

<b>Zeitschrift:</b>	Bulletin des Schweizerischen Elektrotechnischen Vereins, des Verbandes Schweizerischer Elektrizitätsunternehmen = Bulletin de l'Association suisse des électriciens, de l'Association des entreprises électriques suisses
<b>Herausgeber:</b>	Schweizerischer Elektrotechnischer Verein ; Verband Schweizerischer Elektrizitätsunternehmen
<b>Band:</b>	82 (1991)
<b>Heft:</b>	13
<b>Artikel:</b>	Neuronale Netzwerke : Theorie und Praxis
<b>Autor:</b>	Bernasconi, Jakob
<b>DOI:</b>	<a href="https://doi.org/10.5169/seals-902976">https://doi.org/10.5169/seals-902976</a>

### Nutzungsbedingungen

Die ETH-Bibliothek ist die Anbieterin der digitalisierten Zeitschriften auf E-Periodica. Sie besitzt keine Urheberrechte an den Zeitschriften und ist nicht verantwortlich für deren Inhalte. Die Rechte liegen in der Regel bei den Herausgebern beziehungsweise den externen Rechteinhabern. Das Veröffentlichen von Bildern in Print- und Online-Publikationen sowie auf Social Media-Kanälen oder Webseiten ist nur mit vorheriger Genehmigung der Rechteinhaber erlaubt. [Mehr erfahren](#)

### Conditions d'utilisation

L'ETH Library est le fournisseur des revues numérisées. Elle ne détient aucun droit d'auteur sur les revues et n'est pas responsable de leur contenu. En règle générale, les droits sont détenus par les éditeurs ou les détenteurs de droits externes. La reproduction d'images dans des publications imprimées ou en ligne ainsi que sur des canaux de médias sociaux ou des sites web n'est autorisée qu'avec l'accord préalable des détenteurs des droits. [En savoir plus](#)

### Terms of use

The ETH Library is the provider of the digitised journals. It does not own any copyrights to the journals and is not responsible for their content. The rights usually lie with the publishers or the external rights holders. Publishing images in print and online publications, as well as on social media channels or websites, is only permitted with the prior consent of the rights holders. [Find out more](#)

**Download PDF:** 10.01.2026

**ETH-Bibliothek Zürich, E-Periodica, <https://www.e-periodica.ch>**

# Neuronale Netzwerke: Theorie und Praxis

Jakob Bernasconi

**Neuronale Netzwerke haben sich in verschiedenen Anwendungsbereichen als vielversprechende Alternative zu den traditionellen Lösungsansätzen entpuppt. Der vorliegende Beitrag befasst sich mit entsprechenden Untersuchungen, die zurzeit im Asea Brown Boveri Forschungszentrum in Baden durchgeführt werden. Ein konkretes Beispiel aus dem Gebiet der Materialoptimierung demonstriert das Potential von Neuronalen-Netzwerk-Methoden bei der Lösung von komplexen technischen Problemen.**

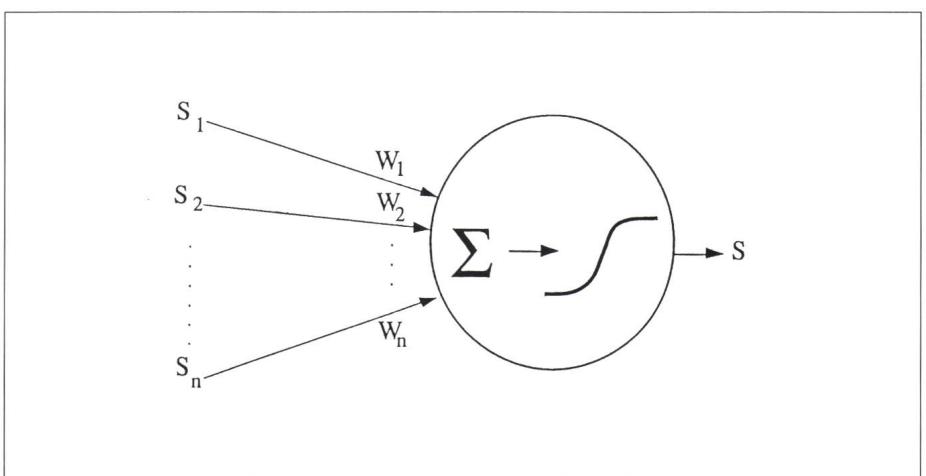
**Dans plusieurs domaines d'application, des réseaux neuronaux se sont trouvés être une alternative prometteuse aux approches traditionnelles. La présente contribution s'occupe de recherches qui à l'heure actuelle sont réalisées au centre de recherches de Asea Brown Boveri à Baden. Un exemple concret provenant du domaine de l'optimisation des matériaux démontre la puissance de méthodes qui se servent de réseaux neuronaux pour résoudre des problèmes techniques complexes.**

Künstliche Neuronale Netzwerke besitzen eine Reihe von Eigenschaften, die sie gegenüber konventionellen Methoden der Informationsverarbeitung auszeichnen [1;2;3]. Sie müssen nicht programmiert werden, sondern lernen ihre Aufgabe aus Beispielen. Sie verhalten sich sehr robust in bezug auf den Ausfall von Teilkomponenten und können auch fehlerhafte oder unvollständige Inputdaten verarbeiten. Ihre spezielle Struktur erlaubt zudem eine massiv parallele Informationsverarbeitung. Bei der Lösung von komplexen technischen Problemen können diese Eigenschaften entscheidende Vorteile bringen, insbesondere wenn es sich um Aufgaben handelt, die analytisch nur schwer oder überhaupt nicht erfasst werden können. Dazu kommt, dass die Entwicklungszeiten für Neuronale-Netzwerk-Systeme oft viel kürzer sind als für entsprechende konventionelle Lösungen.

Vielversprechende Anwendungsbereiche für Neuronale Netzwerke sind Muster- und Spracherkennung, Signalverarbeitung, maschinelles Lernen, Diagnose, Prognose, Optimie-

rung sowie Steuerung und Regelung. Konkrete Anwendungsbeispiele, die in den letzten Jahren realisiert wurden, reichen von der Erkennung handgeschriebener Buchstaben und Ziffern [4] über die Entdeckung von Sprengstoff im Flugzeuggepäck [5] bis zur Steuerung von Robotern [6]. Bei den meisten dieser Anwendungen handelt es sich um Forschungsprojekte oder Prototypsysteme, die aber eindrücklich das Potential dieser neuen Art der Informationsverarbeitung demonstrieren.

Im Asea Brown Boveri Forschungszentrum in Baden existiert seit etwa zwei Jahren ein interdisziplinäres Projekt, das sich einerseits mit grundlegenden Untersuchungen zum Lern- und Verallgemeinerungsverhalten Neuronaler Netzwerke und andererseits mit konkreten Anwendungen von Neuronalen-Netzwerk-Methoden befasst. Das gemeinsame Ziel dieser Studien ist die Entwicklung und Implementierung von effizienten Lernstrategien für verschiedene industriell interessante Anwendungsbereiche. Der vorliegende Beitrag gibt, nach ei-



**Bild 1** Schematische Darstellung eines künstlichen Neurons

## Adresse des Autors

Dr. Jakob Bernasconi, Asea Brown Boveri Forschungszentrum, 5405 Baden-Dättwil

ner kurzen Einführung in die Funktionsweise von Neuronalen Netzwerken, eine Übersicht über die einzelnen Themen dieses Forschungsprojekts. Ein konkretes Anwendungsbeispiel, das den Einsatz und die Leistungsfähigkeit von Neuronalen Netzwerken bei der Optimierung von Materialeigenschaften demonstriert, wird dann noch etwas ausführlicher vorgestellt.

## Neuronale Netzwerke

Die Bausteine von Neuronalen Netzwerken sind sogenannte künstliche Neuronen. Diese imitieren die Signalverarbeitung in biologischen Nervenzellen und können auf einem Computer simuliert oder in Hardware (z.B. auf einem Chip) realisiert werden. Ein künstliches Neuron (Bild 1) bildet eine gewichtete Summe der bei ihm ankommenden Signale  $S_1, S_2, \dots, S_n$  (Skalarprodukt des Eingangsvektors  $\mathbf{S}$  mit dem Gewichtsvektor  $\mathbf{W}$ ) und berechnet daraus mittels der Aktivierungsfunktion  $f(x)$  seinen Aktivitätszustand  $S$ , der dann als Signal an andere Neuronen weitergeleitet wird:

$$S = f\left(\sum_{j=1}^n W_j S_j\right) \quad (1)$$

Die Gewichte  $W_i$  können positiv oder negativ sein, und als Aktivierungsfunktion  $f$  wird meistens eine Schwellenfunktion gewählt, z.B.

$$f(x) = \text{sign}(x) = \begin{cases} +1 & \text{falls } x \geq 0 \\ -1 & \text{falls } x < 0 \end{cases} \quad (2a)$$

oder

$$f(x) = \frac{1-e^{-x}}{1+e^{-x}} = \text{Tgh}\left(\frac{x}{2}\right) \quad (2b)$$

Im ersten Fall kann das Neuron nur zwei Zustände annehmen,  $S = +1$  und  $S = -1$ , während bei der sogenannten sigmoiden Aktivierungsfunktion (2b) der Aktivitätszustand  $S$  kontinuierlich zwischen  $-1$  und  $+1$  variieren kann.

Ein künstliches Neuron ist also ein sehr primitives Element. Durch das Zusammenwirken vieler solcher Neuronen entstehen aber Netzwerkstrukturen, die fähig sind, sehr komplexe Aufgaben zu lösen. Die am häufigsten verwendete Netzwerkarchitektur ist das «Multilayer-Perceptron» [1;2;3] (Bild 2). Es besteht aus einer Schicht

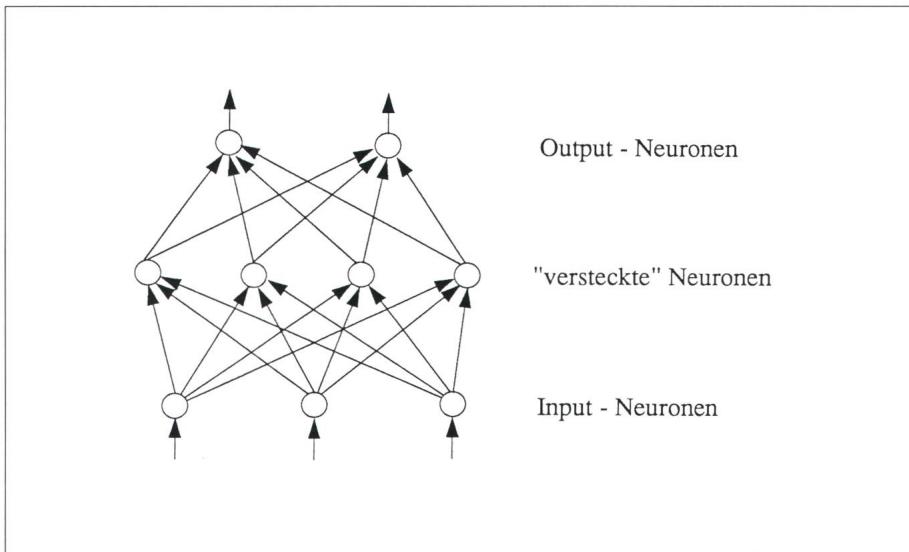


Bild 2 Ein Multilayer-Perceptron

Input-Neuronen, einer oder mehreren Schichten von «versteckten» Neuronen (hidden neurons) und einer Schicht Output-Neuronen. Nur benachbarte Schichten sind miteinander verknüpft, und die Signale werden nur in Vorrätsrichtung, von den Input- zu den Output-Neuronen, weitergeleitet.

Die Informationsverarbeitung in einem solchen Feedforward-Netzwerk besteht aus einem einzigen Durchgang, wobei die Neuronen Schicht für Schicht ihre Aktivitätszustände  $S_i$  berechnen. Die Berechnung erfolgt gemäß

$$S_i = f\left(\sum_j W_{ij} S_j\right) \quad (3)$$

wobei  $W_{ij}$  das Gewicht der Verbindung vom Neuron  $j$  zum Neuron  $i$  bezeichnet. Ein vorgegebener Netzwerk-Input wird durch die Aktivitätszustände der Input-Neuronen dargestellt und der entsprechende Netzwerk-Output durch die Zustände der Output-Neuronen. Werden die Aktivitätszustände dieser speziellen Neuronen mit  $I_i$  bzw.  $O_i$  bezeichnet, so kann die Funktion, die das Netzwerk ausübt, durch

$$O_i = O_i(\{I_i\}, \{W_{ij}\}) \quad (4)$$

beschrieben werden. Das Netzwerk realisiert also eine Abbildung, die einem Inputvektor  $\{I_i\}$  einen Outputvektor  $\{O_i\}$  zuordnet. Die genaue

Form dieser Abbildung ist durch die Netzwerkarchitektur und die Verbindungsgewichte  $W_{ij}$  bestimmt, das heißt in einem Neuronalen Netzwerk ist die Information delokalisiert im Muster der Verbindungsgewichte gespeichert und nicht in einer separaten Datenbank. Das erklärt die Robustheit von Neuronalen Netzwerken in bezug auf den Ausfall von einzelnen Komponenten.

Ein Multilayer-Perceptron kann im Prinzip beliebig komplexe Input-Output-Zusammenhänge modellieren [3], allerdings nur wenn genügend viele versteckte Neuronen zur Verfügung stehen. Die optimale Wahl der Netzwerkgröße ist daher ein kritisches Problem, das meist nur mit empirischen Methoden gelöst werden kann. Ist die Netzwerkarchitektur einmal festgelegt, müssen noch die Gewichte  $W_{ij}$  so bestimmt werden, dass das Neuronale Netzwerk seine Aufgabe möglichst gut löst. Dies geschieht in einem sogenannten Lernprozess, in dem die Gewichte aufgrund von Input-Output-Beispielen sukzessive angepasst werden. Diese Fähigkeit, aus Beispielen zu lernen, ist eine der wichtigsten Eigenschaften von Neuronalen Netzwerken, und soll deshalb im nächsten Abschnitt noch etwas genauer beschrieben werden.

## Aus Beispielen lernen

Das Lernen in Neuronalen Netzwerken ist ein Prozess, in dem die Effizienz des Netzwerks bezüglich einer vorgegebenen Aufgabe optimiert

wird. Wie diese Effizienz gemessen oder bewertet wird, hängt von der jeweiligen Problemstellung ab. In vielen Fällen steht eine Anzahl von Lernbeispielen zur Verfügung, das heisst für gewisse Inputvektoren  $\{I_i^v\}$ ,  $v = 1, \dots, N$ , ist der zugehörige Soll-Output-Vektor, der mit  $\{D_i^v\}$  bezeichnet wird, bekannt. Dann wählt man oft den quadratischen Outputfehler, summiert über alle Lernbeispiele,

$$F = \sum_{v=1}^N F^v, \quad F^v = \frac{1}{2} \sum_i (D_i^v - O_i^v)^2, \quad (5)$$

als Mass für die Effizienz des Neuronalen Netzwerks. Für einen vorgegebenen Inputvektor  $\{I_i^v\}$  hängt der Netzwerkoutput  $\{O_i^v\}$  von den Gewichten  $W_{ij}$  ab, und diese sollen nun in der Lernphase so angepasst werden, dass der totale Outputfehler  $F$  minimal wird. Die meisten Lernstrategien benutzen ein Gradientenverfahren zur Bestimmung eines optimalen Satzes von Verbindungsgewichten. Dabei werden die Lernbeispiele  $\{I_i^v\}/\{D_i^v\}$  in zufälliger Reihenfolge immer wieder präsentiert und die Gewichte  $W_{ij}$  jedesmal um einen Betrag  $\Delta W_{ij}$  verändert, der proportional zum negativen Gradienten des jeweiligen Outputfehlers ist:

$$\Delta W_{ij} = -\eta \frac{\partial F^v}{\partial W_{ij}} \quad (6)$$

Dieses Verfahren konvergiert gegen ein Minimum des totalen Fehlers  $F$ , vorausgesetzt, dass der Proportionalitätsfaktor  $\eta$  genügend klein gewählt wird. Für Feedforward-Netzwerke stellt der berühmte Backpropagation-Lernalgorithmus [3;7] eine besonders effiziente Implementierung einer solchen Gradientenmethode dar. Er ist heute das wohl am häufigsten verwendete Lernverfahren für Neuronale Netzwerke.

Wie alle Gradientenverfahren leidet aber auch der Backpropagation-Algorithmus unter einer Anzahl von Problemen. Die Fehlerfunktion  $F$  enthält nämlich oft sehr flache Täler oder Plateaus, was zu extrem langen Lernzeiten führt. Besitzt  $F$  zudem mehrere lokale Minima, so kann der Algorithmus in einer relativ schlechten Lösung steckenbleiben. In vielen Fällen ist es deshalb nötig, zusätzliche Tricks anzuwenden oder effizientere Optimierungsverfahren zu benutzen [3;8;9].

Die soeben beschriebene Art des Lernens ist für viele Anwendungen von Neuronalen Netzwerken geeignet, zum Beispiel für die Modellierung von komplexen physikalischen oder technischen Zusammenhängen oder für die Entwicklung von Diagnose- und Prognosesystemen. Die Lernbeispiele sind dann experimentelle Daten, die entweder bereits zur Verfügung stehen oder generiert werden können.

Andere interessante Anwendungen von Neuronalen Netzwerken beziehen sich auf das Lernen einer Strategie, zum Beispiel einer Regel- oder Steuerstrategie. In solchen Fällen ist der gewünschte Netzwerkoutput für die Lernbeispiele oft nicht explizite bekannt, und die Effizienz des Netzwerks kann erst nach einer gewissen Zeit beurteilt werden. Die obigen Verfahren zur Minimierung des Outputfehlers sind dann nicht direkt anwendbar, und es müssen komplizierte Lernstrategien entwickelt werden [9].

Neben dem Lernverhalten interessiert man sich natürlich auch für das Verallgemeinerungsverhalten von Neuronalen Netzwerken. Verallgemeinerung bezieht sich dabei auf die Art und Weise, wie das Netzwerk nach der Lernphase neue, unbekannte Inputdaten verarbeitet.

## Analyse und Anwendung neuronaler Lernstrategien

Die detaillierte Analyse der Lern- und Verallgemeinerungseigenschaften

von Neuronalen Netzwerken ist Voraussetzung für eine optimale Implementierung und Anwendung entsprechender Lernverfahren. Dieser Aspekt hat deshalb in unseren Untersuchungen eine zentrale Bedeutung. Ein weiterer wichtiger Punkt ist der Vergleich verschiedener neuronaler Lernstrategien und der Vergleich mit konventionellen Lösungsansätzen. Die folgenden Ausführungen geben einen Überblick über die verschiedenen Studien, mit denen wir uns zurzeit befassen und über bereits vorliegende Resultate.

Die Anwendung von Neuronalen Netzwerken auf Steuer- und Regelprobleme stellt spezielle Anforderungen an die Entwicklung geeigneter Lernstrategien. Als Testproblem für unsere Untersuchungen haben wir das Balancieren eines Stabes gewählt. Das entsprechende System ist in Bild 3 dargestellt. Der Stab ist auf einem Wagen montiert, der sich auf einer etwa 4 m langen Laufschiene bewegen kann und von einem Motor angetrieben wird. Sensoren messen den Drehwinkel  $\varphi$  des Stabes, die Drehgeschwindigkeit  $\dot{\varphi}$ , sowie die Position  $x$  und die Geschwindigkeit  $\dot{x}$  des Wagens. Diese Messdaten werden als Input einem Neuronalen Netzwerk zugeführt, das die Aufgabe hat, daraus die zur Stabilisierung des Stabes erforderliche Beschleunigungskraft  $F$  zu bestimmen. Für die Simulation des Neuronalen Netzwerks wurde ein Multittransputer-System entwickelt, das sich speziell für Echtzeitanwendungen eignet [10]. In einer ersten Studie wurde das neuronale Netz auf-

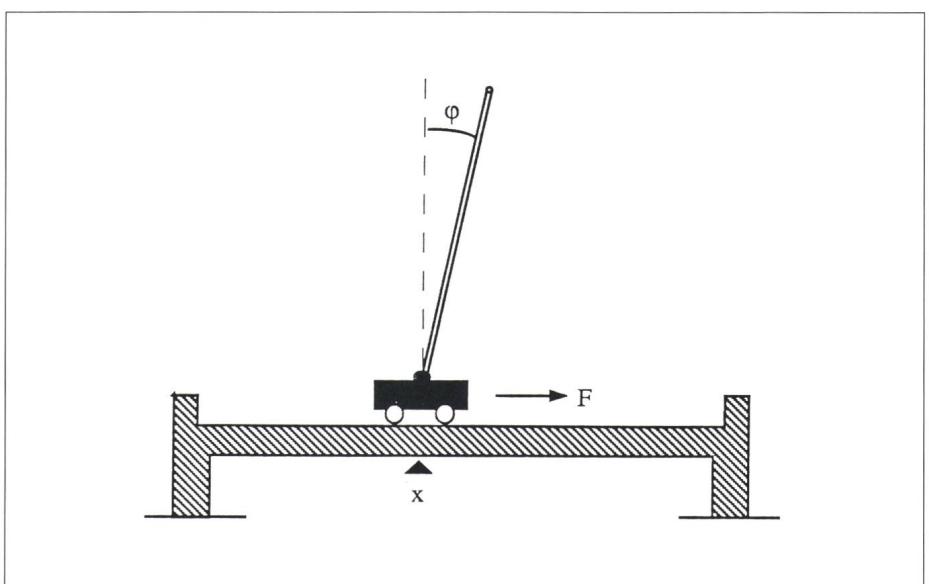


Bild 3 Balancieren eines auf einem Wagen montierten Stabes

grund einer vorgegebenen Regelstrategie trainiert. Diese stellt den «Lehrer» dar, der während des Regelprozesses die notwendigen Lernbeispiele liefert. Nach der Lernphase erweist sich die neuronale Regelung als sehr robust in bezug auf den Ausfall einzelner Komponenten, was durch das Unterbrechen von Netzwerkverbindungen getestet werden kann. Ein entsprechendes Demonstrationssystem wird an der Heureka-Ausstellung in Zürich gezeigt, wobei das neuronale Netz in Analog-Hardware realisiert wurde.

Wenn keine Regelstrategie bekannt ist, muss das Neuronale Netzwerk diese von Grund auf selbst lernen. Eine unmittelbare Bewertung des Netzwerkoutputs ist dann aber nicht mehr möglich, und die Effizienz der neuronalen Regelung kann erst nach einer gewissen Zeit, dass heisst in der Zukunft, beurteilt werden [9;11]. Ohne näher auf diese Problematik einzugehen, sei nur angedeutet, dass beim Stab-Balancieren diese Effizienz zum Beispiel mit einer integralen Fehlerfunktion der Form

$$G(t) = \int_t \left( \Delta\varphi^2 + \Delta x^2 \right) \quad (7)$$

gemessen werden kann. Die Funktion (7) soll möglichst klein sein. Dabei bezeichnet  $\Delta\varphi$  die Differenz des Drehwinkels bezüglich der Senkrechten und  $\Delta x$  die Abweichung der Wagenposition vom Mittelpunkt der Laufschiene. Eine elegante Möglichkeit, die neuronale Regelung kontinuierlich zu verbessern, beruht auf der Verwendung eines zusätzlichen Neuronalen Netzwerks, des sogenannten «Kritikers» [11]. Der Kritiker lernt, die Bewertungsfunktion  $G(t)$  immer genauer vorherzusagen, und das Regler-Netzwerk benutzt diese Vorhersage um seine Strategie zu optimieren. Erste Versuche haben das prinzipielle Funktionieren dieser kombinierten Lernstrategie nachgewiesen.

Weitere Studien befassen sich mit der Anwendung Neuronaler Netzwerke auf Probleme aus den Bereichen Mustererkennung, Diagnose, Prognose und Optimierung:

Bei der Messung von Stromsignalen wird der Sekundärstrom durch magnetische Sättigungseffekte im Stromwandler verzerrt. Die Rekonstruktion des Primärstroms aus dem verzerrten

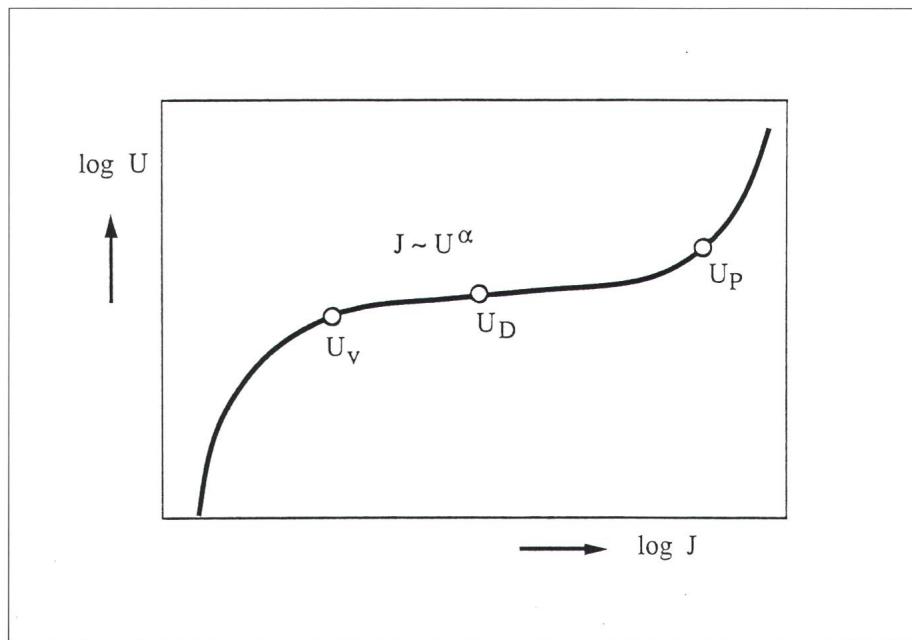


Bild 4 Strom-Spannungs-Kennlinie eines ZnO-Varistors (schematisch)

$U_V$  Betriebsspannung  
 $U_D$  Durchbruchsspannung  
 $U_P$  Norm-Überspannung  
 $\alpha$  Nichtlinearitätskoeffizient

Sekundärstrom ist ein wichtiges Problem bei Schutzanwendungen, das im Prinzip mit numerischen Methoden gelöst werden kann. Diese sind aber sehr rechenintensiv und setzen eine genaue Kenntnis der Stromwandlerparameter voraus. In einer Studie wird deshalb abgeklärt, ob Neuronale Netzwerke diese Aufgabe aus Beispielen von verzerrten Stromsignalen lernen und dann im Echtzeit-Betrieb durchführen können.

Eine weitere Anwendung betrifft die Diagnose von Isolationssystemen. Aufgrund von gemessenen Teilentladungsmustern soll ein Neuronales Netzwerk den Zustand der Isolation und die Art der Defekte erkennen und diagnostizieren. Das Teilentladungsmuster liegt in der Form eines zweidimensionalen Histogramms vor, das die Anzahl der über eine Zeitspanne gemessenen Teilentladungen als Funktion ihrer Phasenlage und ihrer Stärke wiedergibt. Ein Hauptproblem liegt nun darin, aus diesen Daten geeignete Inputdaten für das Neuronale Netzwerk herauszukristallisieren, so dass ein möglichst effizientes Lern- und Verallgemeinerungsverhalten resultiert. Eine detaillierte Analyse der vorhandenen Daten, kombiniert mit physikalischen Modellrechnungen, hat zu verschiedenen Vorschlägen für ein solches Preprocessing geführt.

Darauf basierende Diagnosesysteme sollen nun getestet und miteinander verglichen werden.

In Zusammenarbeit mit der SKA Zürich studieren wir den Einsatz von Neuronalen Netzwerken bei der Prognose von Devisenkursen. Insbesondere werden Methoden untersucht, die es erlauben, die Zuverlässigkeit einer Prognose quantitativ abzuschätzen und entsprechende Konfidenzintervalle zu bestimmen. Neuronale Netzwerke, die eine Wahrscheinlichkeit oder eine Wahrscheinlichkeitsverteilung lernen, scheinen in diesem Zusammenhang von besonderem Interesse zu sein. Solche Netzwerke können zum Beispiel entscheiden, in welchen Situationen eine Prognose so unsicher wird, dass es vorteilhafter ist, keine Aussage zu machen. Das Studium von vereinfachten Prognose-Problemen hat gezeigt, dass dadurch die Vorhersagegenauigkeit in den übrigen Fällen beträchtlich erhöht werden kann.

Eine ausführliche Studie, die weitgehend abgeschlossen ist, befasste sich schliesslich mit der Verwendung von Neuronalen Netzwerken zur Optimierung der Zusammensetzung von ZnO-Varistoren. Die entsprechenden Untersuchungen wurden zum grossen Teil im Rahmen einer ETH-Diplomarbeit durchgeführt [12]. Das gewähl-

te Vorgehen erwies sich als sehr viel-versprechend und wird im nächsten Abschnitt näher vorgestellt.

## Optimierung der Dotierung von ZnO-Varistoren

Varistoren (Variable Resistors) sind keramische Bauelemente für den Überspannungsschutz [13]. Speziell im Mittel- und Hochspannungsbereich basieren Ableiter heute auf solchen Schutzelementen. Varistoren bestehen zur Hauptsache aus Zinkoxid (ZnO), einem Halbleiter mit grosser Energielücke. Durch Zugabe geeigneter Dotierstoffe kann diesem Material im polykristallinen Verbund eine stark nichtlineare Strom-Spannungskennlinie verliehen werden (Bild 4). Beim Überschreiten einer materialabhängigen Schwellenspannung  $U_D$  schalten solche Elemente innerhalb Nanosekunden und reversibel von einem isolierenden in einen gut leitenden Zustand. Die elektrische Charakteristik der Varistoren hängt in komplexer Weise von der Dotierung und von der Temperaturlösung beim Sinterprozess ab. In der technischen Ausführung werden bis zu zehn verschiedene Dotierstoffe verwendet, welche in sehr unterschiedlichen Konzentrationsbereichen (ppm bis %) vorliegen. Die Wahl dieser Zugaben erfolgt heute fast ausschliesslich aus rein empirischen Überlegungen, und der Hersteller sieht sich mit der beinahe unlösbarer Aufgabe konfrontiert, ein bis zu 10komponentiges System bezüglich verschiedener elektrischer Kenngrössen zu optimieren.

Die wichtigsten Kenngrössen sind die im Betriebspunkt  $U_V$  gemessene Verlustleistung  $P_V$ , das Restspannungsverhältnis  $R_V = U_P/U_D$ , wobei  $U_P$  eine durch einen Normstrompuls erzeugte Überspannung bezeichnet, und der Nichtlinearitätskoeffizient  $\alpha = d(\log J)/d(\log U)$ , der bei einer Normstromdichte unterhalb des Betriebspunktes gemessen wird. Um eine gute thermische Stabilität zu gewährleisten, sollte die Verlustleistung  $P_V$  möglichst klein sein. Das Restspannungsverhältnis  $R_V$  zeigt an, wie gut sich das «Stromventil» bei Überspannung öffnet.  $R_V$  sollte also auch möglichst klein sein. Der Nichtlinearitätskoeffizient  $\alpha$  hingegen muss für eine gute Schaltcharakteristik möglichst gross sein. Eine vierte Kenngrösse ist der Langzeitstabilitätsparameter  $P_V$ , die bei 500 oder 1000 Stunden Be-

triebsdauer gemessene zeitliche Änderung der Verlustleistung. Für ein gutes Betriebsverhalten muss  $P_V \leq 0$  erfüllt sein.

Bei ABB wurde eine umfangreiche Studie durchgeführt, die darauf hinzielte, die Zusammenhänge zwischen Dotierung und elektrischem Verhalten von ZnO-Varistoren in den Griff zu bekommen. Ausgehend von einer empirisch optimierten Referenz-Zusammensetzung wurde zuerst der Einfluss jedes Dotierelementes einzeln untersucht. Mit einem Satz von 5 Elementen, die alle einen besonders starken Einfluss zeigten, wurde dann ein systematischer Variationsversuch durchgeführt, der  $2^5 = 32$  verschiedene Zusammensetzungen umfasste. Die statistische Analyse dieser experimentellen Daten mit konventionellen Methoden ergab keine befriedigenden Ergebnisse, was auf die starken Wechselwirkungen zwischen den Dotierelementen und auf die teilweise extrem nichtlinearen Abhängigkeiten zurückgeführt werden kann.

Aus diesem Grund haben wir versucht, das schwierige Optimierungsproblem mit Hilfe der Lern- und Verallgemeinerungsfähigkeit von Neuronalen Netzwerken zu lösen [12]. Als Lernbeispiele standen experimentelle Daten für insgesamt etwa 50 verschiedene Dotierungen zur Verfügung. Der Netzwerkinput bestand aus den Konzentrationen der 5 Dotierelemente und der Output sollte die zugehörigen 4 elektrischen Kenngrössen anzeigen. Die unterschiedlichen Konzentrationsbereiche der einzelnen Elemente und die inhomogene Verteilung der Lerndaten erforderten ein relativ aufwendiges Preprocessing (z.B. Skalierungen) sowohl der Input- als auch der Outputdaten. Es stellte sich heraus, dass ein sorgfältiges Preprocessing die Lern- und Verallgemeine-

rungseigenschaften der neuronalen Netze beträchtlich verbessern kann.

Die Netzwerke wurden mit dem Backpropagation-Lernverfahren trainiert, und die Auswahl von geeigneten Netzwerk-Architekturen erfolgte aufgrund einer sogenannten Crossvalidation-Methode. Dabei werden die Voraussagen von trainierten Netzwerken mit Messdaten verglichen, die im Lernset nicht berücksichtigt wurden. Die Untersuchungen ergaben, dass Feedforward-Netzwerke mit zwei Schichten von «versteckten» Neuronen die Aufgabe befriedigend lösen können. Ein Vergleich hat ferner gezeigt, dass Neuronale Netzwerke den Zusammenhang zwischen Dotierung und elektrischen Kenngrössen wesentlich zuverlässiger zu modellieren vermögen als eine polynomiale Regressionsanalyse.

Im eigentlichen Optimierungsprozess ersetzen die trainierten Netzwerke dann das Durchführen von aufwendigen Experimenten. Durch eine systematische Variation der Inputdaten können so Konzentrationsbereiche für die Dotierelemente identifiziert werden, die zu optimalen Werten für die elektrischen Kenngrössen führen sollten. Dabei werden nur diejenigen Voraussagen berücksichtigt, die durch mehrere, verschiedene trainierte Netzwerke bestätigt werden. Dadurch werden die Prognosen zwar etwas konservativer, ihre Zuverlässigkeit hingegen wird beträchtlich erhöht.

Die Vorhersagen der Neuronalen Netzwerke in bezug auf optimale Dotierkonzentrationen werden zurzeit experimentell überprüft. Die ersten Resultate sind sehr vielversprechend (Tab. I).

Bei der Bewertung der erzielten Verbesserungen muss berücksichtigt werden, dass das Verbesserungspotential der einzelnen Kenngrössen

Zusammensetzung	$P_V$		$R_V$		$\alpha$	
	Prog.	Exp.	Prog.	Exp.	Prog.	Exp.
1	- 25 %	- 28 %	- 3.1 %	- 7.9 %	+ 48 %	+ 65 %
2	- 15 %	- 36 %	- 2.5 %	- 8.0 %	+ 30 %	+ 78 %
3	- 41 %	- 37 %	- 2.5 %	- 4.6 %	+ 74 %	+ 96 %

**Tabelle I** Prognostizierte (Prog.) und experimentell gemessene (Exp.) Verbesserungen der Kenngrössen  $P_V$ ,  $R_V$  und  $\alpha$  in bezug auf das Referenzmaterial.

Die drei Zusammensetzungen wurden aufgrund der Modellierung durch Neuronale Netzwerke ausgewählt

sehr unterschiedlich ist und dass sich nicht nur in bezug auf das Referenzmaterial, sondern auch im Vergleich mit den besten Lerndaten zum Teil noch beträchtliche Verbesserungen ergeben. Zudem erfüllen alle drei Zusammensetzungen das Langzeit-Stabilitätskriterium  $P_V \leq 0$  nach 1000 Stunden Betriebsdauer. Die Tatsache, dass die gemessenen Verbesserungen durchwegs grösser sind als prognostiziert, lässt sich mit der eher konservativen Beurteilung der Netzwerk-Vorhersagen erklären.

## Ausblick

Die bisherigen Erfahrungen und Ergebnisse haben uns dem Hauptziel unserer Studien, die Einsatzfähigkeit von Neuronalen Netzwerken in industriellen Anwendungen zu demonstrieren, einen wesentlichen Schritt näher gebracht. Die Optimierung von ZnO-Varistoren bezog sich zwar nur auf die elektrischen Kenngrössen und auf deren Abhängigkeit von der Dotierung. Der Einfluss der Prozessparameter und die Auswirkungen auf andere, für die Fabrikation wichtige Kenngrössen wurden nicht in Betracht gezogen. Die erzielten Resultate zeigen aber eindeutig, dass auf Neuronalen Netzwerken basierende Methoden bei der Optimierung von Materialeigenschaften mit Erfolg eingesetzt werden können. Auch unsere Untersuchungen von Steuer- und Regelstrategien demonstrieren die potentiellen

Vorteile von Neuronalen-Netzwerk-Lösungen in praktischen Anwendungen. Weitere Studien, die zum Teil erst vor kurzem aufgenommen wurden, zielen ebenfalls daraufhin, vielversprechende Anwendungsbereiche für Neuronale Netzwerke zu identifizieren und diese Methoden dann optimal einzusetzen.

Zusammenfassend kann aber bereits jetzt festgehalten werden, dass selbst relativ einfache Neuronale Netzwerke fähig sind, komplexe Probleme aus verschiedenen Bereichen effizient und ökonomisch zu lösen, falls genügend Lernbeispiele zur Verfügung stehen. Voraussetzung dazu sind allerdings ein fundiertes Verständnis des Lern- und Verallgemeinerungsverhaltens solcher Systeme und ein intelligentes Preprocessing der Lerndaten.

## Dankwort

Mein Dank gilt in erster Linie meinen Mitarbeitern E. Cozzio, E. Scheiben, Dr. H.J. Wiesmann und Dr. F. Greuter, die für die beschriebenen Projekte massgeblich mitverantwortlich sind, sowie den Herren L. Grolimund und H. Häutle, die mit ihrer Diplomarbeit wesentlich zum Erfolg der Varistorstudie beigetragen haben. Dank gebührt schliesslich auch dem Schweizerischen Nationalfonds, der unsere Projekte im Rahmen des Nationalen Forschungsprogramms 23, Künstliche Intelligenz und Robotik, finanziell unterstützt.

## Literatur

- [1] R.P. Lippmann: An Introduction to Computing with Neural Nets. IEEE ASSP Magazine (April 1987), p. 4...22.
- [2] J.-F. Leber und M.B. Matthews: Neurale Netzwerke: Eine Übersicht. Bulletin SEV/VSE 80 (1989) 15, p. 923...932.
- [3] J. Hertz, A. Krogh and R.G. Palmer: Introduction to the Theory of Neural Computation. Redwood City/California, Addison – Wesley Publishing Company, 1991.
- [4] Y. le Cun: Generalization and Network Design Strategies. In: Connectionism in Perspective. Edited by R. Pfeifer et al., Amsterdam, North-Holland, 1989, p. 143...155.
- [5] P.M. Shea and V. Lin: Detection of Explosives in Checked Airline Baggage Using an Artificial Neural System. In: International Joint Conference on Neural Networks (Washington 1989), vol. II. New York, IEEE, 1989, p. 31...34.
- [6] M. Kuperstein and J. Rubinstein: Implementation of an Adaptive Neural Controller for Sensory-Motor Coordination. In: Connectionism in Perspective. Edited by R. Pfeifer et al., Amsterdam, North-Holland, 1989, p. 49...61.
- [7] D.E. Rumelhart, G.E. Hinton and R.J. Williams: Learning Representations by Back-Propagating Errors. Nature 323 (1986), p. 533...536.
- [8] J. Bernasconi: Learning in Neural Networks. In: Dynamics and Stochastic Processes. Edited by R. Lima, L. Streit and R. Vilela Mendes. Springer Lecture Notes in Physics 355 (1990), p. 42...54.
- [9] J. Bernasconi: Learning and Optimization. In: Statistical Mechanics of Neural Networks. Edited by L. Garrido. Springer Lecture Notes in Physics 368 (1990), p. 121...136.
- [10] E. Scheiben and A. Scheidegger: Codegeneratör zur Implementierung von neuronalen Netzwerken auf Transputersystemen. Diplomarbeit ETH Zürich, Abt. III B, Dezember 1989.
- [11] P.J. Werbos: A Menu of Designs for Reinforcement Learning over Time. In: Neural Networks for Control. Edited by W.T. Miller III, R.S. Sutton and P.J. Werbos. Cambridge/Massachusetts, MIT Press, 1990, p. 67...95.
- [12] L. Grolimund und H. Häutle: Optimierung der Dotierung von ZnO-Varistoren mit Hilfe von neuronalen Netzwerken. Diplomarbeit ETH Zürich, Abt. III B, Wintersemester 1990/91.
- [13] F. Greuter, R. Perkins, M. Rossinelli und F. Schmückle: Der Metalloxid-Widerstand: Kernelement moderner Überspannungsableiter. ABB Technik 1 (1989), p. 35...42.