h_k^{\mu})] f'(h_k^{\mu}) \frac{\partial h_k^{\mu}}{\partial \theta_k} ; \\
 &= \epsilon \sum_{\mu} \Delta_k^{\mu} ; \tag{7}
 \end{aligned}$$

Analog können die Änderungen der Gewichtungsfaktoren für die Neuronen der Zwischenschichten berechnet werden:

$$\begin{aligned}
 \delta w_{ij} &= -\epsilon \sum_{\mu,k} \Delta_k^{\mu} w_{ij} f'(h_j^{\mu}) \frac{\partial h_k^{\mu}}{\partial w_{jk}} ; \\
 &= \epsilon \sum_{\mu,k} \Delta_j^{\mu} o_i^{\mu} ; \tag{8}
 \end{aligned}$$

Die Änderung der Offsetwerte  $\theta_j$  der Zwischenschichtneuronen:

$$\begin{aligned}
 \delta\theta_j &= -\epsilon \sum_{\mu,k} \Delta_k^{\mu} w_{ij} f'(h_j^{\mu}) \frac{\partial h_k^{\mu}}{\partial \theta_j} ; \\
 &= \epsilon \sum_{\mu,k} \Delta_j^{\mu} ; \tag{9}
 \end{aligned}$$

Der Parameter  $\epsilon$  ist die Differentialschrittweite für die numerische Gradientenbildung und sollte ausreichend klein bleiben.

In [8] wird zusätzlich noch vorgeschlagen, daß bei jedem Schritt des Lernprozesses



jeweils noch die Vergangenheit mitberücksichtigt werden soll. Dies fließt für die Ausgangsschicht mit dem Parameter  $\alpha$  in die Gewichtungsfaktoränderung mit ein:

$$\delta w_{jk}^{(n)} = \epsilon \sum_{\mu} \Delta_k^{\mu} o_j^{\mu} - \alpha \delta w_{jk}^{(n-1)} ; \quad (10)$$

Die Variable  $n$  kennzeichnet dabei den  $n$ -ten Lernschritt.

Der Parameter  $\alpha$  dämpft ein eventuell während des Lernprozesses provoziertes Oszillieren der Fehlerwerte und hat deshalb einen stabilisierenden Effekt. Eine ideale Kombination der Parameter  $\epsilon$  und  $\alpha$  kann nicht definiert werden. Deren Wahl hängt von der Komplexität der Muster, dem Netzaufbau und der Anzahl verschiedener Muster ab. Nur mehrere Lerndurchläufe des Netzes geben Aufschluß über eine sinnvolle Wahl von  $\epsilon$  und  $\alpha$  für eine bestimmte Art von Mustern. Wird das Netz frisch auf einen Musterdatensatz trainiert, so sollten die Gewichtungsfaktoren günstigerweise per Zufallsgenerator eingestellt werden. Nach [8,9] darf die Gewichtungsfaktor-Matrix keine Symmetrie oder Gleichmäßigkeit der Werte aufweisen, denn sonst kann es sein, daß es nie zu einer Fehlerminimierung während des Lernprozesses kommt.

### 3. Anwendung auf die TE-Mustererkennung

Wie in Kapitel 1 erwähnt wurde, ist es möglich eine Mustererkennung von TE-Signalen mit neuronalen Netzwerken durchzuführen. Für die Mustererkennung sind dazu die genannten signifikanten charakteristischen Parameter entscheidend. Wie stark man die Eingangsinformationen für das Netz gewichten muß, wie fein aufgelöst die Parameter "Amplitude" und "Phase" zur lokalen Erkennung des Musters sein müssen und welcher optimale Netzaufbau und welche Netzparameter am geeignetsten sind, dies ist letztenendes noch zu klären.

In Kapitel 1 wurde angedeutet, daß es nötig sein kann, die TE-Erscheinung über einen längeren Zeitraum zu beobachten, um mit Hilfe einer statistischen Auswertung des abgetasteten und aufgezeichneten Signals eine Musterklassifikation durchzuführen. Auch ein neuronales Netz macht auf indirekte Weise eine solche statistische Auswertung indem vorher einfach lokal an jeder Stelle der Netzperiode

über den Beobachtungszeitraum die Signal-Abtastwerte in Form eines Amplitudenhistogramms für jede vorgegebene Amplitudenstufe aufsummiert und danach z.B. auf die Gesamtzahl der Abtastpunkte normiert werden. Diese Werte der einzelnen über die gesamte Phase verteilten Histogramme werden dann einfach einem neuronalen Netz als Eingangsmuster zugeführt.

Das Ausgabemuster wird durch die Ausgangswerte eines Satzes von genau *so* vielen Ausgabeneuronen beschrieben, wie verschiedene Muster zu unterscheiden sind. Eine klare Entscheidung für ein Muster würde dann bedeuten, daß das entsprechende Ausgabeneuron ideal den Wert "1.0" ausgibt, während dann alle anderen Ausgabezellen "0.0" ausgeben sollten. Ein Muster gilt jedoch generell als erkannt, wenn die ihr zugeordnete Zelle die größte Aktivität d.h. den größten Wert unter den Ausgabezellen annimmt.

Um komplizierte Muster wie sie bei Teilentladungsaufzeichnungen erzeugt werden in einem neuronalen Netz zu verarbeiten, sollten folgende Punkte beachtet werden:

- 1) Die Netz-Eingangsdaten müssen einen vollständigen Satz aller signifikanten Parameter umfassen.
- 2) Während des Lernprozesses müssen in einem Iterationsschritt dem Netz sämtliche typischen Vertreter aller zu lernenden Musterklassen angeboten werden.
- 3) Das Netz sollte in der Lernphase so lange auf die angebotenen Muster trainiert werden, d.h. es müssen so viele Lernschritte durchlaufen werden, bis eine eindeutige Klassifikation der gelernten Muster möglich ist.

#### **4. Anwendungsbeispiel und erste Ergebnisse**

Im folgenden wird gezeigt, wie erste Versuche unternommen wurden, beispielsweise zwei verschiedene TE-Impulsmuster zu identifizieren. Zur Unterscheidung von zwei Impulsmustern sind drei Ausgangsneuronen nötig, da für Stellen des Signals, wo keine Impulse auftreten eine "nicht definiert"-Entscheidung als "drittes" Muster festgelegt werden sollte.

##### **4.1 Aufzeichnungssystem und Meßsignale**

Zur Generierung der beiden Impulsmuster wurden zwei Meßreihen durchgeführt:

- 1) Messung nur von inneren Teilentladungen eines kleinen kunstharzisierten Hochspannungstrafos  $U_{\text{primN}} = 220 \text{ V}$  und  $U_{\text{sekN}} = 7 \text{ kV}$  unter Nennbedingungen. Messung bei ca. 7 kV. Entladungserscheinungen auf positiver und negativer Halbwelle.
- 2) Messung nur von äußeren Entladungen: langer dünner Draht auf Hochspannungspotential und Spitze auf Erdpotential. Messung bei ca. 7 kV. Entladungserscheinungen auf positiver und negativer Halbwelle. Meßaufbau sonst wie bei 1).

Beide Meßreihen wurden über einen Koppelkondensator von  $C_k = 2 \text{ nF}$  und einen HAEFELY-Ankoppelvierpol Typ 568 aufgenommen. Die Hochspannungsanlage besteht aus einem 100 kV-Transformator und dazugehörigen Stell- und Meßeinrichtungen. Das Aufzeichnungssystem bestand aus einem einstellbaren Vorverstärker und einem Vorfilter mit einer Grenzfrequenz von ca. 300 kHz. Die Abtastrate betrug 1 MHz. Die Aufzeichnungen wurden jeweils auf die Phase der anliegenden Hochspannung synchronisiert und mit einem PC (AT 286) aufgezeichnet. In den Speicher des PC's konnten 10 Netzperioden hintereinander aufgezeichnet werden. Die Auswertung der Signale erfolgte ebenfalls auf einem PC.

##### **4.2 Die für die Simulation verwendeten Parameter**

Von den unter 4.1 erwähnten Meßreihen wurden jeweils 3 Messungen von jedem der beiden TE-Muster vorgenommen (siehe Bild 5 und 6). Jeweils 2 von diesen TE-

Aufzeichnungen wurden dem neuronalen Netz als Muster eintrainiert. Für die 3. Meßaufzeichnung jeder Musterklasse wurde nach dem Lernprozeß eine Erkennung durchgeführt.

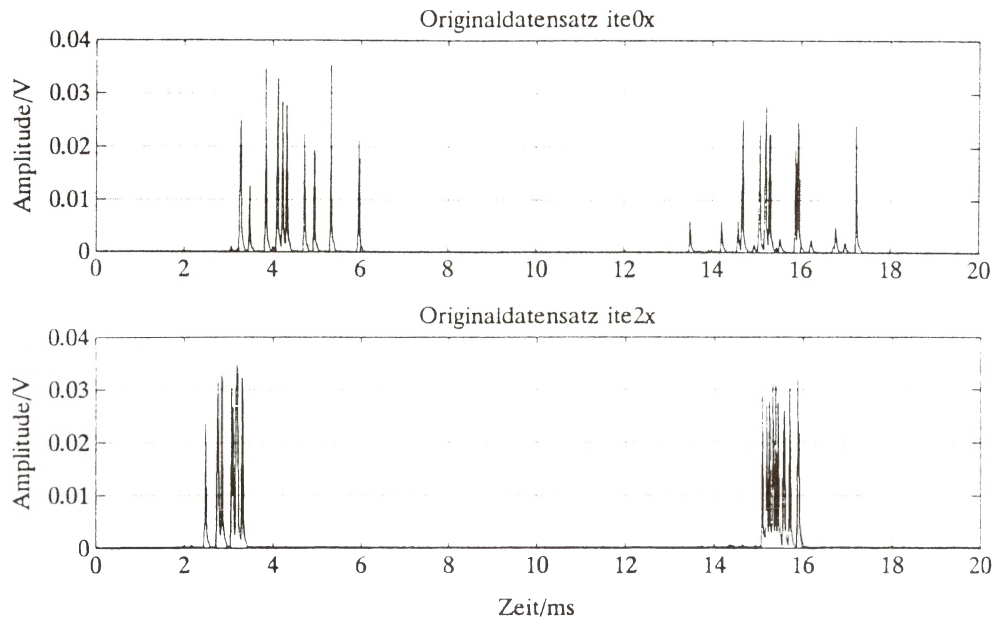


Bild 5: Dem Netzwerk angebotene Muster der Klasse "Muster 1 = innere TE" <sup>\*</sup>.

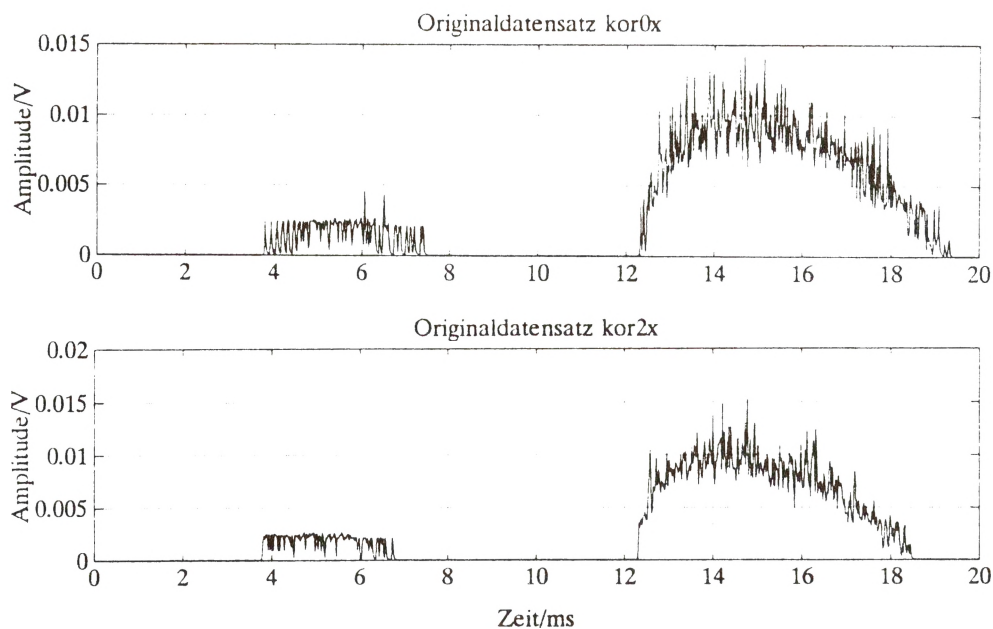


Bild 6: Dem Netzwerk angebotene Muster der Klasse "Muster 2 = äußere TE" <sup>\*</sup>.

<sup>\*</sup>

Der Bereich von 0 bis 10 ms des dargestellten Signals entspricht dem Phasenbereich der positiven Sinus-Halbwellen von  $0^\circ$  bis  $180^\circ$ , der Bereich von 10 bis 20 ms dem der negativen Halbwellen von  $180^\circ$  bis  $360^\circ$ .

Zur Generierung der Eingangsdaten wurde zunächst ein Amplitudenhistogramm über alle Abtastwerte erstellt. Dabei ging man vom maximal auftretenden Signalpegel aus und unterteilte dessen Amplitude linear in  $10^*$  Amplitudenstufen. Ein komplettes Muster besteht aus  $256^*$  Phasenabschnitten, deshalb wurde für alle 256 Stellen des kompletten Phasenbereichs ( $0 - 360^\circ$ ) jeweils ein entsprechendes Histogramm erstellt. Zusätzlich wurden noch Gesamthistogramme, die das Muster in positive und negative Halbwelle einteilten, über alle Abtastwerte ermittelt. Die Netz-Eingangsdaten für ein Muster bildeten nun die 10 (normierten) Histogrammwerte der 256 Phasenbereiche. Als zusätzlicher Eingangswert wurde noch die Nummer des Phasenbereichs gewählt. Als 12. Eingangswert eines Muster-Phasenbereichs zog man die Gesamtzahl der Impulse für die positive Halbwelle, betreffend den Phasenbereich von 1 bis 128 und die Gesamtzahl der Impulse für die negative Halbwelle für die Phasenbereiche 129 bis 256 heran. Die "Gesamtzahl" der Impulse wurde ermittelt indem bei den Gesamthistogrammen (für die positive bzw. negative Halbwelle) alle Abtastpunkte, deren Amplitude größer war als die niedrigste Amplitudenstufe, aufsummiert wurden. Sämtliche 12 Eingangswerte des neuronalen Netzes wurden normiert.

Das Bild 7 zeigt ein für die Musterklasse 1 (= "innere TE") charakteristisches Gesamthistogramm, getrennt in positive und negative Halbwelle. In Bild 8 hingegen werden die Histogramme für die Musterklasse 2 (= "äußere TE") gezeigt. Die Histogramme erstrecken sich jeweils über alle 10 gemessenen Netzperioden.

\*

Die Anzahl von 256 Phasenabschnitten hat eher historische Gründe: der Umfang eines auf 256 Werte pro Netzperiode reduzierten Datensatzes wurde für das TE-Meßgerät nach dem Verfahren von [1] verwendet und aus Gründen der Kompatibilität hier übernommen. Die Amplitudenstufenzahl von 10 wurde bezüglich des Grundstörspegels als sinnvoll angesehen.

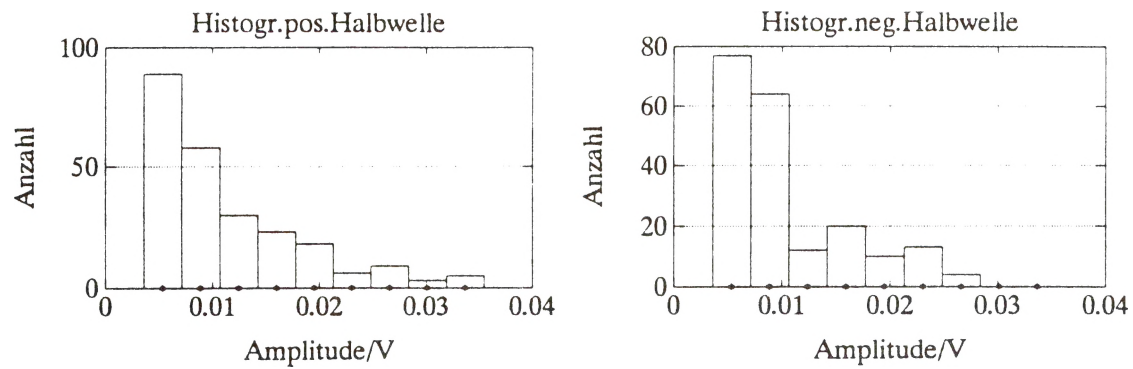


Bild 7: Typisches Gesamthistogramm\* des dem Netzwerk angebotenen Musters der Klasse "Muster 1" (= "innere TE").

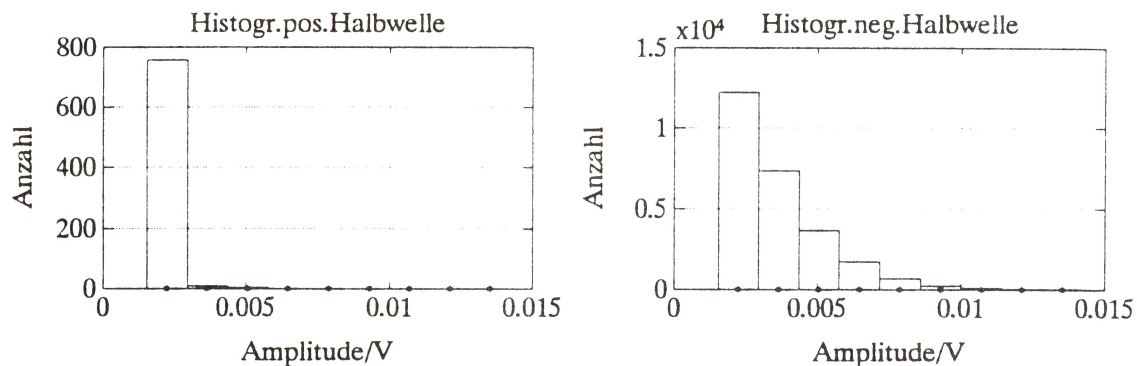


Bild 8: Typisches Gesamthistogramm\* des dem Netzwerk angebotenen Musters der Klasse "Muster 2" (= "äußere TE").

Für das neuronale Netzwerke wurde folgende Struktur gewählt:

- Feedforward-Netz mit einer Eingangsschicht, einer Zwischenschicht und einer Ausgabeschicht.
- Anzahl der Eingangswerte: 12
- Anzahl der Neuronen der Eingangsschicht: 10
- Anzahl der Neuronen der Zwischenschicht: 10
- Anzahl der Neuronen der Ausgabeschicht: 3

\*

Bei den dargestellten Histogrammen wurde der Wert der untersten Amplitudenstufe einfach zu Null gesetzt, da für diesen Bereich die Anzahl der Abtastwerte stets zu groß ist und die Skalierung für die aussagekräftigen oberen Stufen ungünstig wäre.



- Parameter  $\alpha = 0.7$  ;
- Parameter  $\epsilon = 0.01$  ;

Die Tabelle 1 zeigt die Zuordnung und Wertigkeit der Ausgangsneuronen betreffend die beiden Musterklassen.

idealer Ausgangswert für Ausgangsneuron Nr.:			zugehörige Musterklasse
1	2	3	
1.0	0.0	0.0	Muster 1 = "innere TE"
0.0	0.0	1.0	Muster 2 = "äußere TE"
0.0	1.0	0.0	Muster 3 = "nicht definiert"

*Tabelle 1: Ideale Ausgangsentscheidung bei der Erkennung eines angebotenen Musters.*

### 4.3 Simulationsergebnisse

Mit den in 4.2 vorgestellten Netzparametern wurde für die beiden Musterklassen (mit jeweils 2 Mustern) ein Netztrainingslauf vorgenommen. Bild 9 zeigt den Gesamtverlauf des Lernfehlers. Für einen Iterationschritt wurden alle 4 Musterperioden hintereinander und von jeder Musterperiode jeder einzelne Phasenpunkt zweimal als Eingangsmuster angeboten. Wie in Kapitel 3 erwähnt, gilt bei der Ausgabeentscheidung des Netzes ein Muster dann als erkannt, wenn das dem Muster zugehörige Ausgabeneuron den größten Wert ausgibt. Ein Entscheidungsfehler liegt vor, wenn das Ausgabeneuron mit dem größten Wert nicht mit dem Ausgabeneuron des Sollmusters an der entsprechenden Stelle übereinstimmt. In Bild 10 ist der Lernerfolg nach 100 Iterationsschritten gezeigt. Die Entscheidungsfehler sind durch ein "x" auf der Position "1.0" markiert. Der Fehler  $E$  für die Ausgabeentscheidung wurde durch Bildung des mittleren Fehlerquadrates über alle 3 Ausgabewerte ermittelt.

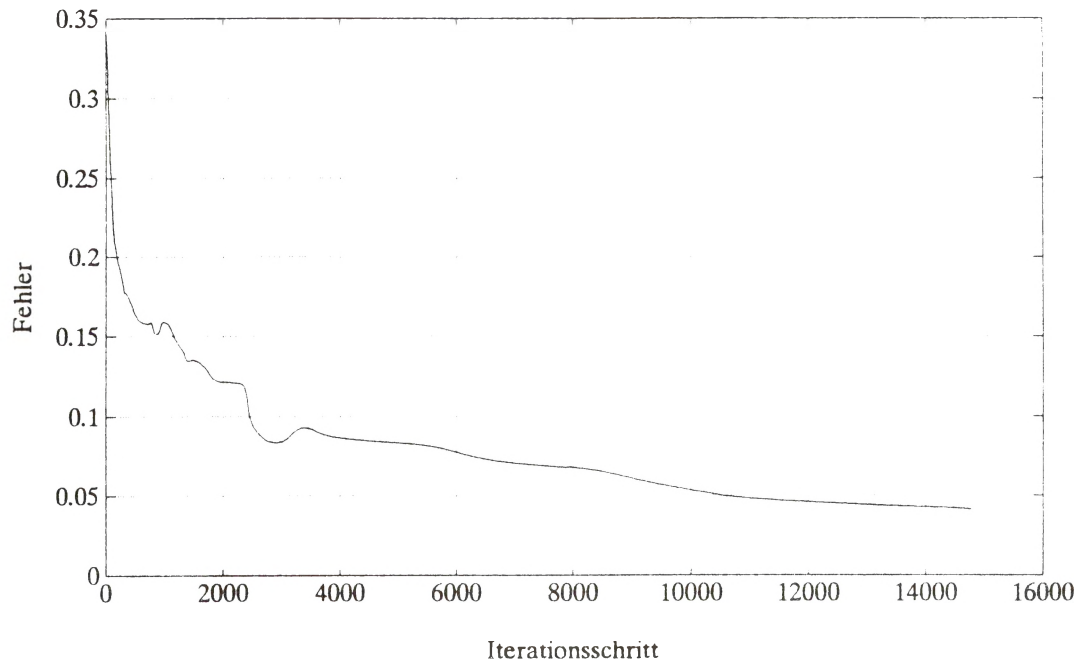


Bild 9: Fehlerrate  $E$  für den Lernprozeß, der nach 15000 Iterationen abgeschlossen war.

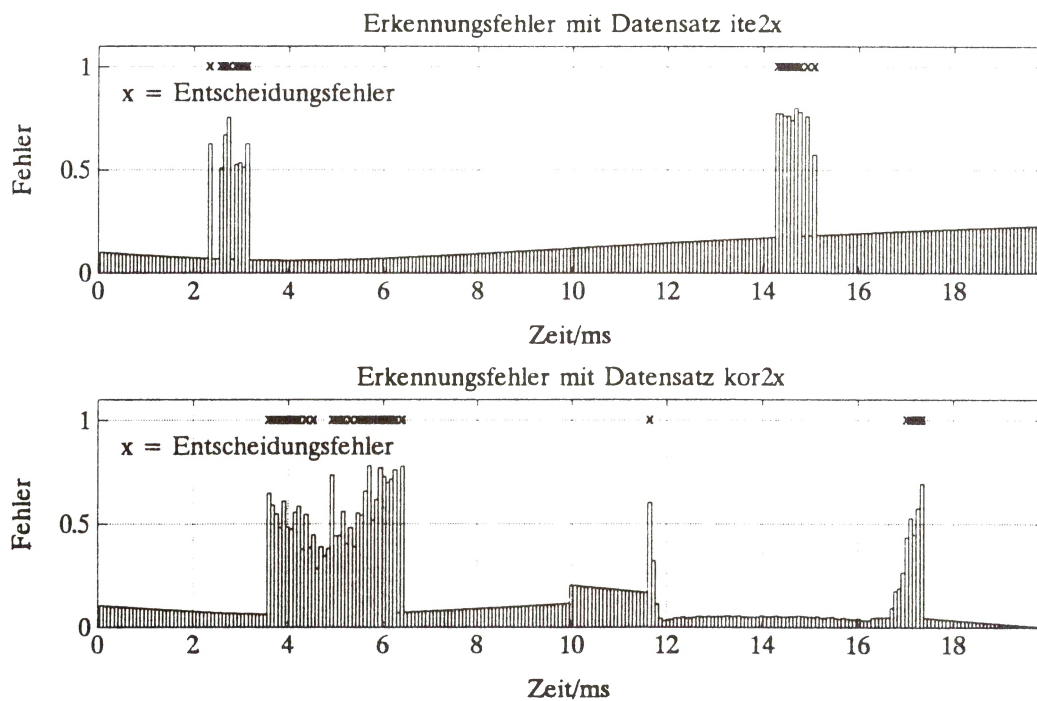
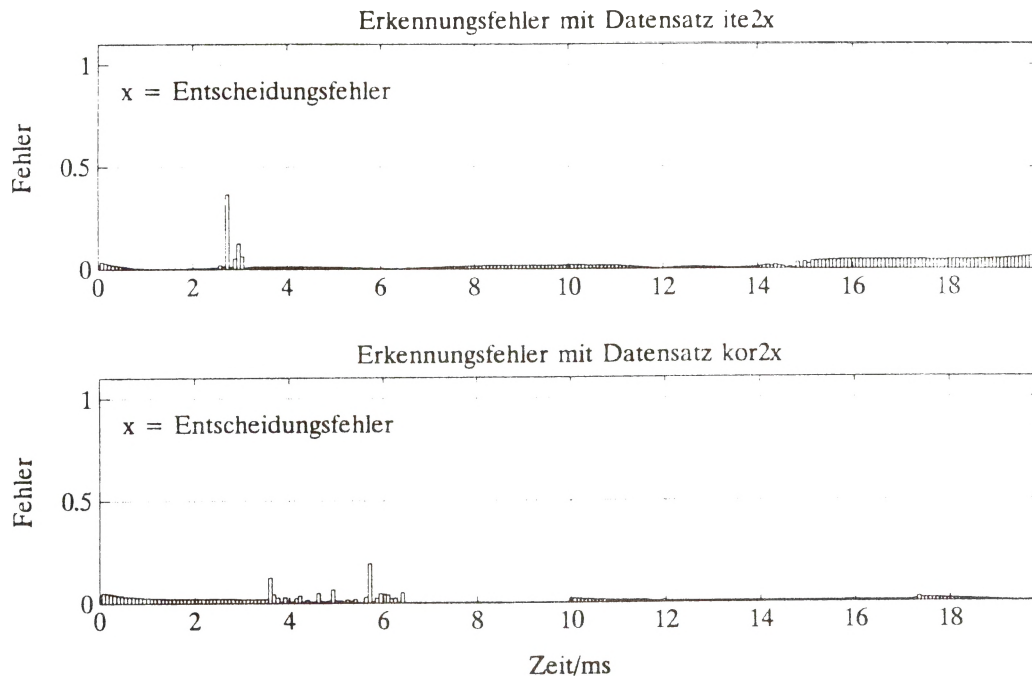


Bild 10: Lernerfolg nach 100 Iterationen. Oben: Erkennungsfehler für ein Muster der Musterklasse 1. Unten: Erkennungsfehler für Musterklasse 2. Die Muster sind erst schlecht "antrainiert".

In Bild 11 ist der Lernerfolg derselben Muster wie in Bild 10 nach 15000 Iterationen gezeigt. Man sieht deutlich, daß beide Muster an jeder Stelle der Phase richtig erkannt werden.



*Bild 11: Lernerfolg nach 15000 Iterationen. Oben: Erkennungsfehler für ein Muster der Musterklasse 1. Unten Erkennungsfehler für ein Muster der Musterklasse 2. Die Muster sind gut "antrainiert" und werden vollständig richtig erkannt.*

Als nächstes wurde jeweils ein ungelernes Muster aus beiden Musterklassen dem trainierten Netz angeboten. In Bild 12 ist der zu klassifizierende Datensatz der Musterklasse 1 mit den entsprechenden Erkennungsfehlern gezeigt. Das Bild 13 zeigt den Datensatz der Musterklasse 2. Es ist deutlich zu sehen, daß nur zwei Stellen des Musters in Bild 12 falsch erkannt wurden. Ansonsten werden beide Muster an jeder Stelle richtig klassifiziert. Damit ist gezeigt, daß ein neuronales Netz durchaus auch unbekannte jedoch ähnliche Muster wiedererkennt.

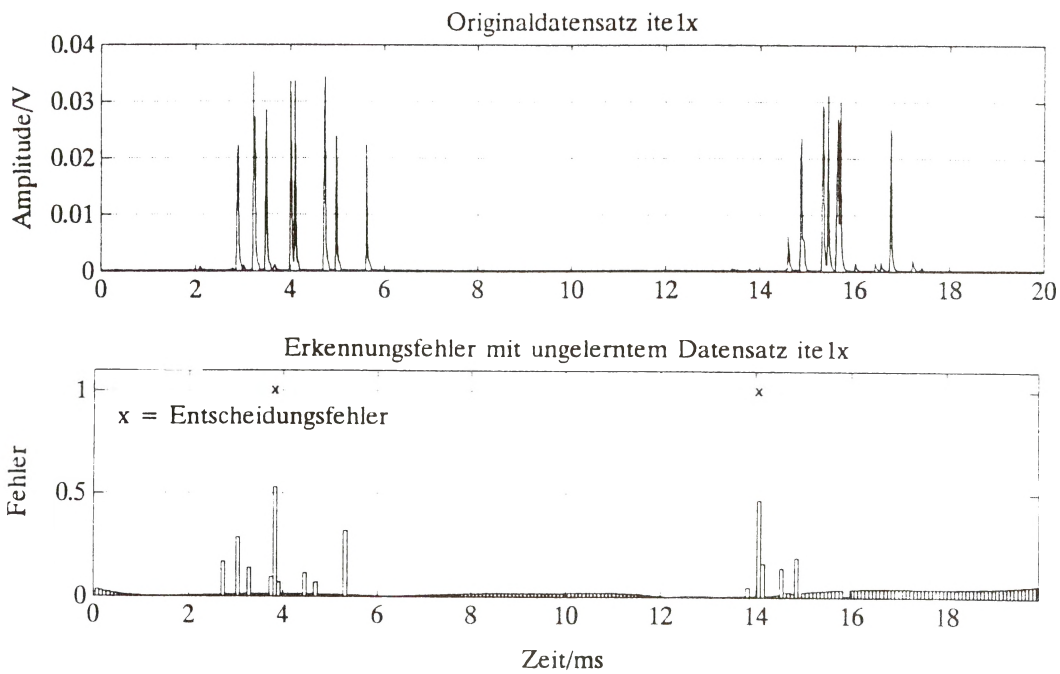


Bild 12: Oben: "ungelernter" Datensatz der Musterklasse 1. Unten: Erkennungsfehler des angebotenen Musters.

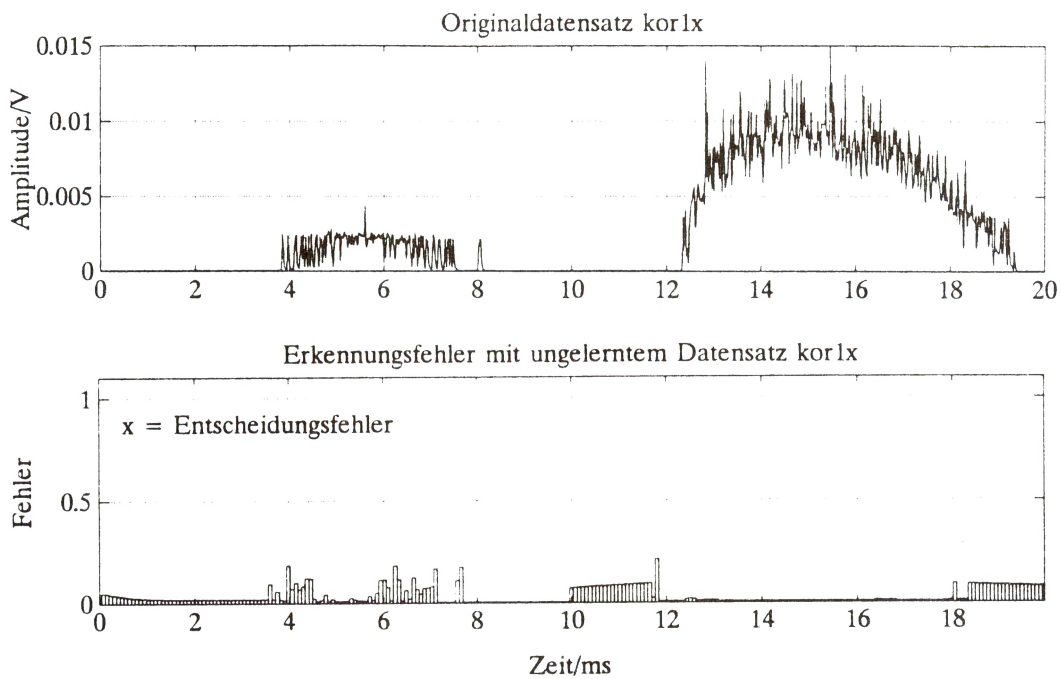


Bild 13: Oben: "ungelernter" Datensatz der Musterklasse 2. Unten: Erkennungsfehler des angebotenen Musters.

## 5. Zusammenfassung und Ausblick

Erste Simulationsergebnisse zeigen, daß es prinzipiell möglich ist, komplizierte TE-Muster mit Hilfe eines neuronalen Netzes zu erkennen. Als nachteilig beim Lernprozeß hat sich die lange Rechenzeit von ca. 20 Sekunden pro Iterationsschritt über alle 4 angebotenen Muster erwiesen (diese Angabe bezieht sich auf eine 386-AT-Maschine mit 25 MHz und Coprozessor). Mehrere Versuche müssen noch gemacht werden um ein Optimum bezüglich Netzstruktur, Netzparameter und der Vorverarbeitung der Eingangsdaten zu erreichen. Auch was die Lernalgorithmen des Netzwerkes betrifft, sind noch viele Möglichkeiten offen. Denkbar wäre z.B. auch der Einsatz von "selbstorganisierenden Netzen" [9, 10, 11], die sich in ihrem Aufbau und bezüglich ihrer Parameter der Art des Musters selbständig anpassen. Ist die Lernphase des neuronalen Netzwerkes einmal abgeschlossen, können die gelernten Netzparameter direkt auf eine Hardwareschaltung übertragen werden (entsprechende "Neuro-Chips" existieren bereits). Auf diese Weise wird aufgrund der hardwarebedingten kurzen Datendurchlaufzeit eine On-line-Verarbeitung der Meßdaten möglich.

Die Verknüpfung der bisherigen Erkenntnisse aus der TE-Meßtechnik und TE-Mustererkennung kann zu der Entwicklung eines Meßsystems führen, mit dem es z.B. möglich wird, durch Mehrstellen-Messung in einer Anlage, das TE-Verhalten des zu überwachenden Betriebsmittels durch Vergleich der errechneten Muster richtig zu interpretieren. Vorrangiges Ziel bleibt zunächst die Erkennung und Unterdrückung der bei der On-line-Messung störenden impulsförmigen Korona-Entladungen z.B. mittels einer Mustererkennung. Erst danach kann eine eindeutige Identifikation der erfaßten TE möglich werden.

## **Schrifttum:**

### **TE-Diagnostik:**

- [1] G. König, K.Feser:  
A new Digital Filter to Reduce Periodical Noise in Partial Discharge Measurements. Sixth International Symposium on High Voltage Engineering, Paper No. 43.10, August 1989.
- [2] G. König, U. Köpf, K. Feser:  
Methods to Improve On-site Partial Discharge Measuring Instruments. Seventh International Symposium on High Voltage Engineering, August 1991.
- [3] E. Gulski, F. H. Kreuger:  
Computer-aided Recognition of Discharge Patterns. Seventh International Symposium on High Voltage Engineering, August 1991.
- [4] CIGRE Extrait d'electra No 11:  
Recognition of Partial Discharges, CIGRE Working Group 21.03 Study Committee No 21 (High Voltage Cables).
- [5] B. Fruth, J. Fuhr:  
Partial Discharge Pattern Recognition-Tool for Diagnosis and Monitoring of Aging, CIGRE, 1990.
- [6] J. Fuhr, M. Hässig, D. Königstein, M. Florkowski:  
PD-Systems for On-site Detection of Aging Processes in High Voltage Apparatus, ASEA Brown Boveri, Seventh International Symposium on High Voltage Engineering, August 1991.
- [7] H.-G. Kranz, R. Krump:  
Expertensystem für die TE-Messung an SF<sub>6</sub>-isolierten Anlagen, HAEFELY TEST SYSTEMS, Symposium 1990.



### **Neuronale Netze:**

- [8] David. E. Rumelhart, J. McClelland:  
Parallel Distributed Processing, Vol.1, The MIT Press, 1988.
- [9] M. Lemmon:  
Competitively Inhibited Neural Networks for Adaptive Parameter  
Estimation, Kluwer Academic Publishers, 1990.
- [10] B. Müller, J. Reinhardt:  
Neural Networks, Springer Verlag, 1990.
- [11] T. Gramß, H.-W. Strube:  
Entwicklung mehrschichtiger neuronaler Netzwerke zur Worterkennung  
und -reproduktion, Zeitschrift Informationstechnik it, 5/89, S 324. ff.